



Virtualizing Sybase ASE with VMware

By Jason L. Froebe

Start with realistic expectations

Virtualizing Sybase's flagship product, Adaptive Server Enterprise, is quite easy and performs quite well... that is, if your expectations are realistic. You won't get the same performance as you would if you ran ASE on the native hardware (without virtualization). So let's look at what you can achieve.

What you will get:

- ◆ Portability
- ◆ Snapshots
- ◆ Disaster Recovery

Moving a virtual machine from one host to another is as simple as copying the files. No configuration changes, license files, or anything. Just copy the files, point the VMware console to the files and start the virtual machine. That's it. Really!

Often a DBA will apply a new ASE patch or upgrade an application. In the past, if the DBA had to back it out, he or she would have to reload all the databases and restore the Sybase directory. In some cases, it was necessary to reload the system databases.

Using snapshots allows for a wholesale failsafe restoration of the virtual machine. Snapshots are quite often used in lieu of database or Operating System backups.

Let's get started

On our test server we have a quad core Intel processor, 8GB RAM, 1TB RAID 5 and quad gigabit network controllers. As the machine was capable of running 64-bit

code, we installed Windows 2003 R2 64-bit for the host operating system instead of Linux because VMware performs better with Windows as a host Operating System. Preferably we would want to run VMware ESX but it did not support our hardware.

In the host machine's BIOS, we need to enable the VT option. This allows for hardware support of virtualization. Work with your IT department and hardware vendor to optimize the other BIOS settings.

Don't swap out memory

Once you've downloaded and installed the VMware Server software, you will want to tell it to not swap out any memory of your virtual machine. This is done in the **VMware console -> Host -> Settings** menu. Select "Fit all virtual machine memory into reserved host RAM". (see Figure 1) We'll allocate 3.5 GB of RAM to the virtual machine after we create it.

Building the virtual machine

When creating the virtual machine, you need to create the disk files as pre-allocated and split into 2GB files for performance. If you create sparse files instead, VMware will spend a great deal of time allocating more space from the host machine a bit at a time which will also lead to fragmentation of the disk files. (see Figure 2)



Jason L. Froebe is a Senior DBA at Sungard Investment Systems, a member of TeamSybase, an ISUG Board Director, the editor of My Databases magazine, a Perl Monger, "Go to TechGuy" at Froebe Fibers and can speak fluent munchkin. He can be reached at jason@froebe.net

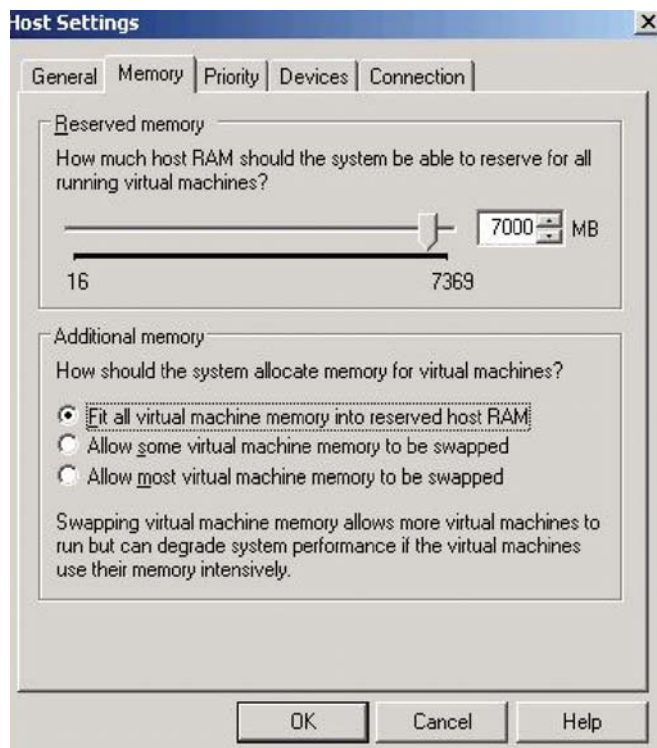


Figure 1

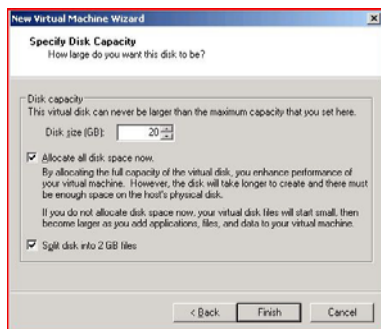


Figure 2

Once you click “Finish”, go to your local coffee shop for a siesta. It will take a while.

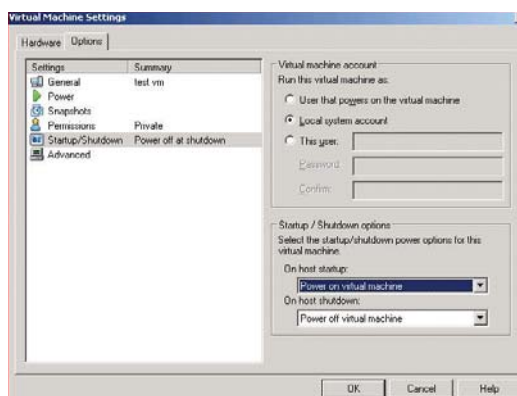


Installing the virtual machine OS

Within the VMware console, we pointed the virtual CD-ROM drive to our CentOS 5.2 DVD image and performed a default server install.

Virtual machine auto start

In the VMware console we modify the VM to power up the virtual machine when the host OS is up.



Adaptive Server Enterprise

We installed ASE 15.0.2 using the EXT3 filesystem for all database device files except for tempdb. For the tempdb device we used the EXT2 filesystem.

It seems that locking ASE's shared memory into physical RAM can be a bit tricky. Simply specifying “glock shared memory” in ASE (sp_configure) isn't enough.

In many distributions, you can simply add the line into `/etc/security/limits.conf` to enable locking of shared memory up to 2GB for the sybase user:

```
sybase - memlock 2147483648
```

Unfortunately this doesn't always work so you need to specify the memlock limit in the `$$SYBASE/$SYBASE_ASE/install/RUN_server` file by adding the following above the dataserver line:

```
ulimit - l 2147483648
```

Whether you are able to lock the memory or not will depend on your environment (you do have enough free memory right?). You can check if the memory is locked by simply checking the `/proc/<pid of a dataserver process>/status`:

```
grep VmLck /proc/<pid>/status
```

```
VmLck: 1140000 kB
```

“Moving a virtual machine from one host to another is as simple as copying the files. No configuration changes, license files, or anything. Just copy the files, point the VMware console to the files and start the virtual machine. That’s it. Really!”

Keep in mind that you won’t have a 1 to 1 memory size match between the locked memory (VmLck) and the total memory VmSize allocated simply because Sybase ASE only locks the shared memory and not the process.

Using the noop scheduler enables the host operating system to better optimize I/O resource usage between different virtual machines. This can be added by adding this line to the kernel boot parameters in `/boot/grub/menu.lst`:

```
elevator=noop
```

Set the read ahead value on the blockdevices (added to `/etc/rc.d/rc.local`) from 256 to 16384:

```
blockdev --setra 16384 /dev/sda
blockdev --setra 16384 /dev/sdb
blockdev --setra 16384 /dev/sdc
```

Lessons Learned

- ◆ While virtualizing a DBMS sounds great, it may not be practical in your environment
- ◆ For QA or development servers, the quick reverting to an original environment can save hours or days for developers/testers
- ◆ ASE can saturate the CPU when inefficient queries are issued. You will find that some queries that you thought were optimized really aren’t. This will become a boon for your production environment as you optimize the queries
- ◆ Locking the VM and ASE into physical RAM is an absolute must
- ◆ Enabling VT BIOS option can give a 5% boost in performance

The following virtual memory & I/O subsystem tunables (`/etc/sysctl.conf`) are useful, they help significantly with I/O performance:

```
vm.swappiness = 0
vm.overcommit_memory = 1
vm.dirty_background_ratio = 5
vm.dirty_ratio = 10
vm.dirty_expire_centisecs = 1000
dev.rtc.max-user-freq = 1024
```

Disabling the kernel from over committing memory and only using swap when physical memory has been exhausted helps overall performance (vm.swappiness). The maximum user frequency covers how fast a virtual machine can set its tick count to. The vm.dirty options tune how the VM subsystem commits I/O operations to disk.

Backup server

While ASE performs quite well under virtualization, ASE’s backup server does not due to the excessive small I/O read/writes. We will use VMware’s snapshot capability to perform virtual machine image backups instead.

If you do need to use ASE’s backup server, backup to a remote folder with no compression else it will monopolize the virtual CPUs.

Software Used

- ◆ Sybase Adaptive Server Enterprise 15.0.2 ESD 3 for Linux x86 (32-bit)
- ◆ Microsoft Windows 2003 R2 x86-64 (64-bit)
- ◆ CentOS Linux 5.2 x86 (32-bit)
- ◆ VMware Server 1.0.5 for Windows x86-64 (64-bit)

Jason L. Froebe’s links:

WebBlog <http://jfroebe.livejournal.com>

Tech log <http://www.froebe.net/blog>