of these mutilated texts] . . . however . . . most of the . . . reports . . . suggest[ed] little attempt to grasp the development of the argument within the text.' Strange that. '[I]t is not expected that such a framework could be handled over to non-human scientists . . . ' with their funny pointy ears.

But what does he have to say? (That is, other than the incomprehensible pun in the dedication and the literary quotations in the chapter headings—*human* science, you see; science but not at we know it.) From the first four chapters we gather that most human factors experts have found nothing—except what is false, trivial or commonsense—to say about reading or any other aspect of system design and (consequently?) designers ignore them. From the last six chapters we learn, when designing a text presentation system, (1) to consider what kinds of information the texts convey, why people read them and how people access them; and (2) to work on the users' requirements (the 'Task Model'), the structures of the texts (the 'Information Model') and the methods of text handling to be provided ('Manipulation Skills and Facilities'); also, to get to understand something of how people read (the 'Serial Reading Processor').

There needs no ghost, my lord, come from the grave, to tell us this. (Hamlet Act 1, Sc. 5)

ADRIAN LARNER
de Montfort University

BRIAN RATCLIFF
Introducing Specification Using Z. McGraw-Hill. 1994. ISBN 0-07-70796505. £18.95. 308 pp. softbound.

This book is another 'practical' introduction to the Z formal specification language. How does it compare with John Wordsworth's *Software Development with Z* (Addison-Wesley 1992)? Here is Wordsworth introducing subsets and power sets: "One set is said to be a *subset* of another if all the members of the one are also members of the other . . . . The *power set* of a set . . . is the set of all its subsets." And here is Bryan Ratcliff: "Since a set is itself a (structured) value, it must have a type *and hence belong to a set itself* . . . . [I]f we declare . . . `setOfInt` to be a 'set-of-integers variable' . . . `setOfInt` can be *any* set whose elements are integers . . . To declare `setOfInt`, we would use the powerset operator [thus] . . . `setOfInt`:$\mathcal{P}\mathcal{X}$. For this to make sense, the expression $\mathcal{P}\mathcal{X}$ must denote a set whose elements *are sets themselves* . . . $\mathcal{P}\mathcal{X}$ must therefore denote *the set of all possible sets which can be constructed from the elements of* $\mathcal{X}$—that is, all different possible *subsets* of $\mathcal{X}$."

Z is a very large and complex notation for set theory, along with a schema language that facilitates its application to imperative programs. Any introduction—'practical' or not—has to convey a huge amount of theory. Alas, Ratcliff is neither logically nor linguistically up to the task. He remarks: "We often speak of a set 'containing' a member . . . . But you should understand that, strictly, a set *is* its consistituent members and is not some kind of 'container' with a certain contents *(sic)*." But if that were so, the empty set would be nothing; and Ratcliff would be wrong to assert that a set and its power set could have no common member: if x is a member of s then $\{x\}$ is a member of the power set of s, but if a set *is* its members, $\{x\}$ is x, which is therefore also a member of the power set of s. Nor is this an isolated mistake: he uses "substitute" to mean "replace"; he uses "axiom" to mean "atomic well-formed formula"; he says that a mathematical variable "denotes a unique element", which may be "non-specific"—which of the integers is it that is non-specific? He has heard about some problem of confusing logic with metalogic, so objects to the perfectly proper use of truth functions like "$\wedge$" *(and)* in metalogical propositions; but he says that tautologies are "propositional laws (of the predicate logic)", and identifies the formula, "$P \vee \neg P$", with the Law of Excluded Middle.

Unfortunately, the whole subject of formal specifications is riddled with confusion; worse confounded by Ratcliff. Firstly, the use of mathematics (rather than merely logic) does not bring rigour; mathematicians are notoriously non-rigorous; their notion of 'set' was shown incoherent by Russell and (they hope but cannot tell) can be saved from contradiction only by some *ad hoc* adjustment of the theory, which, in the case of Z, unfortunately prohibits sets having members of different types. Secondly, in formal systems, as in life, a good maxim is: take what you like and pay for it. If we take all the powers of set theory, it is no use complaining that we have had to sacrifice any mechanical theorem prover (but has Professor Dijkstra taught us nothing of simplicity and weakness?). Thirdly, the concepts of *notation* and *formality* are entirely orthogonal: nothing prevents us writing formal, unambiguous statements in English. How could it? Perhaps taught by a good introduction to Z, we write such statements in a formal notation and translate the notation into English: there are the required English statements—technical *and* 'human-understandable'. At least, they are understandable if we avoid Z-locutions like "The set of cats is a member of the power set of the set of animals", and say instead, "Each cat is an animal". But perhaps some people's grasp of English precludes all logic, formal or informal. Ratcliff interprets "he and she do not love each other" as "he does not love her *and* she does not love him"; and from his major case study scenario, on Van Hire, which makes no mention of fuel, he concludes that if one transit van runs on diesel so does any other.

"[D]iscussion of the precise relationship between proof and truth is outside the scope of this book." Sadly true.

ADRIAN LARNER
de Montfort University