program, titled *WinText*, *Windows Filer*, and *WinScan*, respectively. By the time you read this, those applications may already be available.

Windows Spell does what its name implies. Anytime you need to spell check a document created by a fair sampling of *Windows* products, you simply run *Windows Spell*, then open the document in question. Either on screen or in the background, *Windows Spell* then proceeds to check the document for spelling errors. Because of the way it operates (the program alerts you that it has found an error, then patiently waits for you to correct it before proceeding to the next problem), the background mode of operation is practical only in such specialized cases as final proofing of a long document, such as a novel, but it's welcome, nevertheless. Unfortunately, the blinking icon that is supposed to alert you to *Windows Spell* being in a waiting condition isn't properly implemented; the icon blinks once, then re-displays as a solid item.

Learning to use *Windows Spell* is easy. The documentation consists of a single thirty page pamphlet containing numerous illustrations, and can be covered in under an hour. You are taken on a sequential tour of the product, which implements a 130,000 word standard dictionary, and allows you to create multiple supplemental dictionaries consisting of the words unique to your circumstances or occupation. When *Windows Spell* encounters an error, you are presented with a list of alternative words to choose from, may skip the word once or for the remainder of the document, or may add it to your dictionary for future allowance. A feature of *Windows Spell* that we particularly like is that an icon representing an "active" version of the program changes color proportionally to indicate the percentage of your document that has undergone spell checking.

Windows Spell can check files created by *Window*'s own *Write* and *Draw*, as well as Micrografx's *In*A*Vision*, *Designer*, and *Graph Plus*, Microsoft's *Word* and *Excel*, Aldus' *Pagemaker*, WordPerfect's *WordPerfect*, and both ASCII and ANSI compliant text files. The product is dated, however. The current versions of *Designer* and an updated and renamed *Graph Plus* (now known as *Charisma*), are not yet supported, and only the straight DOS version of *Word* is compatible. Officials at Palsoft explained this by saying the file formats were different, without giving us a feel for when the issue might be addressed. In any event, *Windows Spell* is not smart enough to sense automatically the type of document you are working with - you must tell it.

There are also a couple of production problems. The illustrations in the manual look as if they were captured in a low resolution graphics mode and printed without being touched up, and the dated documentation (in fairness, since *Windows Spell* will still run under older versions of *Windows*, the age of the documentation can be forgiven) gives instructions in several place that are imprecise. For example, under "Starting *Windows Spell*" you are given instructions that work exactly as stated, but no mention is made of your need to be in a particular directory. Novices, heads up.

An earlier *Windows 3.0*-compliant version of *Windows Spell* (version 3.06) that was released had a serious bug in it. If you touched any of several keys while *Windows Spell* was either active or iconized (including CTRL, SHIFT, and TAB), while running in either standard or 386-enhanced modes, the system would be severely crippled, and messages telling you that applications had been terminated abnormally became the rule, rather then the exception. This problem has since been corrected, but our overall impression hasn't changed. It does it's job, but *Windows Spell* makes you work a bit harder then it ought to, and needs an serious update.

Q+E
Pioneer Software

Q+E is one of those products that we love to review. However, it's also the kind of product about which we feel exactly the opposite. We love *Q+E* because it does a few tricks that are very useful, yet not terribly common in other software, imposes very little system resource overhead, and even at full retail price, it's cheap. We hate *Q+E* because it defies easy definition, and makes the job of writing a review extremely difficult. Is *Q+E* a database? A query tool? A *Windows* tool? Well, it's all of the above, and then some. The question is, who needs it?

My Profile

We have exactly two bad things to say about *Q+E*. One is that the preceding question is not easily answerable. Whether this is a marketing problem or simply built into the product's nature is open to interpretation; we lean toward the latter. The utility of *Q+E* is so broad, it boggles the mind. Hmm . . . that sure didn't **sound** negative, did it? Complaint number two gets no such reprieve. Despite it's heritage (this product has been around for awhile, a version exists for OS/2, and no less an authority than Microsoft has teamed up with Pioneer to distribute the product), *Q+E* for *Windows 3.0* **doesn't multitask**. Execute a query or sort operation, and *Q+E* grabs control of your system until activity is complete. This isn't usually a big deal, but try performing a multiple join on moderately sized, unindexed databases. We did. There were three, numbering 975, 235, and 350 records, and the join was on a single, field, identically named in all three. Then we waited . . . and waited . . . and made coffee . . . and waited some more. All told, we waited more than five minutes, using a 386/20 machine in *Windows*'