

One, Stac Electronics' *Stacker*, has established a significant market presence, spending most of the last eighteen months on the software best sellers lists. Rightly so. *Stacker* works transparently, exacts no discernible performance penalty versus uncompressed disks on systems with 80286 processors or better, and after what can be a harrowing installation process, is compatible with pretty much anything you throw at it. *SuperStor*, in the recently-released version 2.0, is an equally viable solution to the problem of tight disk space. In fact, under some circumstances, it's better.

Before proceeding, we need to cover some ugly ground. AddStor rushed the first release of *SuperStor* to market so as not to give Stac too much of a head start with their initial release. The product was a disaster. It came in a box emblazoned with a cartoon that did nothing so well as scare potential buyers away, and that was the good news. *SuperStor* was alleged to come with a disk defragmenter suitable for work on the compressed files (it didn't), a better user interface than the competition (not really), and an algorithm that would run rings around anything else available—a false claim. The good news is, all of these problems have been eliminated, including a redesign of the box. No, it doesn't outperform *Stacker*, but it's no slouch, either.

In informal tests, we came to the conclusion that *SuperStor* typically achieves a compression ratio 10-15% less efficient than *Stacker*'s, and that the speed of program code execution from disk lagged by a similar amount. We checked both program/data storage and load operations with *Excel*, *FoxPro*, *WordPerfect*, and the Paintbrush tool that comes with *Windows* to come to this conclusion. Examples include *Stacker* compressing collections of .DBF and .PCX files at 1.7:1 and 2.4:1, while *SuperStor* managed only 1.6:1 and 2.1:1, respectively. On the other hand, *SuperStor*-compressed software making only occasional disk access seems to perform marginally faster than programs residing on *Stacker* volumes after it's been loaded, and certain hard-to-compress files, such as .ZIP volumes, compress a bit better under *SuperStor* than with *Stacker*, which seldom manages any savings at all.

*SuperStor* includes two features that *Stacker* can't touch. Although you can use *Stacker* to compress data stored on floppy disks and other removable media, the files are useless on systems not equipped with *Stacker*. *SuperStor* provides a distributable software driver that lets you transport compressed data between machines, (but you can't write to it without decompressing it). Also, *SuperStor* can be configured—and by default, is—to recognize compressed floppies automatically. Using floppies on a *Stacker*-equipped system is much more cumbersome.

The second unique feature in *SuperStor* is something called Recompress. As the name suggests, this provides the ability to take files already located on a *SuperStor* drive, and shrink them further. The process must be performed manually, and the files lose their extra compression factor when you write to them (sometimes incrementally, as with database files), but if you have a collection of files that are seldom changed, Recompress brings *SuperStor*'s compression ratio from a second place finish to the leader position.

Operationally, *SuperStor* and *Stacker* are virtually identical. Once you've created *Stacker*'s stacked volumes or *SuperStor*'s container files, they take on all the characteristics of any other DOS-accessible drive. The two products can even coexist. Both products consist of several hundred kilobytes of files, only a fraction of which are critical to operation, and both can use compressed volumes as large as 256 megabytes. Memory usage is similar, too, but *SuperStor* is the leader here, based on flexibility. While you can load the entire product into upper memory on 80386-and above systems, *SuperStor* lets you break the software into components, so users with limited upper memory available can still

save part of their conventional memory. All told, *SuperStor* needs about 50K RAM to operate with one container file available and the automount floppy recognition feature enabled. This figure is raised or lowered incrementally by the features you select. To load all of *SuperStor* into high RAM, you'll need a contiguous 64K block at start up.

Like *Stacker*, *SuperStor* now includes a specialized defragmenting utility, and it's much faster than Stac's product. Also like *Stacker*, there's a driver that changes the names of your drives so you needn't rewrite batch files or alter the way you think of your system. Stay away; we consider the need to think of your C: drive as E: (or whatever) as the necessary trade off for what products like *SuperStor* do for you.

Documentation is clear, but focuses exclusively on the "hows", at the expense of information on the "whys". Since *SuperStor* can address the same container file under multiple drive letter aliases simultaneously, we consider this too dangerous a practice to forgive. Additionally, the installation process makes changes to your configuration files—particularly if it finds a memory manager—that may not be to your liking, and places the main *SuperStor* driver in the root directory of your C: drive, where it doesn't belong. As with *Stacker*, our advice is to decline all offers of automatic assistance, and work through the tuning of *SuperStor* by hand.

The point is this: *Stacker* performs a little better under general conditions, but *SuperStor* has some special tricks that might make it a better choice. If you are using Novell's *DR DOS* and got *SuperStor* free, there's certainly no reason to look at anything else for your disk compression needs.

SuperStor 2.0  
AddStor Inc.  
3905 Bohannon Drive  
Menlo Park CA 94025  
800-732-3133  
\$139