graphs. The book includes a chapter on aesthetics and technique which I found especially useful; containing simple and applicable guidelines on how to produce clear and attractive data graphics.

Tufte's theory incorporates a number of principles for producing good data graphics. Perhaps the most important of these is "maximize the data-ink ratio". One of Tufte's definitions of the data-ink ratio is the "proportion of a graphic's ink devoted to the non-redundant display of data information". According to this principle one should endeavour to minimize those parts of a graphic containing information available in other parts, and to delete areas which may be safely taken out without loss of data or comprehension. Therefore, grid lines and axes should be erased from a diagram where possible. Another principle is "eliminate chartjunk". Chartjunk is decoration or frills which add nothing to the data displayed and which may cause the data to be obscured. A common example of chartjunk may be seen in histographs, whereas sometimes line shading of alternate bars can make the bars appear to fall in on each other, or produce disturbing visual phenomena called Moiré effects.

It is good to see the wide range of publications from which material has been drawn. Figures are taken from such varied sources as The New Yorker and Pravda, with diagrams illustrating such graphical delights as the "Shrinking Family Doctor", and a Parisian train timetable set out as a time series graph.

My one problem with this book is that Tufte sometimes goes too far in his evangelical pursuit of graphical excellence. For example, the data-ink maximizing examples are praised as being much more communicative and effective than graphic which retain "redundant" features such as grids or labels but are, I suspect, far too bare to be regarded as aesthetic or easily interpretable by the average reader. It is also becoming increasingly difficult to implement the type of design decisions the author advocates. I suspect that nowadays hand-drawn graphics are very rarely produced for publications or reports, authors choosing instead to employ computer packages for this time consuming task and use the default graphic provided.

Throughout the book the author's style is straightforward, clear and readable. The 150 diagrams are well chosen and always seem to illustrate concisely points being made in the text. The contents are ordered logically, and the index is extensive. The quality of product is undeniably excellent and overall the book amounts to a true bargain for the price. I would recommend that this book be given to every researcher and scientist on their first day, with instructions to read, learn and enjoy.

Crafting knowledge based systems by Jon Walters and Norman Nielsen, J. Wiley Interscience, Chichester 1988, pp 342, £18.95.

What motivates a knowledge-based systems author to put pen to paper? There are probably as many reasons as authors. But in this case, I would hazard a guess that there is a degree of proglatizing for the authors' approach to building expert systems.

This approach is a little bit non-standard—as you may have guessed from the title. They put forward the philosophy that knowledge-based systems are arty (or artful) systems to the crafted, rather than scientific tools to be engineered. This sets the scene for a different slant on the subject which is at times highly interesting and at times rather unusual. Each chapter for example, starts with a photograph of some aspect of yacht building and relates it to knowledge-based systems production.

The book's covers modestly describes the content as "describing the entire process of building a knowledge-based application from concept to operational use". It goes on to promise that it is "packed with advice and examples based on the author's own experience", and that I would not dispute. In fact the content is divided into three parts, planning and design, development (or crafting) and knowledge representation. It is in planning and design that some of the author's more interesting outlooks are to be found. They advise, for example, that all knowledge elicitation interviews should be conducted by two knowledge crafters, who take it in turns to ask the questions.

They do admit that "term interviewing can intimidate an expert", but hold that "the gain in accuracy and amount of information obtained from using this technique usually outweighs any side effects".

The advice in this first section seems to be very much based on their particular experience and philosophy. Looked upon in this light it is quite worth reading. They cover all the important aspects of what you need to know before embarking on the development of a knowledge-based system. For a beginner however, the distinction between personal opinion, generally agreed procedure and hard facts may be difficult to identify.

I did find the development section rather more clear cut. They describe variations on the prototyping approach and give examples of appropriate situations for each. Their testing and evaluation chapter is particularly well presented and I was glad to see a chapter on project scheduling.

The knowledge representation section is mainly a catalogue of the methodologies for recording expert knowledge and strategies for its use. I did not find these examples helpful or practical, but perhaps I am suffering from the cultural difference between Europe and the author's Stateside experience.

So to whom is this text most appropriate? Should a beginner ever try to get to grips with the art, or science, of knowledge engineering from a book? I will side-step that question and say that in any case this would not be the book for that task. It could make interesting reading for knowledge-based systems practitioners who wish to broaden their point of view perhaps. The one category of reader to whom I can be certain in recommending this book is the apprentice knowledge crafters of Messers Walters and Nielsen's own team. These eager young workers are, they insist, an essential ingredient in every knowledge crafting team to ensure the future expansion of knowledge-based systems.

Knowledge engineering for expert systems by M Greenwell, Ellis Horwood, Chichester, pp 184, £25.00, ISBN 0 7458 0513 2.

Knowledge engineering by TB Cross, Brady, New York, pp 266, $34.95/£30.40, ISBN 0 13 516741 8.

"Knowledge engineering": it would seem reasonable to suppose the term has been used long enough for us to have arrived at a stable consensus about its meaning, without allowing it to diffuse into the lay world in a fuzzy state and suffer the increasingly-inevitable debasement. Indeed, in the glossary at the end of each of these recent books are to be found similar definitions: for Greenwell, "the process of building expert systems including identification, feasibility, knowledge acquisition, knowledge analysis, implementation, delivery and maintenance"; for Cross, just "the process of building expert systems". That, however, is about the only common ground, besides a single reference (the perennial 1983 book edited by Hayes-Roth, Waterman and Lenat, "Building Expert Systems") that the authors share. The major part of this review will therefore involve separate discussion of each work, before returning briefly to the meaning of the subject they share.

Mike Greenwell's book appeared in July 1988. Had it been a few months earlier, it would certainly have been cited in this author's recent review of knowledge acquisition (Neale, 1988), for it fills a gap in the market. The audience comprizes knowledge engineers, expert system project managers and potential experts. Greenwell's background is in cognitive psychology; he has been involved in knowledge engineering since at least 1985. His primary concern is the lack of an integrated methodology for building expert systems, and in outlining what should be done he concentrates on perhaps the most problematic area, knowledge acquisition (which subsumes knowledge elicitation). This is where knowledge engineering differs most from conventional systems analysis, because in addition to the external requirements for the system (what the user wants) there is the much harder task of identifying and modelling the expert's personal problem-solving knowledge ("perhaps" is used advisedly above, because although this is very much the conventional wisdom and not hard to justify, there is a case to be made for the most difficult phase of expert system building being validation and testing).