

VDI User Guide

- 1. Prerequisites
 - 1.1. Windows
 - 1.1.1. TurboVNC
 - 1.1.2. Strudel
 - 1.2. MacOS before Catalina
 - 1.2.1. TurboVNC
 - 1.2.2. Strudel
 - 1.3. MacOS Catalina
 - 1.3.1. TurboVNC
 - 1.3.2. Strudel
 - 1.4. Linux
 - 1.4.1. TurboVNC
 - 1.4.2. Strudel
- 2. Strudel Configuration
- 3. Using The Virtual Desktop
 - 3.1. Connecting
 - 3.2. Time Limits
 - 3.3. Logging Out
 - 3.4. Disconnecting
 - 3.5. Storage
 - 3.5.1. Transferring Files
 - 3.5.2. SFTP Transfer Node
 - 3.5.3. Disk Quotas
- 4. Environment Modules
 - 4.1. Python
 - 4.1.1. IPython & Jupyter
 - 4.1.2. Installing Packages
 - 4.1.3. Using Virtual Environments
 - 4.2. PBS
 - 4.2.1 Gadi transition
- 5. Software Activation
 - 5.1. MATLAB
 - 5.1.1. Login Named User
 - 5.1.2. Network Licensing
- 6. Troubleshooting
 - 6.1 Generating A Debug Log



Release notes

If you already use the virtual desktop environment, also check the release notes for any recent updates: <https://vdi.nci.org.au/news>

1. Prerequisites

To access the virtual desktop environment, you will first need to install TurboVNC and the Strudel desktop launcher application from the MASSIVE/CVL project on your local computer. The instructions for installing these can be found for each operating system platform below. If you do not have the necessary access rights to install this software, please ask your system administrator to assist with this process.



Upgrading

If you are upgrading a previously installed version of Strudel then we recommend that you uninstall both Strudel and TurboVNC first (and reboot if you are on Windows).



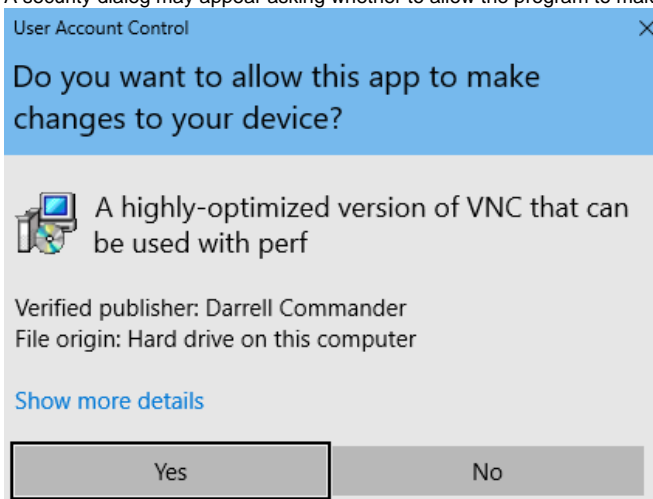
NCI account required

You will require an NCI login account to use a virtual desktop. If you don't already have one then you can register here: <https://my.nci.org.au>.

1.1. Windows

1.1.1. TurboVNC

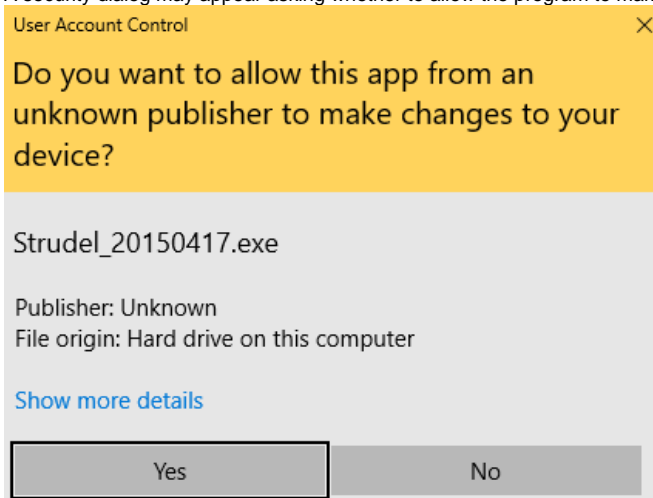
1. Download the TurboVNC installer:
64-bit Windows: <https://downloads.sourceforge.net/project/turbovnc/2.1.2/TurboVNC64-2.1.2.exe>
32-bit Windows: <https://downloads.sourceforge.net/project/turbovnc/2.1.2/TurboVNC-2.1.2.exe>
2. Double click the executable to start the installation.
3. A security dialog may appear asking whether to allow the program to make changes to the system. Click the "Yes" button to continue.



4. Continue through the installation wizard to install the application.

1.1.2. Strudel

1. Download the Strudel installer:
https://cvl.massive.org.au/launcher_files/stable/Strudel_20150417.exe
2. Double click the executable to start the installation.
3. A security dialog may appear asking whether to allow the program to make changes to the system. Click the "Yes" button to continue.

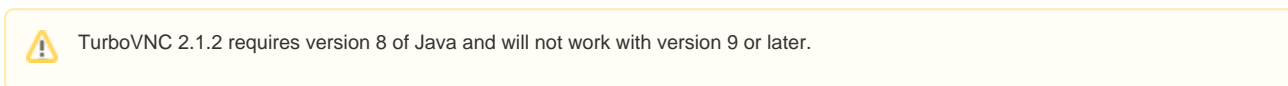


4. Continue through the installation wizard to install the application.

1.2. MacOS before Catalina

1.2.1. TurboVNC

1. Download the Oracle Java Version 8 runtime from here: <http://www.java.com/download/>

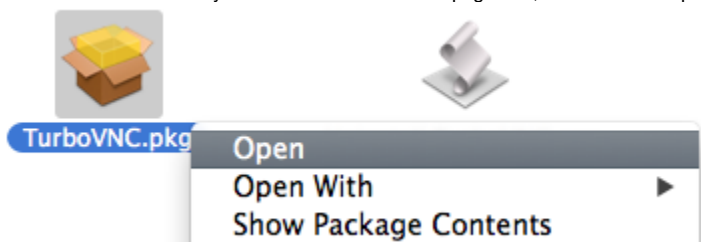


2. In Finder, double click the ".dmg" archive that you downloaded in the previous step.

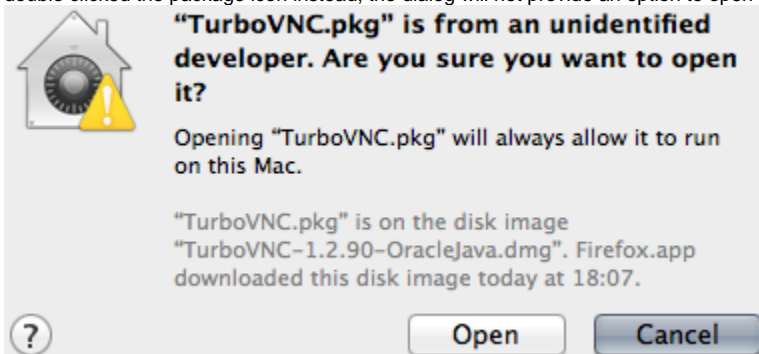
3. Double click on the package icon to start the installation wizard.



4. Click through the installation wizard to install the Java runtime.
5. Download the TurboVNC installation image:
<https://downloads.sourceforge.net/project/turbovnc/2.1.2/TurboVNC-2.1.2.dmg>
6. In Finder, double click the “.dmg” archive that you downloaded in the previous step.
7. Hold down the control key and click the “TurboVNC.pkg” icon, then choose “Open” from the menu.



8. A dialog may be displayed stating that the application is from an unidentified developer. Click the “Open” button to continue. Note that if you double clicked the package icon instead, the dialog will not provide an option to open the application.



9. Click through the installation wizard to install the application.
10. Open a terminal window and enter the following command to enable normal key repeat when holding down letter keys (eg. when navigating in Vim) rather than Apple’s accent character popup which doesn’t work with TurboVNC:

```
defaults write com.turbovnc.vncviewer.VncViewer \  
  ApplePressAndHoldEnabled -bool false
```

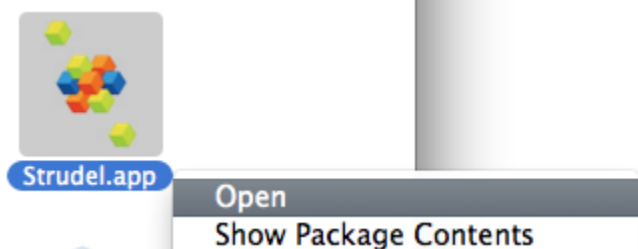
1.2.2. Strudel

1. Download the Strudel installation image:
https://cvi.massive.org.au/launcher_files/stable/Strudel_20150417.dmg
2. In Finder, double click the “.dmg” archive that you downloaded in the previous step.
3. Select the “Strudel.app” icon and drag it to the “Applications” folder.



4. Open the “Applications” folder in Finder.

5. Hold down the control key and click the "Strudel.app" icon, then choose "Open" from the menu.



6. A dialog will be displayed stating that the application is from an unidentified developer. Click the "Open" button to launch the application for the first time; you won't have to do it this way again as the system will now remember that you trust this application. Note that if you double clicked the application icon instead, the dialog will not provide an option to open the application.



1.3. MacOS Catalina

1.3.1. TurboVNC

Follow the [above](#) steps to install TurboVNC.

1.3.2. Strudel

1. Download and install this specific version of either [Anaconda](#) or [Miniconda](#). For the remainder of these steps, substitute "~/miniconda3" with the path to your local Anaconda or Miniconda installation.

 New version of conda does not support wxpython < 4.

2. Download [Git](#) and install it on MacOS if you do not have it.
3. Clone the Strudel source repository to your local machine: (make sure you checkout the specific version shown)

```
git clone https://github.com/monash-merc/cvl-fabric-launcher strudel
cd strudel
git checkout f40b5b6
git submodule update --init
```

4. Setup a new conda environment dedicated to running Strudel:

```
~/miniconda3/bin/conda create --name strudel python=2 appdirs boto cryptography lxml paramiko pexpect
psutil requests setuptools "wxpython<4.0"
source ~/miniconda3/bin/activate strudel
PYTHONNOUSERSITE=1 python create_commit_def.py
```

5. Create a small script which you can run whenever you want to launch Strudel:

```
cat >run.sh <<'EOF'
#!/bin/bash
set -eu
source ~/miniconda3/bin/activate strudel
PYTHONNOUSERSITE=1 pythonw "${0%*/}/*}/launcher.py"
EOF
chmod +x run.sh
```

6. Then start it for the first time:

```
./run.sh
```

1.4. Linux

1.4.1. TurboVNC

TurboVNC 2.0 and later require the Java runtime. For Ubuntu you can install this like so:

```
sudo apt-get update
sudo apt-get install openjdk-8-jre
```

On CentOS:

```
sudo yum install java-1.8.0-openjdk
```



TurboVNC 2.1.2 requires version 8 of Java and will not work with version 9 or later.

TurboVNC packages for Linux are available from the downloads area: <https://sourceforge.net/projects/turbovnc/files/2.1.2/>

1.4.2. Strudel

There are some prebuilt [Linux packages](#) of Strudel however many users have reported problems getting these to work. Therefore we recommend the following approach instead:

1. Download and install this specific version of either [Anaconda](#) or [Miniconda](#). For the remainder of these steps, substitute "~/miniconda3" with the path to your local Anaconda or Miniconda installation.



New version of conda does not support wxpython < 4.

2. Install the "git" package for your distro. For example on Ubuntu:

```
sudo apt-get install git
```

Or for CentOS:

```
sudo yum install git
```

3. Clone the Strudel source repository to your local machine: (make sure you checkout the specific version shown)

```
git clone https://github.com/monash-merc/cvl-fabric-launcher strudel
cd strudel
git checkout f40b5b6
git submodule update --init
```

4. Setup a new conda environment dedicated to running Strudel:

```
~/miniconda3/bin/conda create --name strudel python=2 appdirs boto cryptography lxml paramiko pexpect
psutil requests setuptools "wxpython<4.0"
source ~/miniconda3/bin/activate strudel
PYTHONNOUSERSITE=1 python create_commit_def.py
```

5. Create a small script which you can run whenever you want to launch Strudel:

```
cat >run.sh <<'EOF'
#!/bin/bash
set -eu
source ~/miniconda3/bin/activate strudel
PYTHONNOUSERSITE=1 python "${0%/*}/launcher.py"
EOF
chmod +x run.sh
```

6. Then start it for the first time:

```
./run.sh
```

2. Strudel Configuration

After you have installed Strudel, you need to add the site configuration for the NCI virtual desktops:

1. Start the Strudel desktop launcher.
 - Windows 8 or later: Search for Strudel in the Apps view of the Start screen.
 - Windows 7: You'll find Strudel in the Windows start menu.
 - macOS: Search for Strudel in Launchpad.
 - Linux (installed as above): `~/strudel/run.sh`
 - Linux (binary package): `/opt/Strudel/strudel.sh`
2. The first time that you use the application, you'll be automatically prompted to configure the site list. If you have used Strudel before, then choose "Manage Sites" from the "File" menu to open the site configuration dialog.
3. Click the "New" button to add a new site.
4. Enter the following details (copy and paste) then click "OK":
 - Name: NCI Virtual Desktops
 - URL: <https://vdi.nci.org.au/strudel.json>
5. Click "OK" to close the site configuration dialog.

3. Using The Virtual Desktop

3.1. Connecting

Whenever you want to connect to a virtual desktop:

1. Start the Strudel desktop launcher.
2. From the site drop down list in the main window, choose the NCI desktop pool that you wish to use.
3. Enter your NCI login name into the "Username" field.
4. Click the "Login" button to connect.
5. If this is the first time you have used Strudel:
 - You'll be prompted to confirm which authentication mode to use. Select "Don't remember me" if you are using a shared computer, one that is not under your personal or organisation's control, or if you are unsure either way.
 - Click "OK" to close the settings dialog.
 - If you have chosen to be remembered, you will now be prompted to create a passphrase to protect the SSH private key that will be generated and stored on your computer.
6. You may be prompted for your NCI account password and/or the SSH passphrase that you created above.
7. Once connected, the desktop will appear in a new VNC window. There may be a long delay before the desktop appears while the system waits for an available node.



Pool access

Access to each virtual desktop pool is restricted to certain NCI project groups. If you try to connect to a desktop pool that you don't have access to, then the following error message will be displayed:

```
User's group not permitted to use this partition
```

For further assistance please open a [virtual desktop support ticket](#).

The table below lists the available desktop pools and their current configuration. Note that under certain circumstances your desktop session may be running on a node shared with one or more other desktop sessions.

Pool	Configuration
------	---------------

NCI Virtual Desktop	8 vCPUs, 32GB RAM, 140GB "/local" Max session lifetime: 7 days Inactive session time limit: 24 hours
---------------------	--

3.2. Time Limits

There are two different time limits which apply to all virtual desktops:

- **Maximum Session Lifetime:** Specifies the maximum elapsed time that a virtual desktop is allocated to a user from when the desktop is first started. Disconnecting and reconnecting to the desktop has no effect on this limit. When you disconnect from or reconnect to the desktop, Strudel informs you how many hours there are remaining.
- **Inactivity Limit:** If the Strudel client is disconnected from the virtual desktop for longer than this time limit then the session will be terminated to allow the resources to be returned to the pool for the benefit of other users. Note that connecting for less than one hour may not be registered as activity.



Deletion of files

When the desktop session ends, including when a time limit is reached, any files on the local disk (eg. under "/local") are immediately deleted and any other unsaved work will be lost.

3.3. Logging Out

When you have finished with your virtual desktop, click on your name in the top right hand corner of the desktop and choose "Quit" from the menu. Alternatively, you can choose "Log Out" from the "System" menu. In either case, this will terminate all applications and end the desktop session.



Persistent storage

Make sure that you move any important files to an appropriate location under "/g/data" or your home directory before ending the session.

3.4. Disconnecting

If you just want to disconnect from the virtual desktop and leave it running so that you can return to it a short time later, then simply close the VNC window. If you temporarily lose connectivity to the desktop (eg. network dropout) then the effect is the same. When you connect to the same desktop pool again, Strudel will prompt you to either reconnect to the existing desktop or start a new one. If you choose the later, then your existing desktop session will be terminated first.



Saving work

Avoid leaving work unsaved when you disconnect from the desktop. Your desktop session will be terminated if it is left disconnected for too long, or for other reasons such as system maintenance.

3.5. Storage

Unless specifically noted in the table below, files stored on the virtual desktop node are not backed up. You should treat any local storage locations as temporary for the duration of the desktop session only.

Location	Notes
----------	-------

/home	<p>User home directories. Mounted across all desktop nodes. Backed up to an archival system. Disk usage quotas apply. Good place for source code.</p> <p>Don't use this location for high demand I/O workloads as performance will be poor and will impact on all other users.</p> <p>The following folders in the "Places" desktop menu are included:</p> <ul style="list-style-type: none"> • Home Folder • Desktop • Documents • Music • Pictures • Videos • Downloads <p>Note that "/home" on "raijin" is completely separate and cannot be accessed from the desktops.</p>
/g/data /g/data[1-3]	<p>The global Lustre filesystem which is also available on "raijin". Only a subset of volumes are accessible so please ask if you would like your project area to be made available. Primarily intended for persistent data outputs and shared software installations. More information can be found here: Files and Filesystems</p>
/local	<p>Fast storage on the local node primarily intended for intermediate working files. Neither persistent nor backed up. Each user's temporary directory ("\$TMPDIR") is set to this location. Does not involve any network traffic and is backed by an SSD on the node. Please take care not to fill up this volume as it will impact other users on the same node; currently we aren't applying quotas but may do so in future.</p>
/short	<p>Note that "/short" on "raijin" is completely separate to the VDI and cannot be accessed from the desktops.</p> <p>We recommend using either "/g/data" or "/local" (whichever is appropriate) instead of this area as it's currently under consideration to be removed. No new project allocations are being made here. If you have a specific need for this storage please contact us to discuss your requirements.</p> <p>Not backed up and not guaranteed to be persistent. Unlike "/local", this area is shared amongst all desktop nodes so you can exchange temporary working data with other colleagues if needed. However, each project area is limited to 100GB and files will be purged automatically after 30 days.</p> <p>Don't use this location for high demand I/O workloads as performance will be poor and will impact on all other users. For better performance you should use "/local" instead.</p>

3.5.1. Transferring Files

You can transfer files between the virtual desktop and your local machine using the following tools:

- rsync
- scp
- Any client that supports SFTP (see below)

The name of the desktop node is usually displayed in the VNC window title bar. You can also run the following command in a terminal window on the desktop:

```
hostname
```

You need to append ".nci.org.au" to the node name; eg. "vdi-n1.nci.org.au".



Data movers

You can use the raijin data movers ("r-dm.nci.org.au") to transfer data to/from "/g/data" rather than using VDI for this purpose.

3.5.2. SFTP Transfer Node

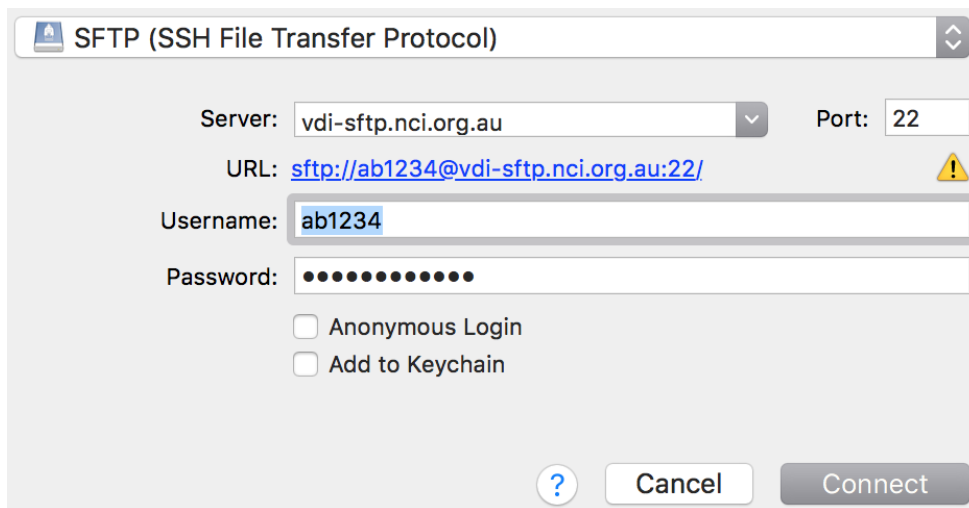
Your home directory can also be accessed using the SFTP transfer node. Accessing this service does not require that you have a running virtual desktop session and therefore allows you to correct problems that are preventing you from starting a new desktop session such as:

- Exceeding your home directory quota.
- Modifying your login scripts (eg. “.bashrc”) in such a way that prevents the desktop session from starting correctly.

To connect to this service you need to use an SFTP compatible client, for example:

- [Cyberduck](#) (Windows and macOS only)
- [FileZilla](#)
- The command line “sftp” client shipped with Linux and macOS.

Set the protocol to SFTP (SSH File Transfer Protocol) where applicable, and then connect to the server “[vdi-sftp.nci.org.au](#)”. Login with your usual NCI username and password. For example, in Cyberduck the connection dialog will look something like this:



Once connected, you can perform typical file operations such as uploading, downloading, renaming, and removing files, etc. Note however that you cannot open a shell prompt or run arbitrary commands.

3.5.3. Disk Quotas

Disk block usage and inode (number of files and directories) quota limits apply to your home directory. Only hard limits are used and therefore there is no grace period. Attempts to create new files or write additional data to existing files will fail with an error if you exceed your quota.



Using SFTP node to recover from quota exhaustion

If your desktop session ends while you have very little or no quota left, when you try to start a new desktop session it will fail with the following error:

```
Unable to create job script file.
[Errno 122] Disk quota exceeded
```

When this occurs, you can use the SFTP transfer node to reduce your quota usage by removing some files. Note that files in the desktop “Trash” folder will consume quota until the folder is emptied by right clicking the icon and choosing “Empty Trash”. When using the SFTP transfer node, the folder path is “.local/share/Trash” in your home directory (you may need to configure your SFTP client to show hidden files).

To see your current usage and limits, use the following command:

```
quota -s -f /home
```

A completely separate quota system is in operation on “/g/data”. You will not see any quota notifications for that filesystem. To see quotas and usage use the following command:

```
lquota
```

4. Environment Modules

The “module” command is available for loading additional software packages into the command line environment. For more information see: [Environment Modules](#)

When loading modules where the version used matters (eg. scripts), it’s recommended that you explicitly state which version of each module to load. This will help avoid unexpected surprises if the default version changes when a new version of the module is deployed in future.

4.1. Python

You can load the default Python 2 interpreter module using:

```
module load python
```

Or for Python 3:

```
module load python3
```

To override the default version, create a “.modulerc” file in your home directory. For example, to change the default Python 2 module:

```
##Module1.0  
  
module-version python/2.7.5 default
```

Note that due to a display bug, after changing the default there will appear to be two default versions in the “module avail” output but a “module show” will tell you which version will actually be loaded by default.



Use 'env' in shebang line

When creating executable Python scripts, use the following shebang line so that it loads the version of Python currently on your path instead of hardcoding an absolute path:

```
#!/usr/bin/env python
```

A limited set of the more commonly used scientific python packages are made available as pre-built modules which are compiled for the CPU features available in the virtual desktop environment. Additionally, some applications can optionally make use of Python. In either case, the “-pyX.X” suffix in each module version indicates which Python version the module can be loaded with. You can only load a module for the same major and minor version of the Python interpreter module that you’ve already loaded. For example, if you are using Python 3.4, you couldn’t load modules built for 3.3 or the 2.x series.

As a convenience, module aliases are defined for the compatibility suffix alone so that you can load the latest version of a module for a given Python release. For example:

```
module load python/2.7.11 netcdf4-python/py2.7
```

If you don’t load a Python interpreter module first, it will usually do so automatically but the version loaded will depend on which one was used to build the module which is often the oldest.

4.1.1. IPython & Jupyter

If you prefer to use IPython you need to load it as an additional module (or install it into your own virtual environment - see further below). For example:

```
module load python/2.7.11 ipython/py2.7  
ipython
```

After you have loaded the “ipython” module, you can also start a Jupyter Notebook server:

```
jupyter notebook
```

4.1.2. Installing Packages

One option for installing additional Python packages is to install them into your own user library located under your home directory (in “~/.local”). The simplest way to achieve this is to use “pip” with the “--user” option, for example:

```
module load pip/py2.7
pip install --user ...
```

Alternatively, if you’re building from a source tarball you can also use the same option with “setup.py”:

```
python setup.py install --user
```

When installing packages, recent versions of “pip” also support Python wheels which are pre-built binary packages. If a wheel is available for the particular package you are trying to install, then it will use that by default which will avoid a possibly tricky compilation step for non-pure Python packages (ie. those partly written in C/C++/Fortran). This is more convenient and faster to install but there is a trade off in that you may be getting a less optimised version in some cases, which may or may not be important depending on your requirements. To prevent “pip” from selecting wheels when installing a package and any of its dependencies, you can use the “--no-binary” option like so:

```
pip install --no-binary :all: ...
```

To list which Python packages are installed and currently visible to the interpreter use:

```
pip list
```

Python searches a number of paths for packages, and generally any package modules that are loaded (which modify “PYTHONPATH”) will override user installed versions of the same packages. Globally installed packages can also do this depending on how they were installed. You can check the “sys.path” variable to see where and in what order Python is looking for packages. Note that by design, packages in your user library won’t be visible at all in a virtual environment unless it was created with the option “--system-site-packages”.

For more information about using “pip” refer to the [user guide](#).

4.1.3. Using Virtual Environments

A better option is to create an isolated Python virtual environment for the packages you want to install. By default, a virtual environment hides any globally or user installed packages, and simply includes just the necessary tools (ie. “pip”) to install further packages. This gives you greater control over what packages are used and allows you to create separate environments to satisfy the different dependencies of applications which might sometimes conflict.

To create and activate a new virtual environment called “py2.7.11”:

```
module load python/2.7.11 virtualenv/py2.7
mkdir ~/venv
virtualenv ~/venv/py2.7.11
source ~/venv/py2.7.11/bin/activate
```

The command prompt will also change to reflect that the environment is now active. Note that because it installs “pip” into each new virtual environment, it’s generally not necessary to also load the “pip” module.

For Python 3, we haven’t provided a “virtualenv” module because the functionality has been incorporated into Python itself, although the command is called “pyvenv” instead. One other difference is that it always installs an older version of “pip”. Therefore, after creating a new environment with “pyvenv” it’s advisable to also upgrade “pip”. For example:

```
module load python3/3.4.3
mkdir ~/venv
pyvenv ~/venv/py3.4.3
source ~/venv/py3.4.3/bin/activate
pip install --upgrade pip setuptools
```

Please bear in mind the following when using Python virtual environments:

- When you create a new environment it is linked to the specific version of Python that was loaded at the time. You should always use the same version of Python whenever you activate the environment. A good idea is to include the version in the name of the environment.
- There is a complicated interaction between activating/deactivating Python environments and the way that software modules modify the environment when they load/unload. Essentially you should load any modules first, then activate the Python environment. Those steps should be done in reverse order when deactivating and unloading the environment. Alternatively, the simplest and safest thing to do is to just close the terminal window when you are finished working in that environment.
- If you load any modules which modify the “PYTHONPATH”, those packages will override any versions of the same packages installed in the virtual environment.

For more information about using Python virtual environments refer to this [documentation](#).

4.2. PBS

The “pbs” module is loaded by default and provides a number of PBS and other gadi commands. These aren’t native commands, instead they are a wrapper script which runs the command on gadi using SSH. Before you can use any of these commands for the first time, you’ll need to run the following command once which will generate a key pair and register it on gadi:

```
remote-hpc-cmd init
```

✔ Host key error

You may encounter the following error running PBS command (such as qstat) on the VDI:

"Host key verification failed."

If this happens, please re-run the "remote-hpc-cmd init" command.

✔ Shared filesystem

Remember that the only file system visible to both environments is “/g/data” so when a command is executed on gadi it won’t be able to access any other locations in the virtual desktop environment; eg. your home directory.

4.2.1 Gadi transition

The PBS scripts have a feature that allows you to interact with Raijin while it remains running. To switch to using Raijin (in your current session), run the following command:

```
export REMOTE_HPC_CMD_HOST="raijin"
```

If you need to go back to using Gadi you can either close your terminal and start again OR run the following command:

```
unset REMOTE_HPC_CMD_HOST
```

5. Software Activation

Some commercial software requires activation of a licence before it can be used. Often this activation process ties the software to the machine that it is running on. This has the following implications in the virtual desktop environment that you need to be aware of:

- Each time that you start a new desktop session, you are allocated a random node in the desktop pool. Even if you terminate a desktop session and then immediately start a new one, you are not guaranteed to get the same node. Hence, the licensing system may think that you are trying to use the software on another computer and refuse to operate.
- The desktop session might end while you are not actively using it, eg. the maximum session lifetime is reached while it is running in the background. In such circumstances, you won’t have an opportunity to deactivate the licence.
- From time to time we rebuild the nodes upon which the desktops are running. This may cause the licensing system to think that the hardware has changed and is a different computer.

5.1. MATLAB

There are various [licence activation](#) options offered by MathWorks for MATLAB. Of these, any of the non-network options such as “Standalone Named User” and “Designated Computer” are affected by the activation issues described above. You might be able to deactivate a MATLAB licence without having access to the node it was activated on by visiting the MathsWorks [licence centre](#) online. Please contact MathWorks for any licensing queries or issues.

⚠ License terms

No matter what method you use to run MATLAB, always ensure that you are using the product within the terms of your MathWorks licence agreement.

5.1.1. Login Named User

MathWorks provides an alternative licensing mode which permits you to use an already activated licence on another machine for the duration that the application is running. This avoids the above issues since the licence is not actually being permanently installed on the desktop. Therefore, this is the recommended method for using MATLAB on the virtual desktops if the option is available for your licence.

Before you can use this method for the first time, you must complete the following:

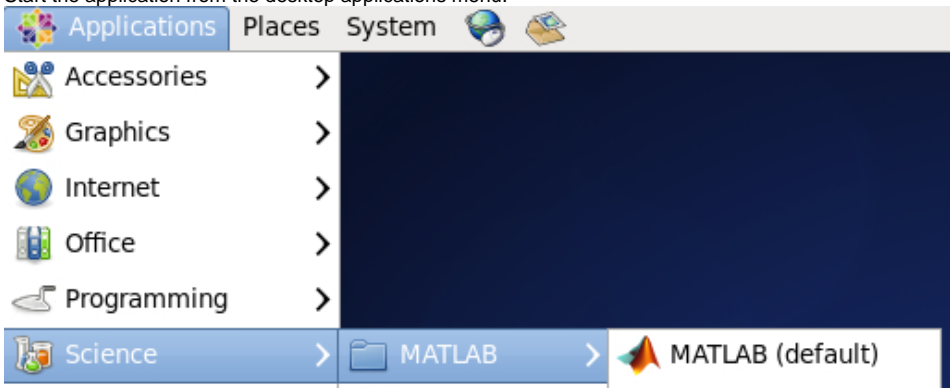
1. Your licence must have already been activated on another machine (eg. your local workstation) as a "Standalone Named User" type. Note that this option is only available to "Individual" and "Group" licences. Don't activate a licence for the first time on a virtual desktop.
2. Enable the "Login Named User" (LNU) option for the licence by visiting the following URL. For group licences, your licensing administrator may have to complete this procedure for you.
<https://www.mathworks.com/licensecenter/lnu>

✔ License registration

If you are a member of an academic institution which has a "Total Academic Headcount" (TAH) licence then ignore the above steps. Simply ask your local MATLAB licence administrator to register your email address.

Then, whenever you want to run MATLAB:

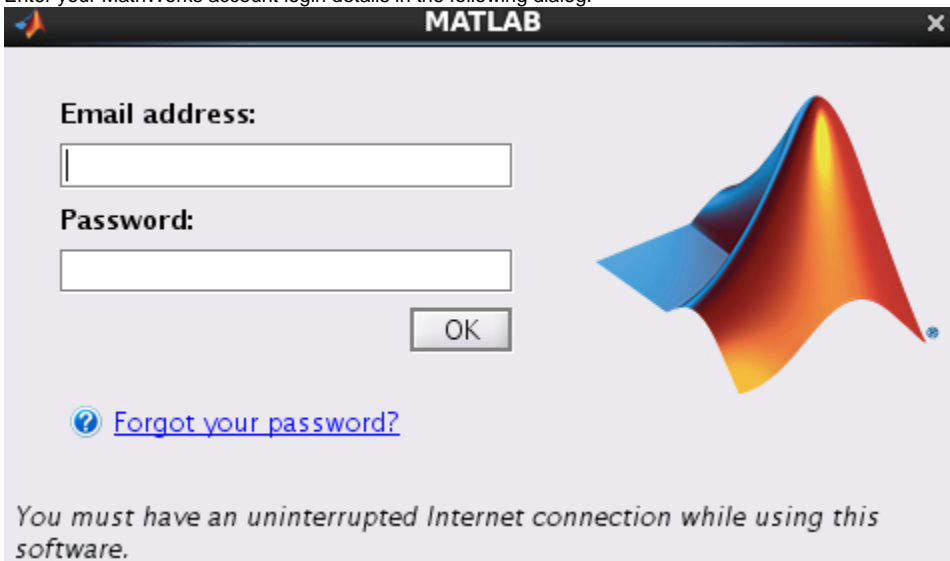
1. Start the application from the desktop applications menu:



Alternatively, open a terminal window and enter the following commands:

```
module load matlab  
matlab -licmode online
```

2. Enter your MathWorks account login details in the following dialog:



5.1.2. Network Licensing

If your organisation has a network based MATLAB licence server, you might be able to use it from the virtual desktops to avoid any activation issues. You will need to contact your local licensing administrator to organise access from the virtual desktop pool.

6. Troubleshooting

The Strudel client may display a dialog inviting you to submit a debug report when an error or other unexpected condition occurs. The form collects information from your system and invites you to enter your name and email address. Please be aware that the information submitted by this form is sent to the developers of the Strudel application and not to NCI. If you are having issues directly related to connecting to or using the virtual desktop system, then please [open a support ticket](#) instead of using the debug report submission dialog. Please make it clear that your request is in relation to the virtual desktops. Ideally, please also attach Strudel's debug log, using the procedure documented below.

6.1 Generating A Debug Log

1. Start Strudel as you normally do. Its main window will appear.
2. At bottom-right of the main window, check the 'Show Advanced Options' checkbox. An additional checkbox labelled 'Show debug window' will appear to the left of the 'Show Advanced Options' checkbox.
3. Check the 'Show debug window' checkbox. An additional window entitled 'Strudel Debug Log' will appear, containing a detailed log of Strudel's operation.
4. Attempt to use Strudel as you normally do. You should encounter the same problem that originally necessitated following this procedure.
5. **Do not** click the 'Submit debug log' button on the bottom of the 'Strudel Debug Log' window. Instead, select all of the text in the 'Strudel Debug Log' window, copy the text, then paste it into the associated support ticket.