

# **Spartan '10 for Linux Installation Instructions**

**version 1.1**

**Section 1: Installing Spartan '10 from DVD-ROM**

**Section 2: Installing the Spartan Molecular Database from DVD-ROM**

**Section 3: Accepting job requests from a Linux client**

**Section 4: Configuring a Linux Client for remote job submittal**

**Section 5: Accepting job requests from a Windows client**

**Section 6: Configuring a Windows Client for remote job submittal**

**Section 7: Files created during installation**

**Section 8: Removing Spartan**

## Section 1: Installing Spartan '10 from DVD-ROM

Insert and mount the DVD-ROM.

As root user perform the following steps.

```
cd /media/cdrom
sh spinstall.sh
```

There are two versions of Spartan'10 for Linux: a 32-bit version supporting Intel Pentium 4, Core, Xeon and AMD Athlon, Phenom and Opteron; a 64-bit version supporting EMT64 (newer versions of Intel Pentium 4, Core and Xeon) and AMD64 (Athlon64, Phenom and Opteron). The installation program will unpack and install the appropriate version for your system.

The installer will prompt for the directory where Spartan is to be installed. This allows flexibility in placing the Spartan executables in a directory with sufficient disk space. This also allows the Spartan executables to be installed on a network storage server. Spartan can then be installed on additional systems that have access to this network storage by running the following abbreviated installation...

32-bit:

```
cd /usr/spartan10_v1.1_x86
```

64-bit:

```
cd /usr/spartan10_v1.1_x86-64
```

(“/usr” is the default, substitute the directory chosen during the installation)

```
./sp_install
```

This installation method skips the unpacking of the Spartan executables and simply configures Spartan for the local system.

After the installer has unpacked Spartan onto the system, a series of questions will be asked to configure Spartan.

### **Queue Type?**

There are two choices for queuing, internal and external. Select internal to use Spartan's integrated queue. This is a simple FIFO (First In First Out) queue, and is useful on standalone systems that don't have other queuing capabilities installed. If a queue is available (e.g. OpenPBS, Torque, SGE, LSF), select external. This will result in Spartan submitting jobs to a queue that is external to Spartan.

If internal is chosen, the “*Maximum number of running jobs*” will be prompted for. This

number defaults to the number of processors/cores in the system.

### **Remote Job Storage Location?**

This is the directory where jobs submitted from remote systems will be stored. Typically this directory should be world-writable with the sticky bit set. The sticky bit should be set such that only the owner of a job is allowed to delete or rename the job. The default choice is “/tmp” which is world-writable with the sticky bit set.

### **Scratch Storage Location?**

Some higher level Spartan jobs require the use of temporary disk space. Enter a directory where sufficient disk space is available. Typically this directory should be world-writable with the sticky bit set. The sticky bit should be set such that only the owner of a scratch file is allowed to delete or rename it. The default choice is “/tmp” which is world-writable with the sticky bit set.

### **Scratch limit per job?**

Specify the amount of disk space available per job. Take into account the number of running jobs allowed by the queue. The minimum recommended amount of disk space per job is 20000MB (20GB). Larger amounts of disk space will help with higher levels of theory (MP2,MP3...).

### **Memory limit per job?**

Specify the amount of physical memory (RAM) available per job. Take into account the number of running jobs allowed by the queue. The installer will attempt to determine the amount of memory and apply a formula to calculate the amount of memory per job. The minimum amount of memory required per job for good performance is 512MB. The recommended amount of memory per job is 1024MB.

### **License Type?**

Three ways to license Spartan are provided.

**USB HASP/dongle (blue)**, The USB HASP plugs into a USB port on the system. A driver/run-time will need to be loaded. The installer for the USB HASP driver/run-time can be found in the Spartan installation directory.

32-bit:

```
cd /usr/spartan10_v1.1_x86/hasp
```

64-bit:

```
cd /usr/spartan10_v1.1_x86-64/hasp
```

(“/usr” is the default, substitute the directory chosen during the installation)

View the README file for further instructions.

**USB NetHASP/dongle (red)**, The USB NetHASP plugs into a USB port on a central server. A driver/run-time will need to be installed on the server. In addition, a driver/run-time will need to be installed on each system that the user wishes to run Spartan.

32-bit:

```
cd /usr/spartan10_v1.1_x86/hasp
```

64-bit:

```
cd /usr/spartan10_v1.1_x86-64/hasp
```

(“/usr” is the default, substitute the directory chosen during the installation)

View the README file for further instructions.

**Activation code**, an alternate way to license Spartan is with a web activation key. An activation key is entered when prompted and the installer automatically contacts Wavefunction's license server and completes activation of Spartan.

## **Section 2: Installing the Spartan Database from DVD-ROM**

Insert and mount the Spartan Molecular Database DVD-ROM.

As root user perform the following steps.

```
cd /media/cdrom  
./spinstalldb.sh
```

The database requires ~2.0GB of storage so be sure adequate disk space is available.

### **Section 3. Accepting job requests from a Linux client**

Spartan '10 makes use of *ssh* and *scp* to accept job requests from a remote Linux client. To make use of *ssh* and *scp*, Spartan requires the ssh server allow the client to have access without prompting for a password or passphrase. This can be arranged in several ways. The easiest is to use public key authentication. The instructions on how to setup public key authentication follow...

On the Linux client...

```
ssh-keygen -t dsa -N '' -f $HOME/.ssh/id_dsa
```

This command will generate two files. The file “\$HOME/.ssh/id\_dsa” is the private key and the file “\$HOME/.ssh/id\_dsa.pub” is the public key.

Copy the public key file to the server (replace *server* with the name of your server).

```
scp $HOME/.ssh/id_dsa.pub server
```

On the Linux server...

```
mkdir $HOME/.ssh  
cat $HOME/id_dsa.pub >>$HOME/.ssh/authorized_keys2
```

Test the configuration from the Linux client...

```
ssh server
```

You should not be prompted for a password or passphrase, and should simply be presented with a shell prompt.

### **Section 4. Configuring a Linux Client for remote job submittal**

By default, Spartan submits jobs to the local system. To configure Spartan to submit to a remote system, access the *Preferences* dialog found under the *Options* menu in the Spartan Graphical User Interface. Now select the *Hosts* tab. To add a new host select the *New* button. Existing entries can be managed by selecting the *Edit* or the *Delete* buttons.

## Section 5. Accepting job requests from a Windows client

Spartan '10 makes use of the Samba Server to accept job requests from Windows PC clients. Two resources are needed to accept jobs. A file share and a pseudo printer.

When a PC client submits a job to the server, it first copies a Spartan Document to a file share on the server. Then the client sends a print request to the server. The print request is a file that contains the path to the Spartan Document on the server. The pseudo printer launches Spartan which processes the Spartan Document performing any requested analysis on the molecules contained in the Spartan Document. When the job is finished the PC client copies the Spartan Document back to the PC and the user is prompted with a job completion message.

A file-share and pseudo-printer are configured by editing the samba configuration file...

```
/etc/samba/smb.conf
```

Adding the following section to the samba configuration file will make the directory “/spartan\_share” available to users *andy* and *phil* on hosts *volvo* and *ferrari*.

```
[Spartan Share]
  comment = Spartan Share
  path = /spartan_share
  valid users = andy phil
  read only = No
  hosts allow = volvo.wavefun.com ferrari.wavefun.com
```

Typically the directory “/spartan\_share” should be world-writable with the sticky bit set. The directory can be created as follows...

```
mkdir --mode 1777 /spartan_share
```

Also, users *andy* and *phil* will need an entry in the `/etc/samba/smbpasswd` file. The SMB password used for each user must match their password on the Windows client. A password entry can be added by the root user by running...

```
smbpasswd -a andy
```

After the entry is added, users can update their SMB password by running...

```
smbpasswd
```

Test the file-share by copying a file from a Windows client to the file-share. The file should be copied without requesting a password.

Next, add the following to the samba configuration file to enable a pseudo-printer. Uncomment the directive “`printing = sysv`” for Samba version 3.0 or later.

```
[spartan]
  comment = Spartan 10
  path = /var/spool/spartan10
  printing = sysv
  printable = yes
  print command = /var/spartan/spartan10_samba %s
  browseable = yes
  guest ok = no
```

Again, the directory “`/var/spool/spartan10`” should be world-writable with the sticky bit set. The directory can be created as follows...

```
mkdir --mode 1777 /var/spool/spartan10
```

The server is now configured to accept jobs from a Windows client. The next step is to configure Spartan on the Windows client to submit jobs to the server.

## Section 6. Configuring a Windows Client for remote job submittal

A dialog panel to configure Spartan '10 for Windows for remote submittal can be accessed by selecting *Preferences* from Spartan's *Options* pull-down menu, and then selecting the *Hosts* tab.

The Preferences/Hosts Panel allows for the addition, modification, and removal of servers to which jobs can be submitted. Select *New...* to add a server to the list. Five input fields will be presented. The first field is for descriptive purposes only, so enter something that describes the server (e.g. Name, Manufacturer, Processor Speed). The second field specifies a file share that the user can access on the server. This file share is where the Spartan molecule file is copied prior to starting a job. The third field specifies a pseudo printer which will be used to start the job on the server. The fourth field specifies the path of the file share as seen from the server. A fifth entry is activated only if the user is an administrator. Selecting *Yes* will make this entry available to all users on the Windows system. Below is a sample of the host definition dialog.

WinNewHostDialog

Steps to define a remote host.

1. Specify the name of the remote host as it will appear in the Spartan Submit dialogue.  
New Host
2. Specify the share path that points to the work directory on the remote host.  
\\Local\Path\To\Share\Directory
3. Specify the Spartan pseudoprinter defined on the remote host.  
\\PseudoPrinter\Spartan
4. The share path from step 2 refers to a work directory on the remote host.  
Specify the directory's local path on the remote host.  
/Remote/Path/To/Share/Directory
5. Make the host accessible to all users on this machine?  
 Yes  No

Cancel OK

Spartan is now configured to submit jobs to a remote server. The next time a job is submitted an additional panel will be presented which will allow the selection of the server to run the job on. For the case of a new unsaved molecule, the *Save As – Submit* dialog will have an additional menu titled *Submit To*.



## **Section 7. Files created during installation**

The installation program will create the “/var/spartan” directory to contain several configuration files. The files created are:

/var/spartan/spartan10	Spartan run script
/var/spartan/spartan10_samba	Spartan Samba run script
/var/spartan/spartan10_snapshot	Spartan monitor system status
/var/spartan/spartan10_ssh	Spartan Ssh run script
/var/spartan/remote_job_path	Path to store remote jobs
/var/spartan/queue_entries	Spartan internal queue
/var/spartan/queue_limits	Spartan internal queue attributes
/var/spartan/webkeys	Spartan activation codes
/var/spartan/qchemrc	QChem resource directory

The installation program will also create the symbolic link...

```
/usr/bin/spartan10 -> /var/spartan/spartan10
```

The link is created to conveniently place the Spartan run script in the search path of all users. This allows users to access Spartan without having to augment their search paths.

## **Section 8. Removing Spartan**

Spartan does not have an “un-install” script, but the removal of Spartan is simple.

NOTE: If the intention is to remove Spartan '10 and replace it with a newer maintenance release, skip the last two remove commands (rm) listed below. This will preserve the access codes and the settings entered during the previous installation.

32-bit:

```
rm -r /usr/spartan10_v1.1_x86
```

64-bit:

```
rm -r /usr/spartan10_v1.1_x86-64
```

(“usr” is the default, substitute the directory chosen during the installation)

```
rm -r /var/spartan10  
rm /usr/bin/spartan10
```