



# **UM Rose Conversion Course: Practical Sessions**

**NCAS-CMS**

15 September 2016

## CONTENTS

<b>1</b>	<b>Getting Set Up</b>	<b>2</b>
1.1	Launch MobaXterm . . . . .	2
1.2	Login to PUMA . . . . .	2
1.3	Set up your PUMA environment . . . . .	2
1.4	Make sure you can login to your ARCHER training account . . . . .	4
1.5	Set up your ARCHER environment . . . . .	4
1.6	Set up an ssh connection from PUMA to ARCHER . . . . .	4
1.7	Check this all works by ssh-ing to ARCHER . . . . .	4
<b>2</b>	<b>Getting Started with Rose</b>	<b>6</b>
2.1	Running a Standard Suite . . . . .	6
2.2	Standard Suite Output . . . . .	7
<b>3</b>	<b>Working with suites - A tour of Rose</b>	<b>8</b>
3.1	Suite Discovery - rosie go . . . . .	8
3.2	Config Editor GUI . . . . .	9
<b>4</b>	<b>Further Exercises</b>	<b>13</b>
4.1	Setting up a suite to cycle (equivalent of CRUN) . . . . .	13
4.2	Restarting a suite . . . . .	13
4.3	Adding a new STASH request . . . . .	13
4.4	Error checking of UM inputs . . . . .	14
4.5	Differencing suites . . . . .	15
4.6	Other things to try . . . . .	15
<b>5</b>	<b>Appendix A: SSH FAQs</b>	<b>17</b>
5.1	Setting up a new ssh agent . . . . .	17
5.2	Restarting your ssh agent . . . . .	18

Practical exercises for the UM Rose Conversion course in Reading, 15th September 2016.

NCAS Computational Modelling Services: <http://cms.ncas.ac.uk/>

Contents:

## GETTING SET UP

### 1.1 Launch MobaXterm

- Login to the windows machine with your University of Reading credentials
- If there is a MobaXterm icon on the Desktop, double click it, choose to run the application, then move on to the next Section “Login to PUMA”.

Mobaxterm may also be available through the Start menu, under All programs.

If there is no MobaXterm:

- Open the Chrome browser
- google “mobaxterm download”
- Download the Home Edition
- Choose Save As - and save to the Desktop
- Double click the MobaXterm folder, drag MobaXterm\_Personal\_9.1.exe to the Desktop
- Double click the MobaXterm icon

---

**Note:** Windows sometimes fails to display a MobaXterm icon on the desktop; if this happens, simply double click the application in the Desktop file listing.

---

### 1.2 Login to PUMA

```
xterm$ ssh -Y <puma-username>@puma.nerc.ac.uk
```

### 1.3 Set up your PUMA environment

#### i. Configure ~/.profile

If this is the first time you have used your PUMA account, you will need to create a *.profile*. Copy our standard one:

```
puma$ cd ~
puma$ cp ~um/um-training/setup/.profile .
```

**Note:** If you already have a `.profile`, make sure it includes the lines from the standard file.

---

### ii. Configuring Subversion to access MOSRS

---

**Note:** It is important that the next steps are followed accurately, so you may find it easier to copy and paste text from the online version of these exercises: <http://cms.ncas.ac.uk/documents/training/RoseSept2016/html/setup.html>

---

You need to configure your `~/ .subversion/servers` file as follows:

```
[groups]
metofficesharedrepos = code*.metoffice.gov.uk

[metofficesharedrepos]
# Specify your Science Repository Service user name here
username = myusername
store-plaintext-passwords = no
```

Remember to replace `myusername` with your actual Science Repository Service user name. **Your user name should be a single word in lower case.**

### iii. Authentication Caching

Please ensure that you **comment out or remove** any blank setting for `password-stores` in your `~/ .subversion/config`, for example, after commenting out the line, your file may look like this:

```
[auth]
...
#password-stores =
```

A blank setting for `password-stores` disables all caching, so it must be removed or commented out. A blank setting may be present even if you have never edited the file directly. As a check you can run this command which should not return any output:

```
rose config --print-conf -f ~/ .subversion/config auth password-stores
```

Next you need to set up and test the appropriate password caching method:

Add the following to the top of your `.profile` file:

```
# Ensure .kshrc is sourced in login shells
# (only add this if it is not already done in your .profile)
[[ -f ~/.kshrc ]] && . ~/.kshrc
```

Add the following at the end of your `.kshrc` file:

```
[[ $- != *i* ]] && return # Stop here if not running interactively
. mosrs-setup-gpg-agent
```

Now, whenever you login to PUMA you should be prompted for your Met Office Science Repository Service password. Just press return if you don't want access to the repositories and you shouldn't be prompted again during that login session.

## 1.4 Make sure you can login to your ARCHER training account

An ARCHER username will be provided on the day of the course. Ask the CMS team if you're unsure. The password for these accounts is listed in a file on PUMA: `~um/um-training/um_training_sep2016.txt`.

From PUMA:

```
puma$ ssh <archer-username>@login.archer.ac.uk
```

---

**Note:** It's best to copy and paste the password from the file rather than type it by hand.

---

## 1.5 Set up your ARCHER environment

Once you have successfully logged into ARCHER, copy the following *profile* to your home directory.

```
archer$ cd ~
archer$ cp /work/y07/y07/umshared/um-training/rose-profile .profile
```

## 1.6 Set up an ssh connection from PUMA to ARCHER

If you already have an ssh-agent set up on PUMA, you can use this one to connect to your Archer training account. If you don't already have an ssh-agent setup on PUMA please follow the instructions on *Setting up a new ssh agent* in the Appendix, then skip to Section 1.7

You can copy your ssh key over to Archer using the `ssh-copy-id` script.

First you need to find the name of the public key in your `.ssh` directory.

```
puma$ cd ~/.ssh
puma$ ls
environment.puma  id_rsa  id_rsa.pub  known_hosts  ssh-setup
```

The public key ends with `.pub` and will usually be called `id_rsa.pub` or `id_dsa.pub`.

Now run the script to copy the key to your Archer account, making sure to use the correct name for your key:

```
puma$ ssh-copy-id -i ~/.ssh/id_rsa.pub <archer-username>@login.archer.ac.uk
```

You will be prompted for your Archer training account password.

If successful, you should now be able to login to Archer without a password. If you are prompted for a passphrase you need to re-start your agent - see the Appendix for *Restarting your ssh agent* for instructions.

## 1.7 Check this all works by ssh-ing to ARCHER

From PUMA type:

```
puma$ ssh <archer-username>@login.archer.ac.uk
```

If you get to ARCHER without a password or passphrase, then you're done.

You are now ready to try running a UM Rose job!

## GETTING STARTED WITH ROSE

### 2.1 Running a Standard Suite

To demonstrate how to run the UM through Rose we will start by running a standard suite at UM10.5

#### i. Checkout the suite

In your \$HOME directory on PUMA type:

```
puma$ rosie checkout u-ag263
```

This command checks out a suite containing the N48 science configuration, reporting the directory where the suite resides. When you check out a suite it is always placed in your `~/roses` directory. In this state, the suite is simply a working copy (this terminology should be familiar if you've worked with FCM) - you can edit it and run it but any changes you make will only be held locally. Change to the suite directory by typing:

```
puma$ cd ~/roses/u-ag263
```

#### ii. Edit the suite

Before you can run the suite you need to change the *userid*, *queue* and *account code*. Start up the Rose config editor:

```
puma$ rose edit
```

Now make the following changes:

- Click on **jinja2** in the left hand panel
- Change HPC\_USER (that's your ARCHER training account)
- Change HPC\_ACCOUNT to '**n02-training**'
- Change HPC\_QUEUE to '**course1**'
- Save the suite (*File -> Save* or click the *down arrow* icon)

We will cover the Rose edit GUI in more detail in the next section.

#### iii. Run the suite

The standard suite will build, reconfigure and run the UM. Click on the triangle symbol on the right end of the menu bar to run the suite. Doing this will execute the `rose suite-run` command (more on this later) and start the Cylc GUI (`gcylc`) through which you can monitor the progress of your suite graphically. The cylc GUI will update as the job progresses.



## 2.2 Standard Suite Output

The output from a standard suite goes to a variety of places, depending on the type of the file. On ARCHER you will find all the output from your run under the directory `~/cylc-run/<suitename>`, where `<suitename>` is the name of the suite. This is actually a symbolic link to the equivalent location in your `/work` directory (E.g. `/work/n02/n02/<username>/cylc-run/<suitename>`).

### Rose bush

The standard output and errors from the suite can be easily viewed using Rose Bush.

For suites submitted from PUMA; in a browser navigate to <http://puma.nerc.ac.uk/rose-bush>.

Enter your userid and click “*Suites List*”. You should then see a list of all the suites you have run. Click on “*Jobs List*” for the suite you have just run (u-ag263). You can examine the output of each task using the links, as well as see whether the suite contains failed tasks, or is currently running.

---

**Note:** To launch Rose Bush on MONSooN run the command `firefox http://localhost/rose-bush`

---

### Compilation output

The output from the compilation is stored on the host upon which the compilation was performed. The output from `fcm_make` is inside the directory containing the build, which is inside the *share* subdirectory.

```
~/cylc-run/u-ag263/share/fcm_make/fcm-make2.log
```

### Standard output

The output from the UM scripts and the output from PE0 (the equivalent to what was in the *.leave* file under the UMUI) is placed in the `log` subdirectory. Stdout and stderr are now written to 2 separate files. For a task named *atmos*, the output from the most recent run will be:

```
~/cylc-run/u-ag263/log/job/1/atmos/NN/job.out
```

And the corresponding error file is:

```
~/cylc-run/u-ag263/log/job/1/atmos/NN/job.err
```

Here NN is a symbolic link created by Rose pointing to the output of the most recently run *atmos* task.

Take a look at the `job.out` and `job.err` for the *atmos* task either on the command-line or through Rose Bush. Scroll down until you see text that is familiar from the *.leave* files.

### Binary output - work and share

By default the UM will write all output to the directory it was launched from, which will be the task’s `work` directory. However, all output paths can be configured in the GUI and in practice most UM tasks will send output to one or both of the suite’s `work` or `share` directories.

```
~/cylc-run/<suitename>/work/1/atmos
```

or

```
~/cylc-run/<suitename>/share/data
```

Take a look at the `work` and `share` directories for your suite.

## WORKING WITH SUITES - A TOUR OF ROSE

### 3.1 Suite Discovery - rosie go

#### i. Searching for suites

`rosie go` is the suite manager GUI. Launch `rosie go` by typing:

```
puma$ rosie go
```

By default it will show all the suites that you have checked out locally. You can search for suites by typing a word/phrase into the search box and either click the search button or press `Enter`. The search looks for the entered word/phrase in **any** of a suite's properties. More advanced searches can be run by clicking the `+` button next to the search button.

For these exercises we will be using the suite `u-ag137`, which is an N96 GA7 suite. In the *Search* box type `u-ag137` and either click the search button or press `Enter`. In the results pane you should see a suite with idx **u-ag137** owned by **annetteosprey** listed.

Hover the mouse over the suite to view a tooltip containing more details about the suite.

From `rosie go` it is possible to create, checkout, delete, edit, and run suites, and more. Details on all the features of `rosie go` can be found in the user guide - <http://metomi.github.io/rose/doc/rose-rug-rosie-go.html>. If you wish, take some time now to explore the GUI some more.

Before continuing make sure you have suite `u-ag137` listed in the results pane.

#### ii. Copy a suite

Now make a copy of the example suite. There are 2 ways to copy a suite either via the `rosie go` GUI or on the command line.

- **In “rosie go”:** Right click on suite `u-ag137` and click `Copy Suite`. This will launch a wizard for you to edit suite discovery information.

or

- **On the command line:** Issue the command

```
puma$ rosie copy u-ag137
```

The `rose-suite.info` file containing the suite discovery information is opened in an editor for you to edit before the suite is created.

A new suite will be created in the MOSRS roses-u repository (<https://code.metoffice.gov.uk/trac/roses-u/browser>) and checked out into your `~/roses` directory.

## 3.2 Config Editor GUI

The Rose config editor in combination with the meta-data file which describes UM inputs is the new GUI for editing UM apps. Building and running the UM under Rose requires (at least) two separate apps: an `fcmake` app to build the model executable and a UM app to configure the runtime namelists and environment variables. Coupled models may require additional `fcmake` apps, one for each executable to be built.

### i. Launch the config editor GUI (`rose edit`)

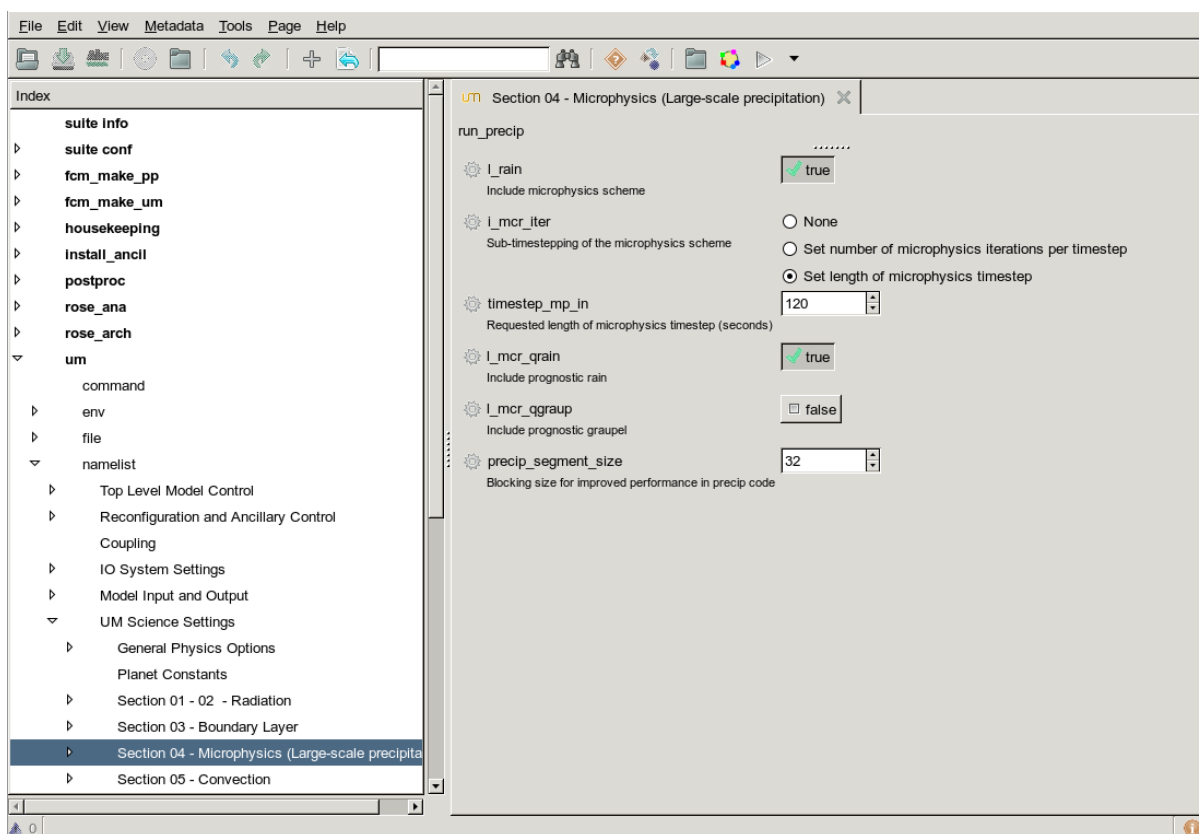
In `rosie` go double click on your newly created suite. (Alternatively, on the command line `cd ~/roses/<new_suitename>` and type `rose edit &`). The config editor GUI will now launch.

**On the left hand side is a panel containing a tree listing the apps in the suite:**

- `suite conf` - General suite configuration options
- `fcmake_pp` - Extract and build the post-processing scripts
- `fcmake_um` - Extract and build the UM source code
- `housekeeping` - Tidies up log files, old work and data directories
- `install_ancil` - Install ancillary files
- `postproc` - Post-processing setting
- `rose_ana` & `rose_arch` - Compare dumps and Archive logs (Met Office specific)
- `um` - The UM atmosphere and reconfiguration settings

### ii. Editing namelists

The namelists for an app are contained in the namelist section. Let's take a look at the science namelist for Large Scale Precipitation, `run_precip`.



One of the first differences you should notice compared to a UMUI panel is that the UM namelist item names are visible in addition to the more familiar descriptions underneath.

Range and type checking of variables is done as soon as the user enters a new value. Try changing the value of *timestep\_mp\_in* to 0. This will cause an error flag to appear, hover over the error for more information and click the undo button to revert to the original value. If a variable has associated help information this can be accessed by either clicking on the variable name or on the cog next to the variable and selecting Help.

Some larger science sections have been divided into subsections, take a look at Section 05 - Convection for an example of this. To open a section in a new tab click with the middle mouse button, expand the section by clicking the page triangles. Rose edit has a search box which can be used to search item names. Try searching for *astart*, you will be taken directly to the Dumping and Meaning panel.

Trigger ignored settings are hidden by default and only appear to the user when the appropriate options are selected. Open the Gravity Wave Drag panel, if you change *i\_gwd\_vn* from 5 to 4 the options available change. Click the save button to apply these changes to your app. Let's take a look at what effect this has had to the `rose-app.conf` file, run `fcml diff` in the suite directory.

```
puma$ cd ~/roses/<suitename>
puma$ fcm diff -g
```

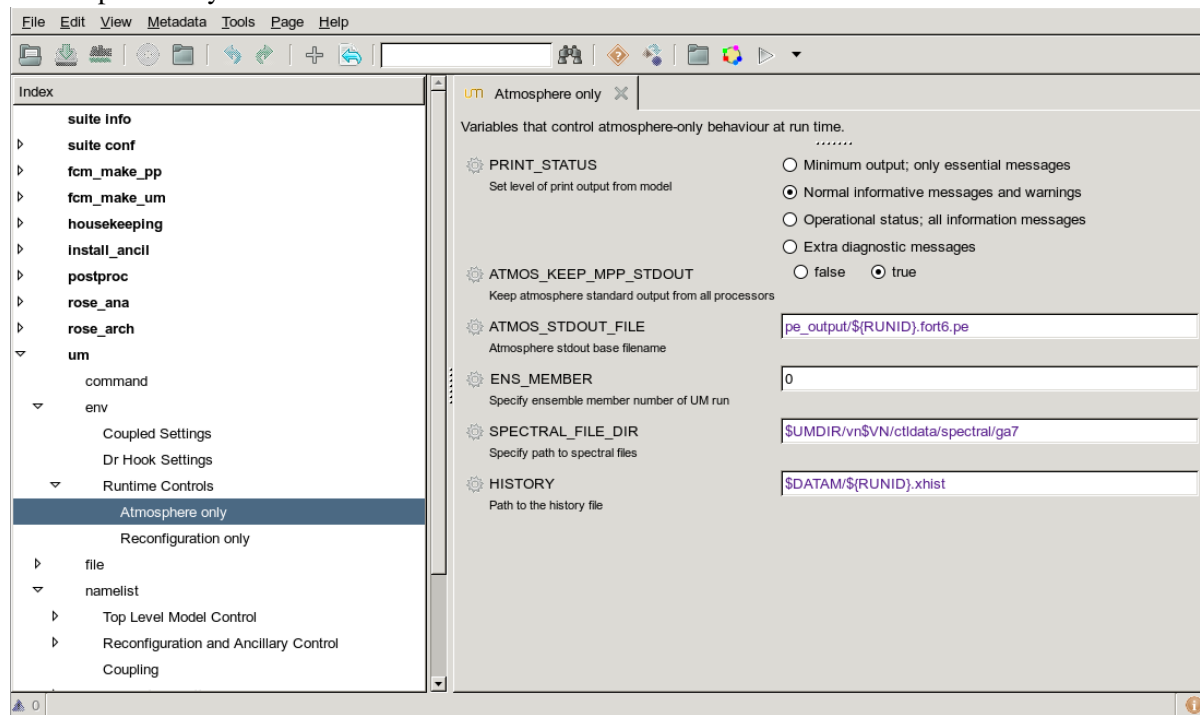
You should see that several namelist items have had `!!` added to the start of the line. This tells Rose to ignore these items when processing the app file into a Fortran namelists. Should you wish to see all variables on a panel select "View All Ignored Variables" and "View Latent Variables" from the "View" menu.

Switch back to the Rose edit window and click the undo button to revert the changes and then save the suite again. To view all changes made to the suite in the current session click on the "Undo/Redo

Viewer” in the “Edit” menu.

### iii. Editing environment variables

In addition to namelist items the UM also uses environment variables to control aspects of runtime configuration. These can be found in the *env* section of the UM app. The image below shows the “Atmosphere only” section.



### iv. Making changes and running the suite

Firstly make the changes required for you to run the suite. Go to panel “*suite conf -> UM hosts*” and change the following variables:

- Change “*Queue to run in*” to be **course1**
- Change “*Account group for HPC tasks*” to be ‘**n02-training**’

This suite is set up so that specifying your username on the remote HPC is optional. If your PUMA username is the same as your username on ARCHER, or if the remote username is set in your `~/.ssh/config` file Cylc will be able to submit your suite without having to explicitly set it in the suite. However, on this course we are using training accounts on ARCHER so we will set the userid.

- Click *View -> View Latent Variables*. You should see *HPC\_USER* appear in the panel greyed out.
- Click the + sign next to it and select *Add to configuration*
- Enter your ARCHER training username. (E.g. ‘ncastr01’)
- Save the suite
- Run the suite either through the GUI or from the command line.

### v. Fixing errors and reloading the suite

If all goes according to plan the suite should fail in the reconfiguration. This will be indicated in the gcylc GUI with a red square and the state “*failed*”. Examine the *job.out* and *job.err* files to find the cause of the problem. You can view these files through Rose bush, however you can also view them

quickly and easily directly from the Cylc GUI. Right click on the failed recon task and select “*View -> job stderr*”.

The error should be obvious in this case, it’s failed to open the initial dump file. Point your suite to the correct initial dump file. Fixing this problem isn’t quite as easy as it sounds. A search for **ainitial** in the Rose edit GUI will take you to the “*General reconfiguration options*” panel. Can you see the problem? The initial dump location is set with an environment variable: *AINITIAL*. Suites can be and are set up differently and there will be times when you need to edit the cylc suite definition files directly.

In your suite directory on PUMA (`~/roses/<suitename>`) use `grep` to search for where the variable *AINITIAL* is set. Edit *AINITIAL* in the appropriate `.rc` file to point to the correct initial dump file. Hint: This suite is setup for multiple platforms, make sure you edit the file appropriate to ARCHER.

Rather than stopping and restarting the suite, we will reload the suite definition and then retrigger the failed recon task to run:

```
puma$ cd ~/roses/<suitename>
puma$ rose suite-run --reload
```

The suite will reload and another gcylc window will appear. In the gylc window right click on the failed recon task and select “*Trigger (run now)*”. The task will resubmit and this time the suite should succeed.

## FURTHER EXERCISES

Now that we have built the suite, there is no need to rebuild it each time you run it. Switch off compilation of the UM and reconfiguration. Hint: See the “*suite conf*” section.

### 4.1 Setting up a suite to cycle (equivalent of CRUN)

Cycling is the equivalent of a CRUN in the Rose/Cylc world. Let’s set up the suite to run for a total of 3 days cycling (resubmitting) every day.

Navigate to the “*suite conf -> Initialisation and cycling*” panel. Change the *Total Run length* to 3 days and check that the *Cycling period* is set to 1 day. Remember the UM itself works in the same way that it did under the UMUI so you will also need to make sure that the dumping period is set appropriately.

Save your changes and run the suite.

### 4.2 Restarting a suite

Let’s now extend this run out to 5 days. Change the “*Total run length*” to 5 days and save the suite.

Having already run the first 3 days we just want the suite to pick up where it left off and run the remaining days. To do this we *restart* the suite, by typing:

```
puma$ rose suite-run --restart
```

The Cylc GUI will pop up and you should see the run resuming from where it left off.

### 4.3 Adding a new STASH request

Let’s try adding a new STASH request to the UM app. The four STASH namelists: streq (STASH Requests), domain (Domain Profiles), time (Time Profiles) and use (Usage Profiles) can all be found in the STASH section. Navigate to “*um -> namelist -> Model Input and Output -> STASH Requests and Profiles*”.

To add a new STASH request click the “New” button in the STASH Requests section. The following window will appear in order for you to browse all STASHmaster entries.

Section	Item	Description	?	#	datat	dumpp	grid	halo	lbvc
0: Primary fields				80					
1: SW radiation				83					
2: LW radiation				90					
3: BL and surface				139					
4: LS Precipitation				25					
5: Convection				74					
6: Gravity wave drag				16					
	2	U COMPNT OF WIND AFTER G.WAVE DRAG	0	1	2	18	3	65	
	3	V COMPNT OF WIND AFTER G.WAVE DRAG	0	1	2	19	3	65	
	101	EASTWARD FLUX - SPECTRAL PSEUDOMOM.	1	1	2	18	3	65	
	102	SOUTHWARD FLUX - SPECTRAL PSEUDOMOM.	1	1	2	19	3	65	
	103	WESTWARD FLUX - SPECTRAL PSEUDOMOM.	1	1	2	18	3	65	
	104	NORTHWARD FLUX - SPECTRAL PSEUDOMOM.	1	1	2	19	3	65	
	105	EASTWARD FORCE FROM SPECTRAL GW	1	1	2	18	3	65	
	106	NORTHWARD FORCE FROM SPECTRAL GW	1	1	2	19	3	65	
	111	E. FLUX SPECTRAL PSEUDOMOM. P LEVS	1	1	2	11	3	8	
	113	W. FLUX SPECTRAL PSEUDOMOM. P LEVS	1	1	2	11	3	8	
	115	EAST. FORCE FROM SPECTRAL GW P LEVS	1	1	2	11	3	8	
	181	GW HEATING TEMPERATURE INC	0	1	2	1	3	65	
	185	U WIND INCR: gwd scheme	0	1	2	18	3	65	
	186	V WIND INCR: gwd scheme	0	1	2	19	3	65	
	201	X COMPONENT OF GRAVITY WAVE STRESS	1	1	2	18	3	65	
	202	Y COMPONENT OF GRAVITY WAVE STRESS	0	1	2	19	3	65	
	203	STANDARD DEVIATION OF OROGRAPHY	0	1	2	1	3	129	
	204	OROGRAPHIC GRADIENT XY COMPONENT	0	1	2	1	3	129	

By default STASHmaster entries are grouped together by Section code. It is possible to group items by any of the STASHmaster codes using the Group drop down list. The View button contains options to display the STASHmaster entry values and/or the column titles with explanation text and to select which columns to show/hide.

Select a STASH item and click *Add* to add it to the list of STASH requests. In the STASH Requests panel click on the empty *dom\_name*, *tim\_name* and *use\_name* fields of the new request and select appropriate profiles from the drop down lists. These lists are populated from the entries of the time, use and domain namelists of the current UM app.

Once you have added a new STASH request, you need to run a macro to generate an index for the namelist. To do so click on the *Macros* button, then select *stash\_indices.TidyStashTransform*. A box will pop up listing the changes the editor is going to make, click *Apply*.

## 4.4 Error checking of UM inputs

In addition to the type and range checking of namelist items and environment variables, more thorough checks can be made using Rose macros and the fail-if/warn-if metadata.

### Check fail-if/warn-if

First let's check if the suite contains any options which trigger the fail-if and warn-if checks in the UM metadata. Select "Check fail-if, warn-if" from the "Metadata" menu. As this suite is setup correctly "FailureRuleChecker: No problems found" should appear at the bottom right of the window.

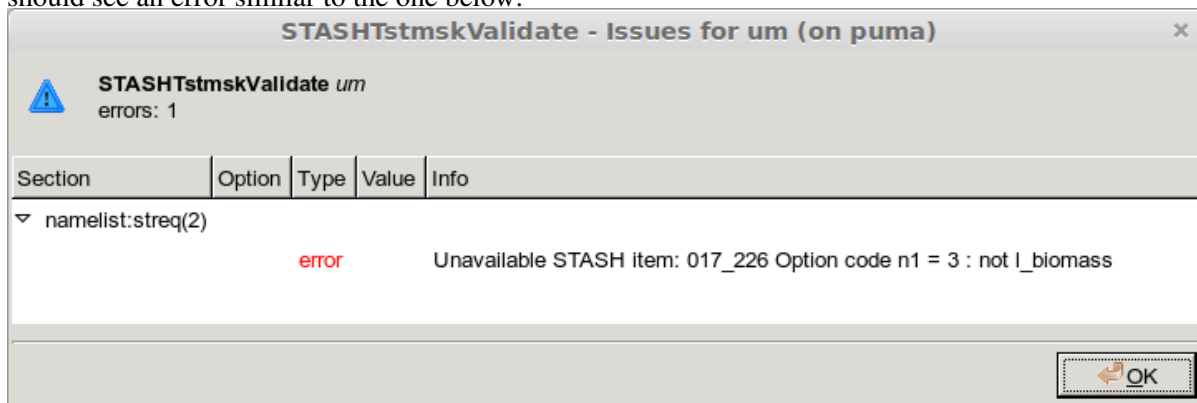
Now let's try and introduce both a warning and a failure. We're going to change the boundary layer option "alpha\_cd". Either navigate to "Section 03 - Boundary Layer -> Implicit solver options" or type "alpha\_cd" into the search bar. Click on the plus sign to add an array element to alpha\_cd and type 1.5 into the new box. Next navigate to "Reconfiguration and Ancillary Control -> Output dump grid sizes



and levels” and increase the number of ozone levels to 86. Now when we run the fail-if, warn-if checker again there should be both a warning and an error. Use the undo button to put the settings back to how we found them and run the checker again. It is strongly recommended that whenever namelists and environment variables are modified that the fail-if, warn-if checker is applied before running the suite.

### STASH validation macro

A Rose macro has been provided to ensure that the STASH output requested is valid given the science configuration of the app. To put this to the test try adding, for example, an Aerosol diagnostic. To run the STASH validation macro select “stash\_testmask.STASHTstmskValidate” from the list of available macro for the UM app within the “Metadata” menu. As Aerosols are not switched on in the app you should see an error similar to the one below:



## 4.5 Differencing suites

Currently there is no Rose tool to difference two suites. Since a suite consists of text files it is simply a matter of making sure all the Rose configuration files are in the common format by running *rose config-dump* on each suite and then running *diff*.

We will difference your copy of the GA7.0 suite with the original one:

```
puma$ cd ~/roses
puma$ rosie checkout u-ag137
puma$ rose config-dump -C u-ag137
puma$ rose config-dump -C <your-suite-name>
puma$ diff -r u-ag137 <your-suite-name>
```

## 4.6 Other things to try

Here are a few suggestions of other things to try if you have time.

- i. Experiment with stopping a suite mid-run and restart it again. (Hint: Use the *Control* menu in Cylc GUI to stop the suite)
- ii. Start a suite in held mode. This allows you to trigger the running of tasks manually.

To start a suite in held mode add `-- --hold` to the end of the `rose suite-run` command:

```
puma$ rose suite-run -- --hold
```

The first `--` tells Rose that all subsequent options should be passed on to Cylc. This is why the `hold` option should be added to the end of the command, after any Rose options. Once the suite has started all tasks will be in a held state. It is then possible to select which tasks are run by right clicking on a task in the Cylc GUI and manually triggering it.

**iii.** Take a look around the suite definition files (E.g. `suite.rc`, `archer.rc`)

## APPENDIX A: SSH FAQs

This Section provides instructions for some common ssh tasks. If you have any problems, contact a member of the CMS team.

### 5.1 Setting up a new ssh agent

#### i. Generate authentication key on PUMA and install it on ARCHER.

Run the `install-ssh-keys` script. This will take you through ssh-key creation and copy the key over to ARCHER.

```
puma$ source ~um/um-training/install-ssh-keys <archer-username>@login.archer.ac.uk
```

When prompted to **Enter passphrase**, this should be a fairly complicated and unguessable passphrase. You can use spaces in the passphrase if it helps you to remember it more readily. It is recommended that you don't use your password in case it is hacked.

**Warning: DO NOT** use an empty passphrase as this presents a security issue.

After generating your ssh-key, the script will copy it over to ARCHER.

When prompted for **Password**, enter your ARCHER password.

#### ii. Verify the authentication works.

```
puma$ ssh <archer-username>@login.archer.ac.uk
Enter passphrase for key '/home/<puma-username>/.ssh/id_dsa':
[TYPE_YOUR_PASSPHRASE]
```

If you don't get asked for your Passphrase (i.e. RSA key), then something has gone wrong. In this case, make sure the public key, was successfully copied over to ARCHER by logging into ARCHER and opening the file `~/.ssh/authorized_keys`. It should contain something similar to:

```
ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAt1JmHYgsuf0UWVVLqNqnDSaUUP2xJ+Um0H5WnUt/i
2mxhlBrwOtvVWRjnzo5Ecy1ZJs/Cg5JVe4UR6toqNXbZG1RXscLQnQoPAvzFoWLzfp7Q3lrzeC1S
kM2FWfWC38ga3Svs6fm63/I7WmJy+4D8BWWaXj/9yM1OskFj6yfwItr150rwwNauOQbWJh17I/Kk
fhVPBvZ9vHiAK4cjUMQ9fFS1dij3GSBmOfu2RuMgNNg9y1MLSzEk2242F4tOg7paTk7wwUZ+ZLqR
BtT2aREnjIGI7KvACBZD1y40tXXPIZw9m2D10dK7mFQ2/YFWh2/NAmkFMXzDOmkg0biq1m+QKw==
ros@puma
```

If it doesn't, and no errors were reported from the `install-ssh-keys` script, please ask for assistance.

Once you have this part working, log out of ARCHER.

### iii. Start up ssh-agent.

Run the following command and type your passphrase:

```
puma$ ssh-add
Enter passphrase for /home/<puma-username>/.ssh/id_rsa:
[TYPE_YOUR_PASSPHRASE]
```

The ssh agent should keep running even when you log out of puma, however you may need to restart it from time to time. For instructions on how to do this see [Restarting your ssh agent](#) below.

## 5.2 Restarting your ssh agent

Normally your ssh agent persists even when you log out of puma. However, from time to time it can vanish.

If you are prompted for your passphrase, this means the ssh agent has stopped for some reason. The agent *should* have been re-initialised when you logged into puma, but you will need to re-associate your ssh keys to the agent.

To do so, run:

```
puma$ ssh-add
```

If successful this will prompt for your passphrase:

```
Enter passphrase for /home/<puma-username>/.ssh/id_rsa:
```

Sometimes this step will fail with the following error:

```
Could not open a connection to your authentication agent.
```

In this case, the agent is not running. Usually this is because of an environment file. Delete the following:

```
puma$ rm ~/.ssh/environment.puma
```

Then log out of puma and back in again. You should hopefully see a message similar to:

```
Initialising new SSH agent...
```

And you should now be able to run ssh-add successfully.