The NeXT works with the Wang DAT right out of the box. I was able to test and verify this for myself. The Wang DAT costs about $1,500 - $2,000 US, and each 1.3 gigabyte cartridge costs a mere $20 US, as opposed to $150 per 256M for an optical platter.

To use the Wang DAT, plug into SCSI, turn on power, then power on your NeXT. Go to Terminal, and type:

```
# tar cvf /dev/nrxt0 /
```

This backs up your entire hard disk to the DAT. You can also use dump or dd. The device mapped to is /dev/nrxt0 and /dev/rxt0. The difference is /dev/rxt0 will rewind the tape when the device is closed, and /dev/nrxt0 won't.

To manually rewind the tape, use this command:

```
# mt rewind
```

Read the manual pages on mt and rxt for the nitty gritty details. You will note the Wang DAT is very fast on rewinds. The tape can rewind from end to start in less than a minute. The cartridge is also very "cute"; tiny, light, about half the size of a regular cassette tape.

**Software Distribution, NeXT Style**

Being a developer, one really appreciates the NeXT for its binary compatibility. Furthermore, all NeXT have some very good "lowest common denominator" support: PostScript on every box, automatic color/mono support (write your code one way, for color, and it works on all the machines), standard fax support, a minimum 17" of display real estate, etc.

If NeXT should introduce a RISC box, I can see that almost all software companies will start to bundle both the 68040 and the RISC version on the distribution disk. The compiler will also have the ability to generate native code for both 68040 and RISC, so one can do cross development on any machine in the family. NeXT will even take this one step further, by allowing both binaries to be bound into one file. When executed, the operating system will automatically branch to the correct binary. This way, small programs and installation programs can run on both machines. Larger programmers will want to be split into two binaries, and only one of which will be copied to the hard disk during installation.

One eye witness claims to have seen NextStep running on HP's new killer workstation, the HP-9000 series 700 Snake.

Will we see NextStep on DecStation or the Snake? It's hard to say. NeXT is naturally wary of what happened with IBM's licensing of NextStep, i.e. nothing. IBM did very little to promote NextStep, understandable since IBM wants to deliver an uniform message to its customers: X-windows is the IBM way. DEC and HP, both OSF members like IBM, are obligated and committed to back OSF-1, Motif, and X-windows. Even if NextStep is licensed to either vendor, we may not see very much result.

More interesting is the prospect of NeXT adapting the MIPS R6000 64-bit, 50 MIPS monster RISC chip, or the even more spectacular HP PA-RISC. If the benchmark numbers HP engineers toss around are any indication, such a computer will be about five times faster than the 68040. Even the biggest, slowest NeXT applications will fly with that kind of muscle.

**NeXT's New Strategy: Connectivity**

The history of NeXT strategy can be divided into roughly four parts: first, in the starting days, it was "academic computing". After that was poorly received by the press, it became "best desktop publishing computer". Money didn't start tumbling in, so the battle cry turned to "awesome spreadsheet machine with Improv". Now that Improv has shipped, and though impressive, is ridiculously missing a scripting language. Knowing that skepticism will quickly follow from the pundits and the press, NeXT is now working on its next marketing strategy, connectivity.

To start with, Mac disk formats will be supported in System 3.0. As well, Steve Jobs has spoken about Novell and Apple EtherTalk connectivity. The plan is to directly convert EtherTalk packets to make Macs think NFS is an AppleShare server, and NeXT thinks AppleShare is NFS. Ditto with Novell: you'll be able to run Novell applications right off of a NeXT with NFS, and NeXT apps can reside on Novell servers.

SLIP or DialUp IP will finally become a part of System 3.0 as supported software. Now that the brain dead serial port of the old motherboard has been fixed, we can finally do network bridging with a dialup line.

**Cheap Backup Alternative**

Many of the new converts into the NeXT world will have bought the popular NextStations, which has a floppy drive, but no optical drive. The OD drive has a warm place in my heart as an awesome backup device, but I realize the OD glory days are past. NextStation owners cannot use OD, and Cube owners may be reluctant to shell out $3K for an OD drive, especially if they only know about the DAT alternative. Yes, the DAT is the best way to backup your NeXT, and the cheapest too.

of how things should be, so Sybase still consults with /etc/host to find out the IP address of the host name.

Surprisingly, the isql program, and in fact any program compiled with the DB-Library routine, consults NetInfo and not /etc/host for the IP address of host names, unlike the old fashioned server. So on machine "beta", you can need to make sure "beta"'s NetInfo manager is aware of machine "alpha". Go to NetManager, Open a host, and see if alpha is already in the table. if not, create an entry with New, then save with Cmd-S.

If you've followed all this so far, you can now be rewarded with the wonders of having true network based SQL server in action. Run isql -Usa -P from "beta", and you will find all your Sybase requests have been routed over to alpha.

So in summary, here are the steps:

(on machine alpha, using csh)
# setenv SYBASE /usr/sybase
# setenv DSLISTEN setup1
(now edit /usr/sybase/interfaces to add setup1, then edit /etc/host)
# /usr/sybase/bin/dataserver -d/usr/sybase/database/d_master &

(on machine beta, using csh)
# setenv SYBASE /usr/sybase
# setenv DSQUERY setup1
(now edit /usr/sybase/interfaces to add setup1, then run NetManager and add "alpha")
# /usr/sybase/bin/isql -Usa -P

I've found the performance for network based server to be exceptionally good. When I run both my client and server on an 8M NeXT, there was quite a bit of paging activity. But with the load thus distributed, updates and selects were lightning fast.

**NeXT RISC Rumors**

Just before NeXT's September 1990 announcement, rumor was flying everywhere about a NeXT RISC. With the introduction of 68040, rumors were given to rest, and the whole topic fell out of fashion. But I have on good sources that NeXT is very close to deciding a RISC strategy.

NeXT has evaluated the latest and best RISC architectures out there, and have, according to sources, narrowed the choice down to two. No, not the Sparc, not the Motorla 88000, not even the IBM RS/6000. It's now down to MIPS or HP Precision Architecture (PA) RISC.

In fact, rumor has it that NeXT has gone through extensive negotiations with both DEC, who use MIPS, and HP, on licensing NextStep for their prospective UNIX workstations.

```
2> go
1> select * from authors
2> go
```

And you will see a screenful of author names, etc. "pubs" is a sample database that gets generated automatically when you ran CreateMaster. You should purchase or print the on-line Sybase user documentation and play with this database.

When you want to shut down the system, you can use the command /usr/sybase/install/stopserver. To start it up again, type

```
# /usr/sybase/bin/dataserver -d/usr/sybase/database/d_master &
```

Okay, now the really fun stuff. You can setup two NeXTs and have one act as the SQL server. Let's suppose you have two machines, one named alpha and the second named beta. You want to run the Sybase server on alpha, and use beta to access the database. You will want to start by defining the environment variable SYBASE to be /usr/sybase. This tells Sybase where to look for the file $(SYBASE)/interfaces. This is the file that controls network-based database access.

Sybase looks at another environment variable, DSLISTEN, to find out what port the Sybase server is listening to. So before you start your server on "alpha", you should set this variable to something like "setup1". The name is entirely arbitrary. Now modify the /usr/sybase/interfaces file on machine "alpha"; add the following text:

```
setup1
    master tcp next-ether alpha 3696
    query tcp next-ether alpha 3696
    console tcp next-ether alpha 4696
    debug tcp next-ether alpha 5696
    trace tcp next-ether alpha 6696
```

Now go to machine "beta" and add the same text to the end of /usr/sybase/interfaces. Set the environment variable DSQUERY to "setup1".

Here comes the tricky part. Before you start up the server on machine "alpha", you must edit /etc/host on machine "alpha" to have the Internet number for host "alpha":

In my case, I've configured alpha to the Internet number (IP address) of 192.42.172.10, so I type into /etc/host

```
192.42.172.10    alpha
```

There are a whole bunch of other entries in there, and they can all be deleted. This file is in fact never consulted by NeXT OS if NetInfo is running; NetInfo is the network global database that is now used instead of /etc/host. However, Sybase has an antiquated notion

would hang permanently.

It’s really not the OS vendor’s fault. They have to work with so many non-standard parts that may be added to a non-standard 386 clone. The same goes for the peripheral card vendors. My PowerPort multi-serial card won’t work on one 386 clone, but will on another, whose bus timing was more conventional.

Today the Interactive UNIX crashed with a kernel panic message: "Unrecoverable disk I/O error at sector=053323 on controller=0" or something like that. I don’t think I have any more hair left to pull out.

So after all the sweat, hair pulled by ends, tears, and cursing, I’ve just decided to give it all up for the good of my mental health. The ease of installation and use on the NeXT is something you don’t see in the price tag, and you never appreciate until you try to install your very own 386 UNIX system.

**Sybase on Release 2.0**

As most of you have already noticed, Sybase is not with Release 2.0. However, if you have the old Release 1.0, you can still get Sybase up and running under Release 2.0. This is how you do it:

1. Copy /usr/sybase from your old Release 1.0 OD to your Release 2.0 hard disk
2. Go to /NextLibrary/Packages/ThirdParty and launch Sybase_Db-library.pkg
3. Click on the Expand button and watch the fireworks. When asked to overwrite old files, click Continue.

Okay, that’s the first phase. Now you can create your very first Sybase database. The procedure was outlined rather vaguely in an obscure page in the old System 1.0 doc, and I can’t even find any thing about Sybase installation in System 2.0 docs. This is what you need to do: bring up Terminal, and type the following commands:

```
# cd /usr/sybase/scripts
# source SetVars
# CreateMaster
```

The process takes about five minutes to complete. You can verify that your system works by using isql, the Sybase command line SQL utility. Type:

```
# /usr/sybase/bin/isql -Usa -P
```

-Usa means User "sa" (Sybase Administrator) and -P means no password. That’s how it would be setup in the beginning. Try these two commands:

```
1> use pubs
```

**Tao #7**
**Feb 26, 1991**

It's been quite a while since I wrote Tao #6. I've taken some time off to devote myself to my fax project. Originally intended for NeXT, MirageFax was cancelled when NeXT announced fax software in System 2.0. To salvage the disasterous situation, I proceeded to rewrite the whole thing from scratch, to be used on other non-NeXT machines. Now the software runs on Xenix and UNIX 386, as well as a Sun SparcStation.

The R&D being complete, and the software being stable, I finally have some time on my hand to get back into more NeXT Programming and Tao Musing.

As usual, the standard disclaimers: almost everything you read here can be considered untrustworthy bar gossip. Tao is said to be a tabloid for good reasons. We'll run articles about a two headed baby NeXT, son of Steve Jobs clone and alien from Mars.

## Welcome to UNIX Hell

While programming UNIX workstations, I've come to fully appreciate just what a hopeless situation X windows under UNIX can be. The Sun SparcStation comes closest to being acceptable. Its high performance, excellent bus/video bandwidth (better than the NeXT, which really shows when you scroll large bitmaps). But the software base is just not there to make me productive. Office productivity tools can be counted on one hand. The good ones are incredibly expensive. I suspect 1991 will be called "The Year of Desktop UNIX", as will 1992, and 1993, and 1994. Something like the way it's been "The Year of Networking" since early 1980s.

Trying to make the lab 33Mhz 386 behave like a workstation is just hopeless. The video bandwidth is incredibly restricted, the screen so small and lacking resolution, the AT&T UNIX software so unwieldy, that I've simply given up the fight. Yes, you can start with a basic cheapo $2,000 386, and spend $8,000 in kludges and poorly supported extras to make it behave like a NeXT or a Sun SparcStation. But why bother?

At one time I had 8M of memory, running X windows, and Visix Looking Glass, a popular X desktop program. The disk swapping was incessant. Every window that came up required a good disk head thrashing. I used to complain about how 8M on the NeXT was slow... until today, I never realized how good I had it.

The installation for Interactive AT&T UNIX alone took hours. I've been through the installation process many times, so I made no mistakes. Still, one must be extremely well prepared; at one point, I had to shutdown the installation process and pull apart the machine just to check out the ethernet card IRQ and I/O addresses.

I won't even mention the horrors I've had trying to make an older version of Interactive UNIX run with my Adaptec SCSI controller. Suffice it to say that one boot out of three