for editors. Either they can focus on a particular area or orientation and neglect a substantial amount of relevant work, such as in the collection by Galegher *et al.* (1990) and the later books in the Springer-Verlag series, or they can present the research as a set of unstructured readings (e.g. Baecker, 1993; Greif, 1988). This collection has adopted an approach that is something of a compromise: a lot of short papers largely focused on one topic—multi-media video-conferencing systems. However, even on this subject there is already a large and dispersed literature including work by telecommunications engineers, computer scientists, psychologists and sociologists. Four chapters of largely introductory material tend to drift between distinguishing previous work chronologically and technologically, between when it was done and what it does. As an exception, the chapter by Gale concisely provides technical details, including descriptions of standards for video conferencing, available speeds of transmission and actual prices of equipment.

The majority of the remaining chapters report research undertaken in Europe on various issues associated with multi-media systems. These include descriptions of the various tools that can be provided to support collaborative work, possibilities for interface design and the computational architectures necessary for the system development. Unfortunately, by including a large number of chapters each contribution tends to be brief, resulting in little discussion of the advantages of these systems or the difficulties individuals have using them. The few evaluations are derived from small-scale experiments, sometimes by the developers themselves and are generally reported anecdotally. Moreover, the results tend to be rather ambivalent concerning the success of the technology. It is particularly problematic to assess the effect of collaborative systems on group work and organizations; however, a discussion of the use of such a system in a real-world setting would have been useful, particularly for those considering deploying the new products and services. Although the chapter by Land directly addresses problems related to the introduction of technology, his contribution discusses Information Technology in general. As the other contributors repeatedly emphasize, deployment of collaborative systems raises novel and additional problems.

At present, there is a further difficulty in collecting together research in CSCW. The continuing technological and methodological developments in the field means that papers can become out-of-date very quickly. Unfortunately, little modification appears to have been made to the papers in this collection since the original seminar in 1992. This means that recent related developments in Japan, the USA and Europe are not referred to and this leads to some curiosities. For example, Cullen reports the aims and goals of the European RACE I research programme at a time when the subsequent RACE II programme is coming to an end. Finally, given the time the book has taken to be published and its price, it is a shame that the production quality is so poor, with a large number of typesetting errors, some of which are rather distinctive.

A. PAUL LUFF
University of Surrey

REFERENCES

Baecker, R. M. (eds) (1993) *Readings in Groupware and Computer-Supported Cooperative Work*. Lawrence Erlbaum Associates, Hillsdale, NJ.

Galegher, J., Kraut, R. E. and Egido, C. (eds) (1990) *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work*. Lawrence Erlbaum Associates, Hillsdale, NJ.

Greif, I. (eds) (1988) *Computer-Supported Cooperative Work: A Book of Readings*. Morgan Kaufmann, San Mateo, CA.

PHIL PICTON

Introduction to Neural Networks. Macmillan. 1994. ISBN 0-333-61832-7. £13.99. 168 pp. softbound.

The aim of this book is to introduce neural networks to the technically-minded general reader. It opts for a largely routine organization with chapters on topics such as adalines, perceptrons, Boolean nets, associative memory, probabilistic networks and self-organizing nets.

The title of the first chapter is a promising one: 'What is a neural network?'. Unfortunately, the author fails to provide the general reader with a sure footing. He asserts that neural networks are pattern classifiers but also that pattern classification is just one operation of a neural network. The fog had not cleared by Chapter 5 where we find that a neural net can do other things besides pattern classification but (*errm*) 'as a kind of consequence of its pattern classifying ability'.

For neural networks the excitement really ain't what they do, but the way that they do it. The author sows seeds of confusion for the general reader here too. One thing which sets neural nets apart from other classification systems is their ability to respond intelligently to noisy input and to patterns not previously encountered. The author indicates that this is so, but says that an Adaline would 'behave unpredictably' in the presence of a yet-unseen pattern, surely losing the sympathy of his engineering audience at this point. When the term 'graceful degradation' is introduced there is a weird restraint in the exposition, as if the author had heard of, but didn't really believe, the claims '*it is supposed that* if part of the network malfunctions, the whole system could still continue to operate' and the even weirder '*it is said that* this is preferable to complete failure' (emphasis added).

There is much to bemoan in the rest of the text. There are technical weaknesses which are most apparent in the material on Hopfield nets. For example, the author asserts that asynchronous updating is essential for the correct operation of a Hopfield net, whereas synchronous updating is equally acceptable. The author continually speaks of the existence of local minima in

the energy landscape as if they were a problem for Hopfield nets, although it is only because of their existence that patterns are retrievable. So-called *spurious minima* are the problem and probabilistic networks help to shake the net from these (relatively shallow) minima, to deeper minima, not to a global minimum as the author implies.

Since the text is for the technically-minded general reader it presumes no particular mathematical knowledge. However, little attention is paid to the difficulties which the absence of such knowledge might present. In his earliest exposures to mathematical derivations in the text, the general reader is left without even the odd 'where' or 'therefore' to light the way. Unlike many such texts, no mathematical appendix is offered for reference.

The book has the potential for confusing and misleading, and is not one which I would recommend as a starting point.

T. W. ROUTEN
De Montfort University

ROGER S. PRESSMAN (adapted by DARRELL INCE)
Software Engineering: A Practitioner's Approach. McGraw-Hill. 1994. ISBN 0-07-707936-1. £22.95. 801 pp. softbound.

This is the European, and scarcely changed, version of the third edition of Pressman's *Software Engineering*. Current trends in this subject point to textbooks weighing several tons by the next millenium, but the increment from the (1992) international third edition to this one is tiny. Darrel Ince has added a few words on software quality, including ISO 9001 (a topic as fascinating as watching paint dry) and has rewritten the section on formal methods, mostly Z (wisely taken from Spivey's 1990 paper in *IEEE Software* for the international edition). 'European' does not mean 'British English'.

Pressman's original text was born in 1982, and has thriven and spread ever since. But, as the weight of these general texts increases, their content thins. If you wanted to know about any of the topics covered—say, database normalization, data structures, object-orientation, or CASE—you would not read a book on 'Software Engineering'. And if you wanted a general overview of Software Engineering, you would not read 800 pages. However, if you really do want one of these massive tomes, Pressman's is as good as any and better than some.

But what has Ince done for the book? Not enough to justify a new edition; and worse: once again we have claims for rigour made by a writer who has not exercised even *care*. The term 'subset' is introduced without definition, and the *subset* and *proper subset* operators are given the interpretations of each other. The *union* operator is carelessly called 'intersection' and defined thus: *The ... operator takes two sets and forms a set that contains all the elements in the set with duplicates eliminated.* The conditional ('if ... then') operator is (as ever) misleadingly called 'implies' and is introduced without definition. On sequences, we read: *Since a sequence is set of pairs, then all the set operators described in the previous subsection are applicable.* If you can ignore the syntax of that sentence, and think about its meaning, you will appreciate that 'applicable' is used in a Pickwickian sense: without closure. A sequence of n things is defined as a set of n ordered pairs, the first members of those pairs being the integers 1 to n. So you can form (say) the union of two sequences, but it is unlikely itself to be a sequence. As it happens, Ince's carelessness provides him with a logically impeccable defence. The subsection on sets is not 'the previous subsection', which is about logic, and describes no set operator.

In the international edition, Pressman wisely says that *[a]mbiguity, incompleteness, and inconsistency can be discovered and corrected more easily* using formal methods. Ince substitutes the claim that *there is little possibility of ambiguity, specifications can be mathematically validated for contradictions and incompleteness, and vagueness disappears completely.* If you are waiting for Ince's proof procedure for set theory, don't hold your breath. Indeed, why not fill in the time by working as a much needed technical proof-reader for McGraw-Hill?

ADRIAN LARNER
De Montfort University

GUSS J. RAMACKERS
Integrated Object Modelling. Thesis Publishers Amsterdam. 1994. ISBN 90-5170-244-2, 255 pp. softbound.

In this book, the author has provided a clear exposition of an executable (and therefore validatable) formalization of IS/IT business analysis systems. A large proportion of the book attempts to prove that the proposed modelling formalism is beyond the reach of most case tools and thus requires an extension of such technologies. The major benefit of the proposed approach over existing case technologies is the facility to allow several different views of a system corresponding to the various users of such systems.

The described modelling framework combines all the familiar business and structured/MIS diagrammatic notations, as well as textual descriptions (as alternative representations). Underlying this formalism is a mathematical structure known as a high level Petri Net. This is essentially an algebraic extension of basic Petri Nets allowing for the denotation of objects *a la* O-O analysis/designs. Thus in addition to validation, i.e. Petri Nets are executable, due to the mathematical semantics of Nets, verification of internal consistency is also facilitated.

As a modelling paradigm there is no doubt that the present approach makes a significant advancement in