



A monolithic app... and tickle

Chris Bidmead doesn't approve of monolithic applications but shows that the Linux-FT package's Xadmin, written in Tcl, is the exception. He has some fun with WPS objects and reviews the latest Linux support issues, too.

In the past, I've whinged not just about the shortage of OS/2 applications, but about the difficulty of getting hold of those few that are out there. It seems only fair, then, to report the good news that a UK company has heard my plea and is now delivering a product for review.

I wish I could get a little more excited about the first two packages that One Stop Software (talk to geoffd@cix.compulink.co.uk) has sent me so far, but that's hardly their fault. In any case, I've never believed in the "killer app" scenario that says OS/2 needs to come out with an application that everybody will clamour after, and to which no other platform can do justice. In my humble opinion, OS/2 is the killer app — a somewhat brash statement, the truth of which will emerge (or not) over the coming year when we see (or don't see) wondrous things happening with OpenDoc.

Actually, I'm being a little unfair to Partition Magic and Impos/2, because they do what they do rather well. Impos/2 is a modest but subtle bitmap manipulation package for creating images or touching up pictures. It's hardly going to make a dent in the sales of Windows products like Picture Publisher, but for OS/2 users it's fully 32-bit (although it uses 16-bit stuff to read Kodak PhotoCD images) and integrates with the WPS through drag 'n drop. It comes with a comprehensive manual translated from the original German and is, in places, almost totally impenetrable.

Partition Magic is a utility for expanding,



Partition Magic is just the utility you need if, say, you decide you want OS/2's Boot Manager on a disk that's already fully partitioned: just shrink one of the partitions by 1Mb to make room. Unfortunately, once it's on, Boot Manager doesn't take kindly to subsequent movements of partitions, which is just one more reason I'm putting off using the thing for real

contracting and moving partitions. It can also convert FAT into HPFS on the fly. I'm proceeding cautiously with it because it assumes a world of OS/2 and DOS, and seems to know nothing about NTFS ext2 Linux partitions and NeXTStep, and most of my machines carry a veritable tutti-frutti of different operating systems. If I take the bold step, I'll let you know what happens next month.

More Fun with WPS Objects

I like OS/2's WPS for the neat, simple things you can do with it. Here's something I knocked up the other day that may be of some use to you.

I've said before that it's not smart to work directly from the icon of an executable file in its home directory. Instead, create a program object and put that in a folder, then work from a shadow of that object which you keep somewhere convenient. Or work with data objects that have been connected to appropriate program objects via SOM, using Type in the data object and Associate in the program object. (If the WPS doesn't provide a ready-made type that fits the bill, make one up and write it in. I prefix all my new types with CHB so I can recognise them.) The advantage of a data object is that like a folder it has a menu setting with which you can mess around.

If you're like me there will be a number of icons floating around on your desktop that belong in groups. Maybe one icon is for editing a batch program and the other

for running it. Or one icon launches a comms package like Ameol and another inspects the download directory. The WPS offers several ways of dealing with this. An obvious one would be to put related icons into the same drawer of the Launchpad. I've already described another in this column: create a folder for Ameol and put the Ameol launcher icon, and a shadow of the download directory into it. Then — and here comes the neat bit — drop the Ameol

launcher icon into the Open settings of the folder, and make it the default action. Round the whole thing off by giving the folder the Ameol icon. Thus, the download directory is effectively hidden inside Ameol. But you can always get to it through the Open menu.

Getting to Verity's Topic

I get to my text retrieval package, Verity's Topic, through a pair of program icons: one for evoking the program, and another for doing its indexing and general maintenance from the command line. Why would you need a program object to launch a command line utility? Well, the program object actually opens a tailored OS/2 window, with its own distinctive Window Title and background colour and a custom built PATH that points to the nest of special Topic directories where I keep my Topic

binaries and batch files.

When I began to take the Launcher seriously, I realised it would be nice to put the pair of objects into a single icon. The Ameol Folder solution came to mind. I run Topic more often than I maintain it, so the logical thing would be to make running Topic the default double-click action, with opening the maintenance environment as a secondary action accessed through the menu. Unfortunately, a folder has Icon View, Tree View and Details View as irremovable menu items, and all I really wanted was the choice of Run or Maintain.

The solution was to use a data object instead of a folder object. A data object doesn't have to have data: it can be empty and merely used as a launcher. Drag a new object off the Data object template, open the Settings and remove Plain Text from the Type window. (When you close the Settings again this will remove the unwanted System Editor from the Menu options.) Then put the Topic program object into the Open menu as default, and the Maintenance program object as the other Open menu option. Stick a fancy icon on the front of the data object, and lo and behold, you have a program object that knows two tricks instead of just one. You could use this technique to combine ZIP and UNZIP, or give yourself a choice of text editors. I'm beginning to think that the rule should be to drive everything through WPS data objects.

Linux: the nightmare continues...

...But not for me, I hasten to add: I'm very happy with the Linux-FT distribution that Ian Nandhra, of Lasermoon, sent me for review and you'll be reading about it soon.

But last month, I mentioned the problems Ian was having with the free support he offers. "Linux Tech Support is a nightmare," he says. "There is a hard core of non manual-reading users who don't want to think for themselves, and demand you drop everything until they're sorted. It only takes three of those ringing you up on the same day to completely wipe you out — at which point everything else gets stacked up. The people who call up have paid virtually nothing, consider they are entitled to free support, and have no reason to moderate their consumption of this free thing. We manage to stay sane (ish) here by confining tech support to the hours between 2.00 and 6.00 in the afternoon."

Ian has had to do this because he's seen the effects on other vendors of what he calls "Linux Tech Support Stress". He tells me: "There's an outfit in the US who will remain nameless, because the people

concerned are very good friends of mine: one of them is in hospital, and the other only manages to recover at home, at the end of the day, with the aid of five or six bottles of beer."

So the utterly great idea of distributing a major operating system for free is under serious threat. Ian calls the technical support issue "The biggest Achilles' heel in the free software revolution". In theory, you download the software for nothing over the Internet, and turn to the Internet for support when you need it. But because your modem isn't fast enough and you don't have the expertise to install raw Linux, you probably get it as a ready-to-go distribution from a CD-ROM distributor instead. Money changes hands. That's perfectly within the spirit of the GNU licence, but it tends to change the expectations you have about support. As will your experience of the Internet if you're a raw beginner. As Ian says: "What you get on the Internet is not tech support, it's opinion. Some of it happens to be good, and some is awful. The average Joe has no way of telling the difference."

Linux was developed by, and for, people who care about Unix. They are prepared to invest time in it, and that includes sorting the genuine information from the junk that gets handed out over the Net. But Linux has grown up since these ideas were formulated. A modern distribution like Linux-FT is pretty well as ready to run as the copy of Windows NT 3.51 I installed this afternoon, and is ending up in the hands of the same kind of people who buy shrink-wrapped operating systems. These people will almost certainly need outside help at some stage, and that means somebody's time being devoted to their problems. Time is money. The bits may be free, but the bods always cost. The present ambiguities of the "free software" idea are leaving some users frustrated, and turning distributors into screaming wrecks.

It can't go on like this. Linux is in danger of collapsing under the weight of its own success. Or Caldera could save it by eventually replacing all other Linux distributions (I doubt this). Neither of these are welcome scenarios in my opinion.

A new approach

One way around this is the robust yahoo-sucks approach taken by software developers like Peter Childs (pjchilds@apanix.apana.org.au). He's written a utility that converts OS/2 .INF help files to HTML, either as a one-off effort or dynamically so that a Web server can serve up .INF files directly. He gives it away free as

