JULES From Scratch

Updated for the Hydro-JULES Summer School 2021

Dr Toby Marthews

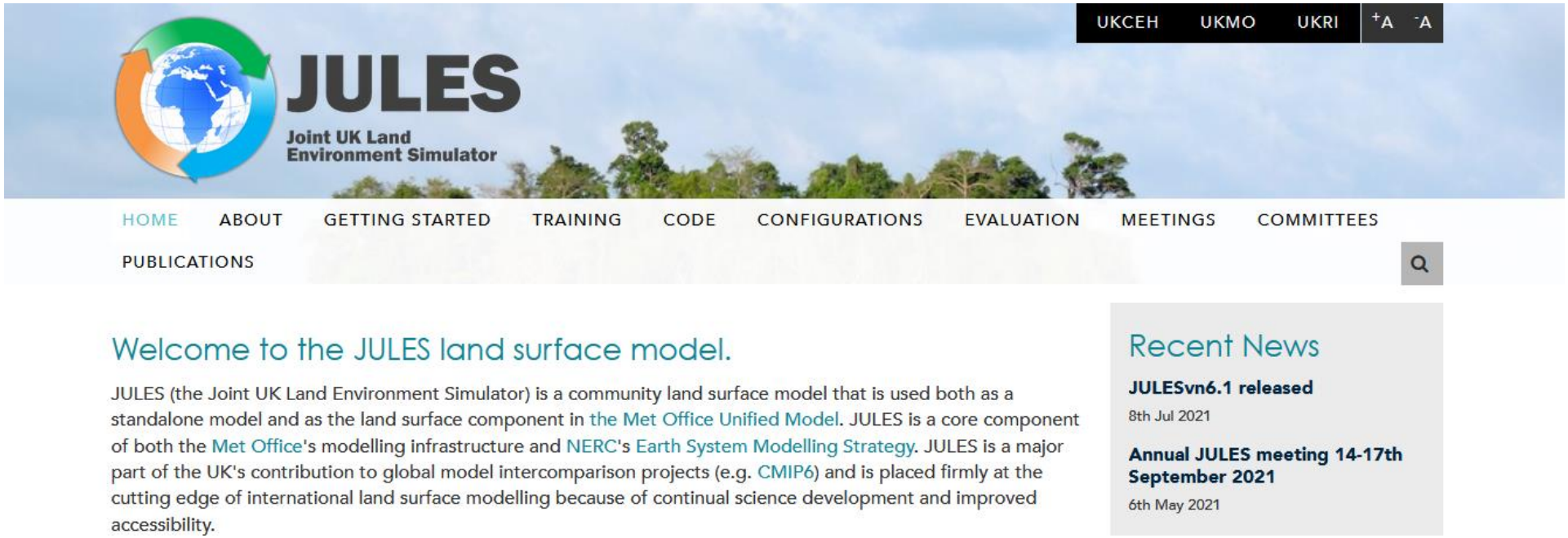
Hydro-JULES

UK Centre for Ecology & Hydrology

British Geological Survey
NATURAL ENVIRONMENT RESEARCH COUNCIL

National Centre for Atmospheric Science
NATURAL ENVIRONMENT RESEARCH COUNCIL

The JULES Land Surface Model

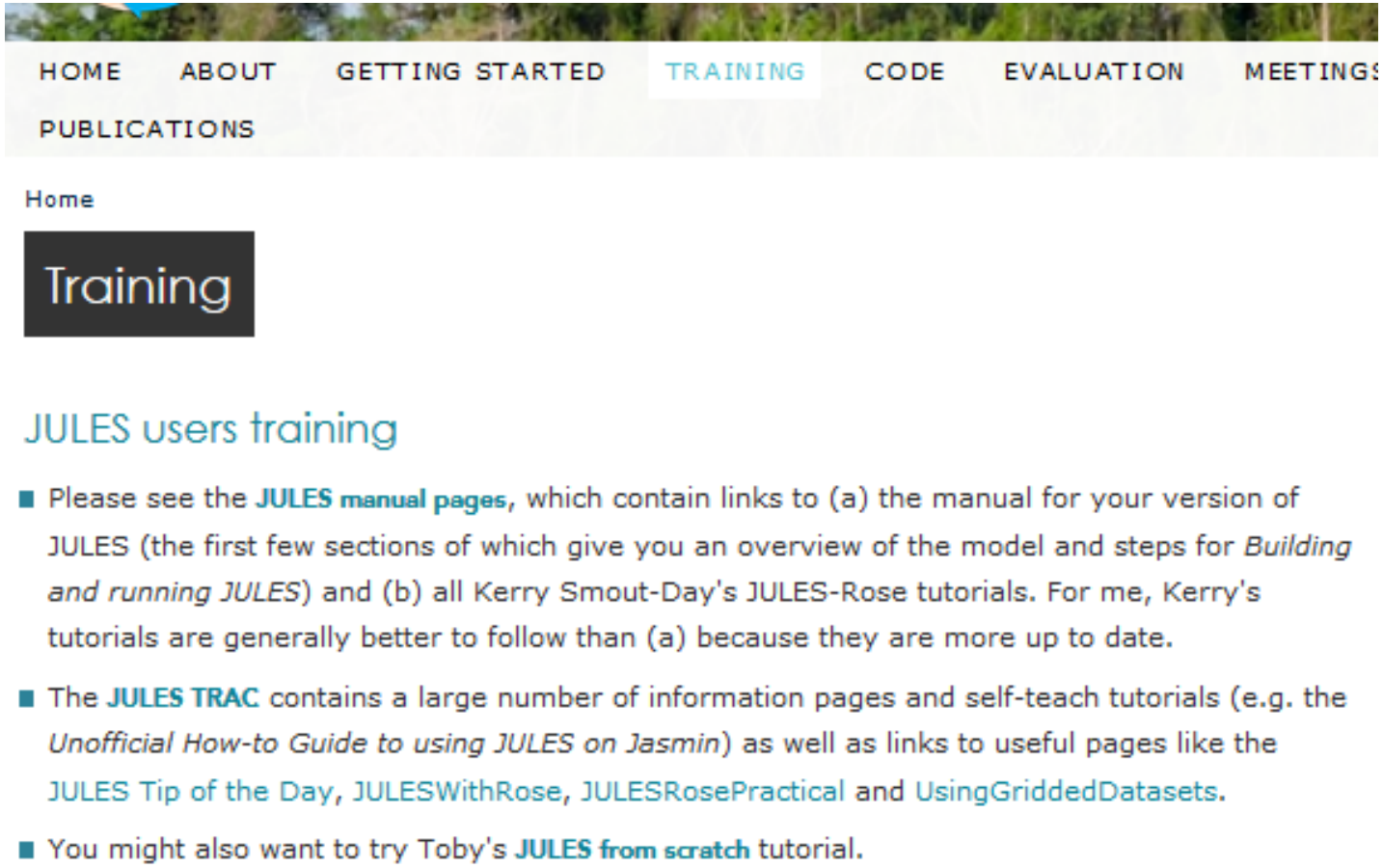


The main JULES website is:

https://jules.jchmr.org/

and this contains a wide range of information including how to get set up with JULES, what data you need and much other information including PDFs of presentations given at the Annual JULES Science meetings so that you can get an idea of what is happening in the wide international JULES Community of users and developers.

The JULES Land Surface Model



The JULES website has lots of training tutorials (many of them are actually hosted on the *JULES TRAC*, which is a ‘sister website’ to the JULES website containing more developer-oriented material).

JULES From Scratch

From scratch

JULES From Scratch is an alternative how-to for those who don't work on a system where Cylc, Rose and FCM have already been installed and/or can't use the Met Office Virtual Machine (e.g. for security reasons). I'm also assuming little or no familiarity with any recent version of JULES.

In order to follow this tutorial you need:

- A **MOSRS login** (see here) (this is free to everyone, whether in the UK or not). You should also subscribe to the **email support** lists (see here), by the way.

Recent News

JULESvn6.1 released
8th Jul 2021

Annual JULES meeting 14-17th September 2021
6th May 2021

JULESvn6.0 released
2nd Mar 2021

JULESvn5.9 released
19th Nov 2020

The *JULES from Scratch* tutorial is my attempt to lead you through all you need to get started on JULES. It takes you through setting up a JULES run right from the basics, including all required installation steps.

I (Toby) wrote this in May 2017 and it has been updated several times since.

Please note you may need a bit of familiarity with UNIX commands to accomplish the steps below. If you need a quick refresh of this, please see my ***UNIX Basics*** on https://www.tobymarthews.com/resources.html .

JULES From Scratch: Cylc, Rose, FCM and JULES

Start here:

- Access your command prompt (either locally or on a server)
- Run these test commands to see which (if any!) of Cylc, Rose and FCM you have installed
- EVEN IF YOU HAVE THEM ALL, it's almost certain that your profile will not be set up correctly, so you'll need to go through all the steps on the 'get ...' pages anyway
- Start with Cylc, then do Rose, FCM and JULES.

https://jules.jchmr.org/content/scratch

CEH TOOLS G Img Send WoS GDrive Code News TV RBS Gather ME Weebly Toby C

Install Cylc, Rose and FCM and cache your MOSRS password

You need to have three programs installed on your system: Cylc, Rose and FCM. *All three* of these are necessary for running JULES and I've done a page about installing each with all the steps required (linked here below).

Ideally, these three will be installed globally by your system administrator, but if that isn't possible (or you are the system administrator) then you will need to go through some/all of these steps for a local installation. First check what is on your system by putting these commands into a UNIX shell:

```
echo $SHELL
     #You need to be in a bash shell, so if this returns anything except "/bin/bash" ask your
system administrator how you can get into bash
cylc check-software
     #If Cylc is installed, this will give you a message saying "Full-functionality OK" (and
it'll also warn you if you are missing Python (you need vn>=2.6); n.b. you can still run JULES
even if missing libraries PyGraphviz and PyGTK)
rose --version
     #Just check you don't get "command not found"
fcm --version
     #Just check you don't get "command not found"
```

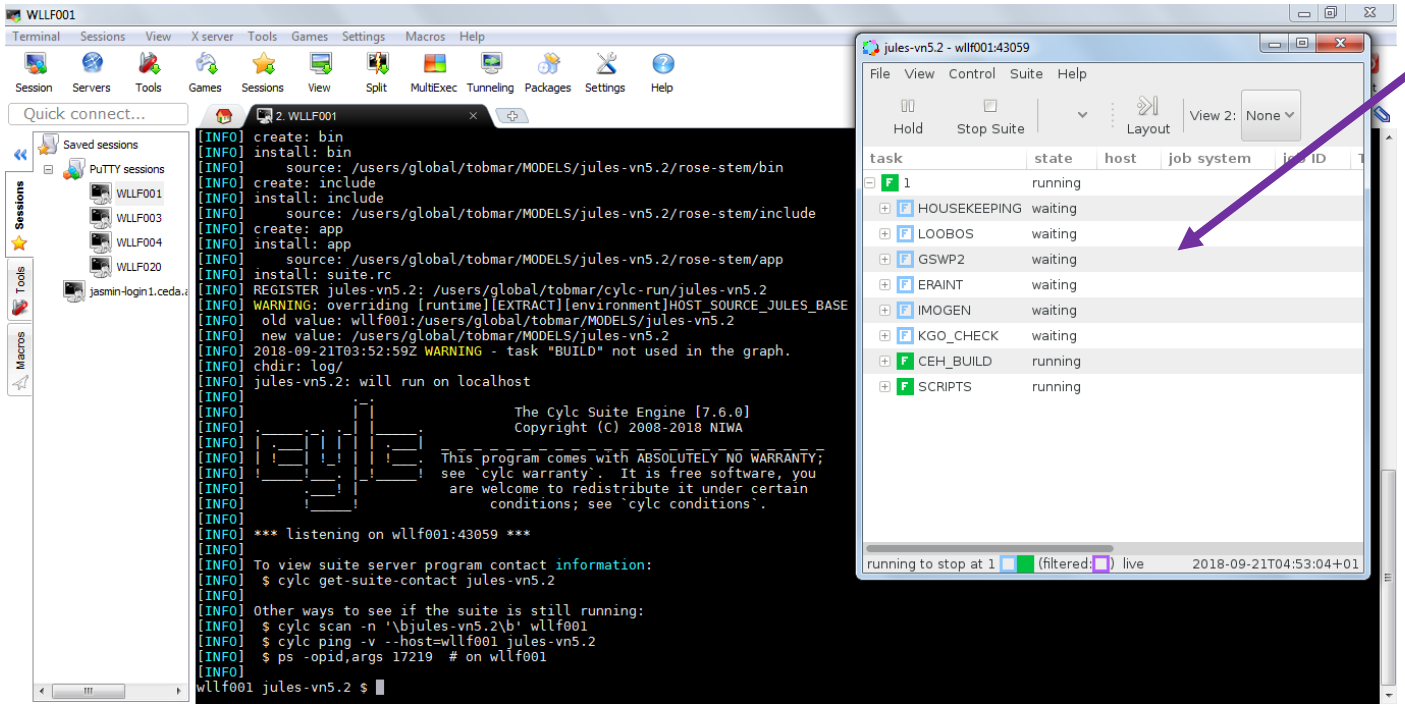
EVEN IF ALL SEEM TO BE INSTALLED OK, your personal profile settings may not be correct for using JULES so please do anyway go through the installation links below to make sure your environment setup is OK and your MOSRS password has been correctly cached.

- Install *Cylc* from GitHub following these steps

JULES From Scratch: Check installation

We can't yet do a run of JULES, but I can show you some screenshots of what running it looks like:

- First, some text comes up on the command line including an ASCII-art "cylc" (see black background text below)
- Next, a new window appears called the *Cylc GUI* window



New 'Cylc GUI' window showing the progress of the Rose Stem tests.

If you close this window by accident before they complete, type:

```
cylc gscan &
```

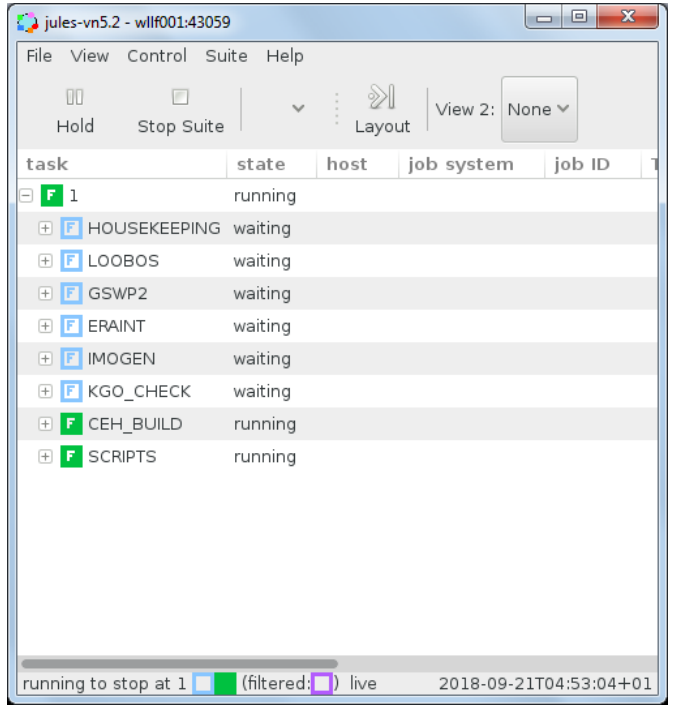
and double-click on the right job to find it again.

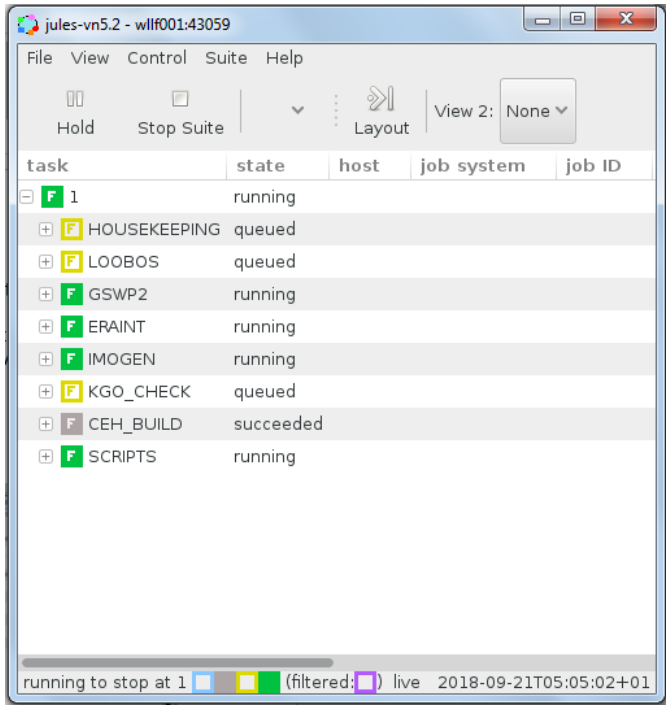
Do the following if the suite has disappeared from Cylc Gscan:

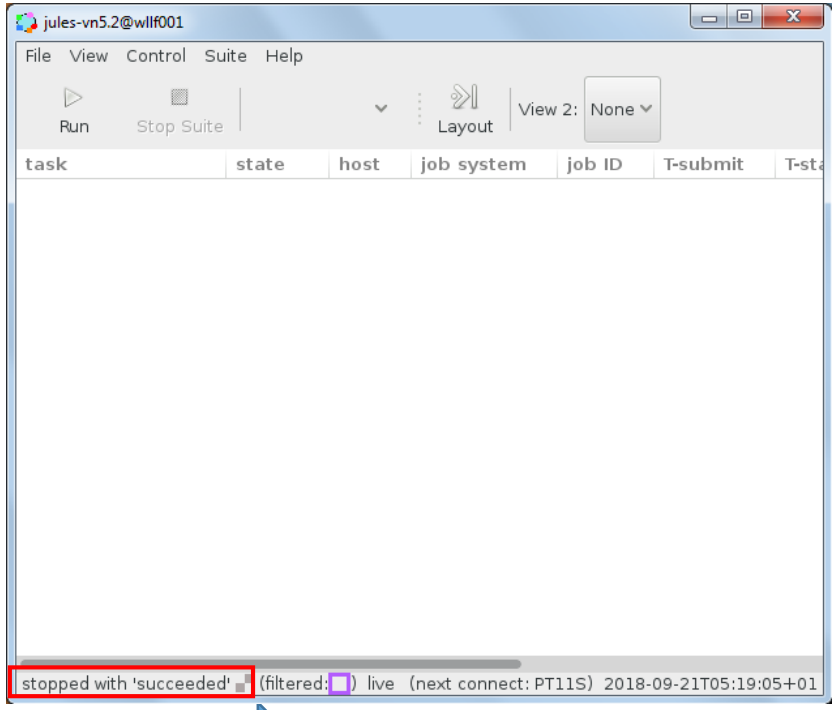
```
cd ~/roses
rose suite-run -C $RSUITE --restart
```

JULES From Scratch: Check installation

A run then progresses to completion something like this (n.b. this is a rather complicated JULES run with lots of jobs):







At least 15 min

Note 1: The “Stop Suite” button doesn’t actually stop the whole suite (!!) - see ‘Troubleshooting’ slide below
Note 2: The jobs will all disappear when the last one completes (there’s no option for changing that).

JULES From Scratch: Doing your own simulation

OK: Now we should all be set up and we know what a run should look like. How can we actually use JULES for something?

In common with any model of this type, JULES needs THREE elements to do a simulation:

Driving data,
Ancillary/prescribed data and
Control files

(together these are called the **model configuration**). I've put advice on where to get these from on https://jules.jchmr.org/content/getting-started.

The control files usually come in the form of a *Rose suite*, which is what we need to look at next.

Here's how to get started as a JULES user:

- Get started
- Email/support lists
- Other files you need (configuration files):
  1. Driving data
  2. Ancillary data
  3. Control files
- Analysis tools

JULES From Scratch: Rose suites

A *Rose suite* is a ‘container’ for *apps*. JULES Rose suites always have two apps that run sequentially: *fcm_make* (the compilation step) and *jules* (actually running the model) (see my FAQ about Rose suites on https://jules.jchmr.org/content/rose-suites).

For example, my suite *S* is stored on my system at ~/roses/<idx code for S>/ (which from now on I’m going to refer to as $RSUITE):

```
export RSUITE=$HOME/roses/<idx code for S>
echo $RSUITE
```

I can open this suite in two different ways:

1. On UNIX using the Rose Edit GUI: `rose edit -C $RSUITE &`
2. On UNIX opening individual parts in a text editor: `nedit $RSUITE/rose-suite.info $RSUITE/suite.rc $RSUITE/app/fcm_make/rose-app.conf $RSUITE/app/jules/rose-app.conf &`

JULES From Scratch: Rose suites

As mentioned above, you can edit a Rose suite in TWO different ways:

1. On UNIX using the Rose Edit GUI: `rose edit -C $RSUITE &`

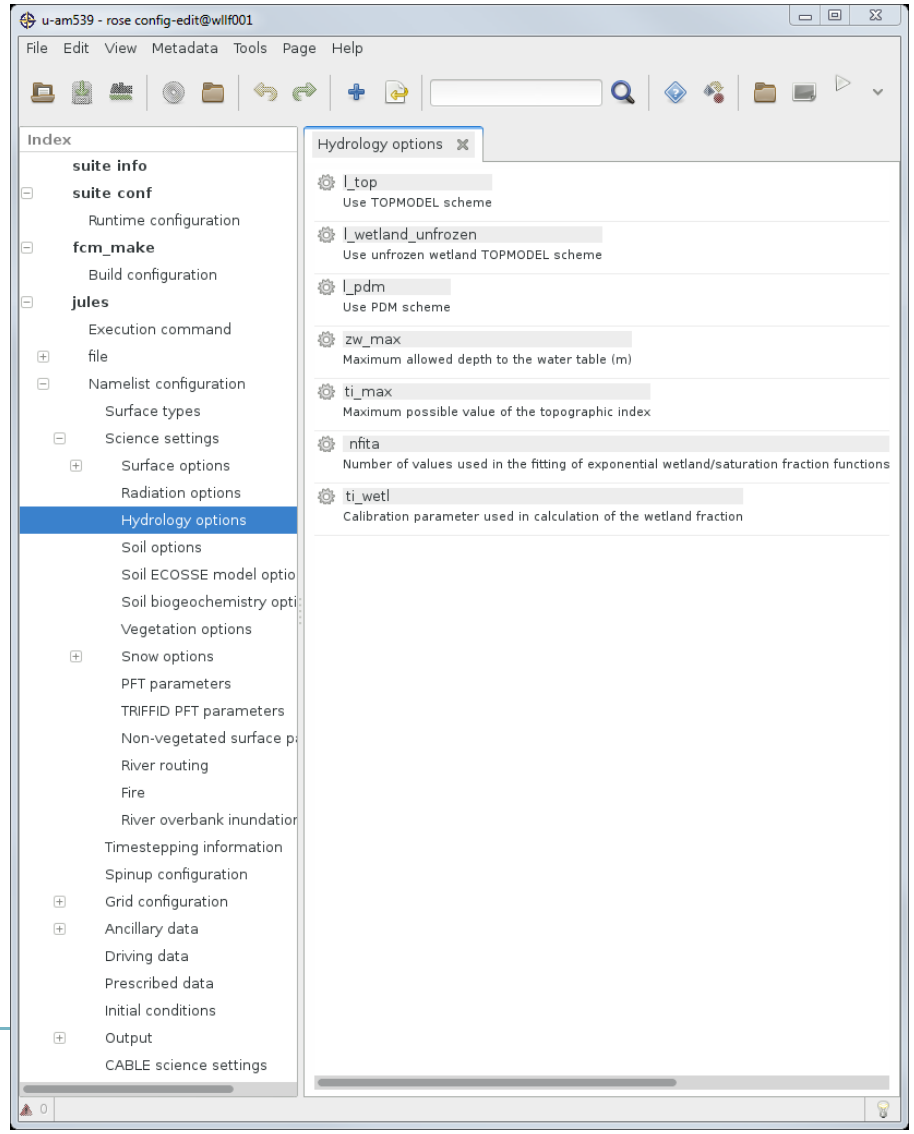
Tip: Always change your settings in the Editor so that View -> View All Ignored Variables is selected. This is because the search box on the main screen only searches visible parameters (e.g. try searching for *l_veg_compete*).

2. On UNIX using a text editor: `nedit $RSUITE/rose-suite.info $RSUITE/suite.rc $RSUITE/app/fcm_make/rose-app.conf $RSUITE/app/jules/rose-app.conf &`

Tip: it's the last of these four textfiles that contains all the parameters and options for the JULES run.

When editing JULES suites, all parameters are explained on the JULES manual pages so I recommend to keep that open at the same time too (there is a search box there for you to find any parameter you are unfamiliar with):

http://jules-lsm.github.io/latest/namelists/contents.html



JULES From Scratch: Rose suites

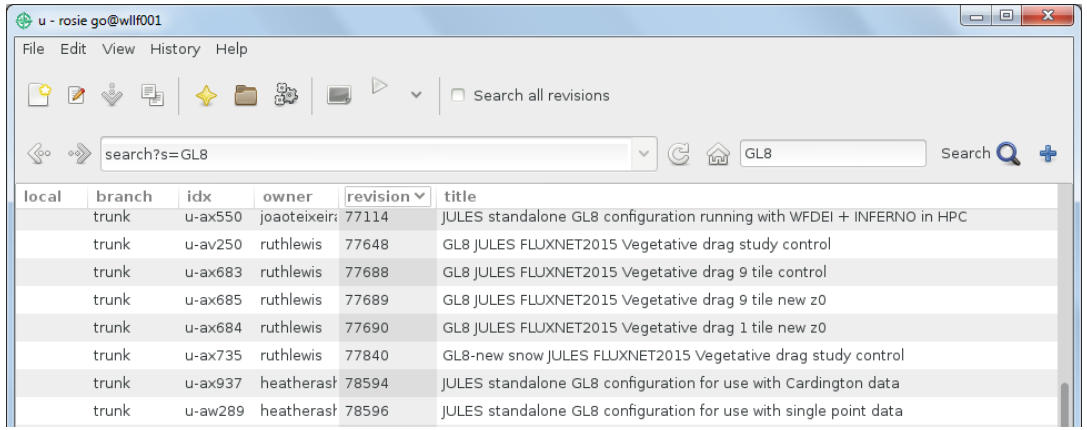
See information on https://jules.jchmr.org/content/getting-started#control-files for how to get hold of Rose suites. There are basically two ways:

1. Download a suite from one of the (many) standard configurations on https://jules.jchmr.org/content/configurations

2. On UNIX using the *Rose Suite Discovery Engine*: `rosie go &`

(select Edit → the 'u' data source, then use the search box). To download, choose:

- **Checkout Suite** to take a duplicate of it
- or **Copy Suite** to use it for your own work (you'll get a new suite ID and this will belong to you).



The *Rosie Go* window. *Rosie Go* expects a directory ~/roses/ to be on your system, which it uses as a store for downloaded Rose suites.

Every suite has its own URL based on its idx code (e.g. *u-am539*) and you can use this to see the changes to the suite over time, e.g.

https://code.metoffice.gov.uk/trac/roses-u/browser/a/m/5/3/9

(to see a changeset, click on the red number (not the cog) in the **Rev** column → View changes).
Equivalently, go to https://code.metoffice.gov.uk/rosie/u/, put an idx code in the search box, find the right suite and then click "../" once to go up to 'trunk'.

JULES From Scratch: Rose suites

It's important to remember that Rose suites are NOT version-independent and are also NOT platform-independent: for example, if you have downloaded a suite suitable for JULESvn5.1 that ran on a Univ. Durham's computer, you will need to modify it to have it run for JULESvn5.2 on a Univ. Exeter's computer (or JASMIN, or ARCHER, etc.) even if all the science settings are identical.

Apart from very tiny suites (e.g. my examples in this tutorial), suites also do NOT contain the driving and ancillary files they need to run (they just point to them), so if you want to use that suite from Durham you will also generally have to download these files separately (by which I mean email the suite authors and ask for copies).

Details of the modification steps are on this page:

https://jules.jchmr.org/content/modify-suite

JULES From Scratch

We're mostly through *JULES From Scratch* now. Let's recap:

- We started with **installation**, starting at the page https://jules.jchmr.org/content/scratch, which got you set up with *Cylc*, *Rose* and *FCM* installed and also JULES itself (each has their own page describing the installation steps).

- **Rose suites** came next https://jules.jchmr.org/content/rose-suites, what they are and where to get them. These are the control files for a JULES run and details of the essential modification steps you need to go through to get a downloaded suite working on your system are here: https://jules.jchmr.org/content/modify-suite

See my 'getting started' page https://jules.jchmr.org/content/getting-started for the other files you need like driving data, ancillary files (and links to analysis tools are there too because you will need to plot out the results of your runs too).

Let's go through **a worked example** ...

JULES From Scratch: Worked example of a point run

Here's a point run using the old Loobos example that used to be included in JULES:

- If you were going to use standard JULESvn6.0, you would download it something like this:
  - cd ~/MODELS
  - fcm co fcm:jules.x_tr@vn6.0 jules-vn6.0
  - export JULES_ROOT=$HOME/MODELS/jules-vn6.0

- However, I would like you NOT to do that but to use a slightly modified version of vn6.0 called vn6.0_hj:
  - cd ~/MODELS
  - fcm co https://code.metoffice.gov.uk/svn/jules/main/branches/dev/tobymarthews/vn6.0_hj vn6.0_hj
  - export JULES_ROOT=$HOME/MODELS/vn6.0_hj

- Download suite u-cg242 from *Rosie Go* and modify the paths inside it to point to your own profile rather than mine, i.e.
  - Search-replace path /home/users/tmarthews to whatever is your home directory on JASMIN.
  - export RSUITE=$HOME/roses/u-cg242

- You can see that I have a number of suites in my ~/roses/ directory:

```
[tmarthews@cylc1 roses]$
[tmarthews@cylc1 roses]$ pwd
/home/users/tmarthews/roses
[tmarthews@cylc1 roses]$ ls
nlists  u-bq564  u-bu747  u-bu748  u-bu749  u-cf638  u-cf682  u-cg242
[tmarthews@cylc1 roses]$
```

JULES From Scratch: Worked example of a point run

Now run the model. There are two ways to do this - with or without Rose:

- RUNNING OUTSIDE ROSE (this is NOT the advised way to do things, but is easier):

  - module load jaspy
  - cd $JULES_ROOT
  - fcm make -j 2 -f etc/fcm-make/make.cfg --new
  - cd ~/roses
  - mkdir nlists
  - cd nlists
  - rose app-run -i -C $RSUITE/app/jules
  - export NAMELIST=$HOME/roses/nlists
  - alias jrun="$JULES_ROOT/build/bin/jules.exe $NAMELIST"
  - jrun

A stream of [INFO] lines like this indicate JULES is running:

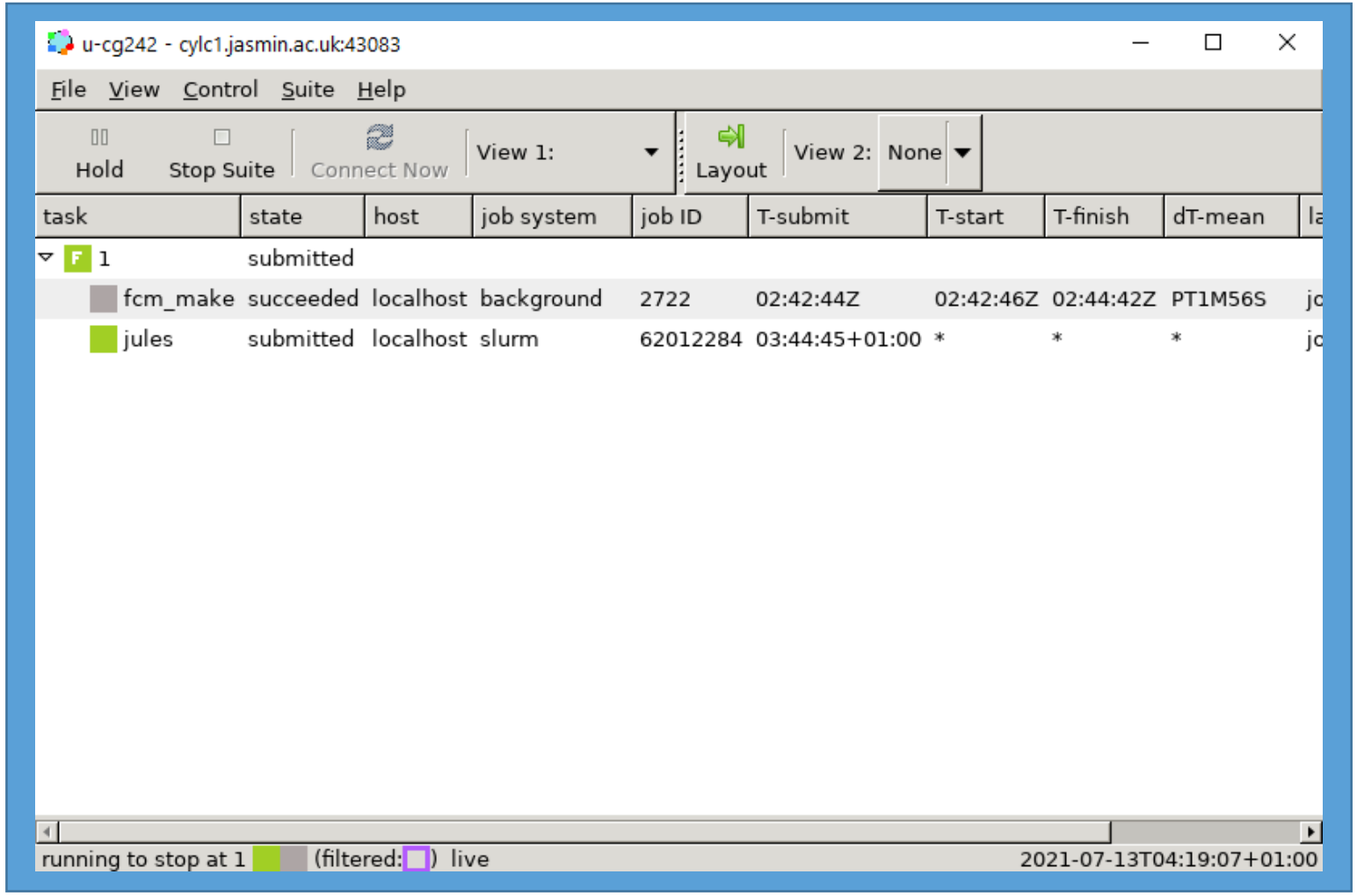
```
[INFO] next_time: Timestep: 743; Started at: 1997-01-16 10:00:00
[INFO] next_time: Timestep: 744; Started at: 1997-01-16 10:30:00
[INFO] next_time: Timestep: 745; Started at: 1997-01-16 11:00:00
[INFO] next_time: Timestep: 746; Started at: 1997-01-16 11:30:00
[INFO] next_time: Timestep: 747; Started at: 1997-01-16 12:00:00
[INFO] next_time: Timestep: 748; Started at: 1997-01-16 12:30:00
[INFO] next_time: Timestep: 749; Started at: 1997-01-16 13:00:00
[INFO] next_time: Timestep: 750; Started at: 1997-01-16 13:30:00
[INFO] next_time: Timestep: 751; Started at: 1997-01-16 14:00:00
[INFO] next_time: Timestep: 752; Started at: 1997-01-16 14:30:00
[INFO] next_time: Timestep: 753; Started at: 1997-01-16 15:00:00
[INFO] next_time: Timestep: 754; Started at: 1997-01-16 15:30:00
[INFO] next_time: Timestep: 755; Started at: 1997-01-16 16:00:00
```

- RUNNING THROUGH ROSE (this IS the advised way and you get the Cylc GUI as described above):

  - rose edit -C $RSUITE &
  - Ignore the red warning triangles and click the Play button in Rose Edit  .

- If that all works, try a modified run. Change the duration of the run (Timestepping namelist) to just the first 15 days of June 1997 (midnight is always at the start of the day, so this means main_run_end should be 1997-06-16 00:00:00). Does it re-run?

JULES From Scratch: Worked example of a point run



JULES From Scratch: Worked example of a gridded run

Here's a grid run using WATCH data that you should be able to follow if you can download the Met data files from https://www.tobymarthews.com/jules-short-course-2016.html :

- Download the ancil files, unzip and save them all in a directory called **ancils/** on your system.
- Download the driving data files (all the other .zip files), unzip and put them all in a directory called **WFD-EI-Forcing/** (i.e. you should end up with several files each in their own directory like /.../WFD-EI-Forcing/Tair_WFDEI_land/Tair_WFDEI_land_200306.nc).
- Check out a copy of JULES, using any version you like between vn5.1 and vn5.6.
- Check out any version of my "Global run at 0.5° resolution" suite from the end of https://jules.jchmr.org/content/scratch (get it from *Rosie Go*) and save it in **~/roses/** on your system. That suite is set up for the UKCEH linux boxes, so you need to modify it: go carefully through all the 'modify suite' steps on https://jules.jchmr.org/content/modify-suite to get it working on your platform, including:
    - Search-replace path /users/global/tobmar/MODELS/iofiles/io_wetlands/ancils/ to the absolute path of **ancils/** on your system.
    - Search-replace path /prj/nceo/WFD-EI-Forcing/ to the absolute path of **WFD-EI-Forcing/** on your system.
- Open the suite in Rose Edit. Change the output directory (Output namelist) /users/global/tobmar/MODELS/iofiles/io_riv/output1 to a directory (of your choice) on your system (check it exists).
- Click the Play button in Rose Edit.
- Finally, if that all works, try a modified run. Change the duration of the run (Timestepping namelist) to the first 15 days of June 2003 (midnight is always at the start of the day, so this means main_run_end should be 2003-06-16 00:00:00) and turn OFF spin-up. Does it re-run?

JULES From Scratch: Troubleshooting

While you have that example running:

- Check that you can right-click on jules → View → job stdout to check on progress (this file holds the `[INFO]` lines JULES produces during execution).

- If you close the Cylc GUI accidentally, type `cylc gscan &` and double-click on your Rose suite job to reopen it. In the same way, you can reopen a suite you've left running e.g. overnight.

- A new directory ~/cylc-run/ will appear on your system when the run starts (if it is not there already). When you run a rose suite, a copy of the whole suite is put in this directory at location $CSUITE defined like this ($CSUITE is what I call the suite's 'run directory' described on https://metomi.github.io/rose/doc/html/tutorial/rose/suites.html#suite-directory-vs-run-directory):

```
wllf001 ~ $ export CSUITE=$HOME/cylc-run/${RSUITE##*/}
wllf001 ~ $ echo $CSUITE
```

- Note that Rose suites run independently of the session you're in (Cylc suites run as daemons) so you gain nothing by opening three separate shells and initiating three runs in each rather than running three in the same session.
- If you find that the *fcm_make* step works, but it hangs when it moves on to the *jules* app, check that you have correctly added the JULES command set to $PATH (see https://jules.jchmr.org/content/get-jules).

JULES From Scratch: Troubleshooting

If you want to abort/stop a suite:

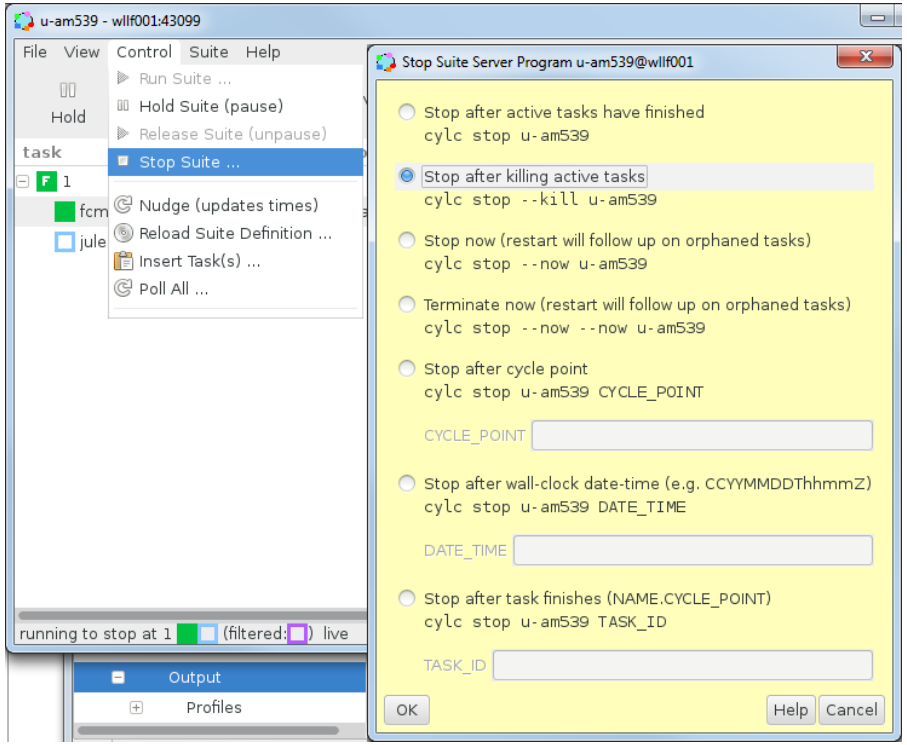
DON'T JUST CLOSE OFF THE CYLC GUI WINDOW (the one with the): the suite will still continue in the background and will prevent you starting another run of the same suite.

DON'T CLICK THE BIG STOP SUITE BUTTON: Surprisingly, this *doesn't* stop the suite **: Cylc will actually wait for all submitted / running jobs to complete (during which it says "stopping"), and only when they've completed will it stop the suite. I feel it really should be called a 'Complete submitted jobs & exit' button (but that's just my opinion)

If you want to actually kill/stop your suite, you need to go into the menus like I'm showing in the screenshot right and choose 'Stop after killing active tasks', which will do the equivalent of the command:

```
cylc stop ${RSUITE##*/} --kill
```

If that still doesn't work, try NCAS's advice at http://cms.ncas.ac.uk/wiki/RoseCylc/Hints#Problemsshuttingdownsuites .



** In *Cylc* terminology, the suite is just the calling structure for a set of jobs, so this button does 'stop the suite' in the sense that no further jobs will be initiated, but it doesn't stop any jobs that have already been called (see http://metomi.github.io/rose/doc/html/cheat-sheet.html).

JULES From Scratch: Viewing the log files

Rather than printing progress information to the screen (e.g. as `[INFO]` lines) and any execution errors, when JULES is run through Rose/Cylc these two are diverted into two textfiles called ‘log files’ stored in the directory ~/cylc-run/ . Progress information goes into a file called **job.out** and any errors go to a file called **job.err**, which you can open from the command line like this:

```
more $CSUITE/log/job/1/jules/NN/job.out
more $CSUITE/log/job/1/jules/NN/job.err
```

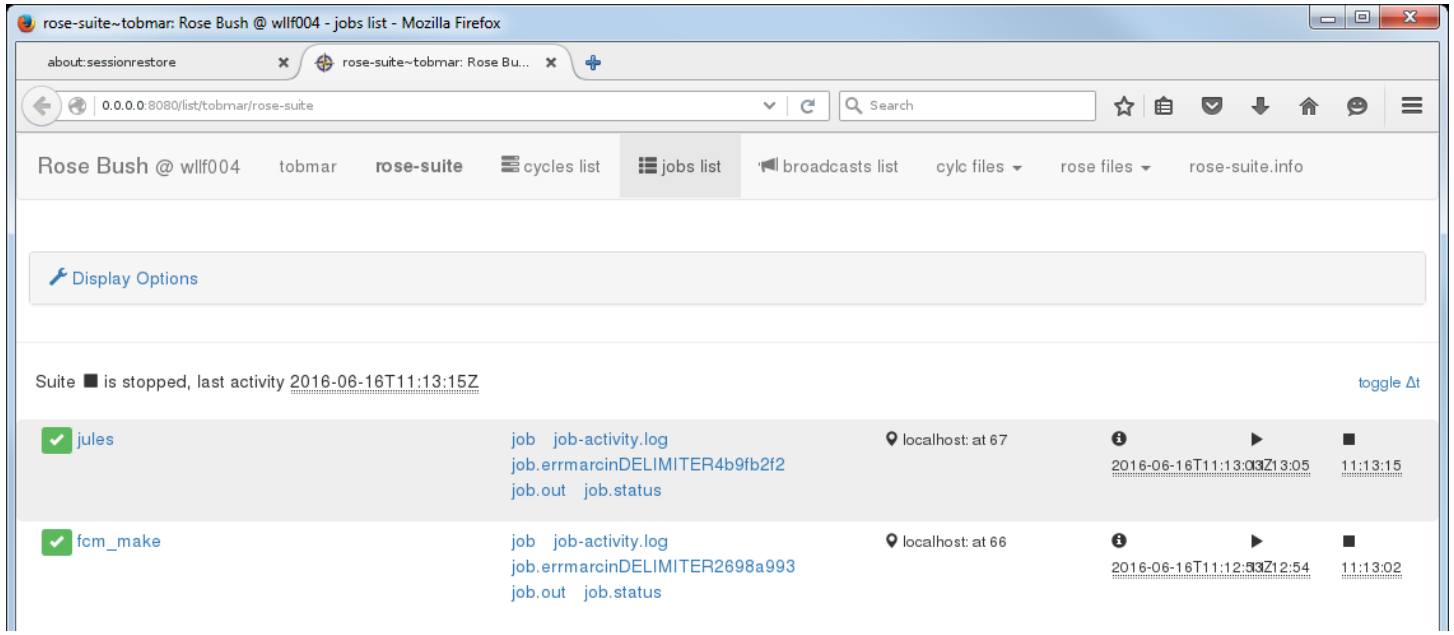
Or, as mentioned above, you can also open these two log files through the Cylc GUI by right-clicking on the 'jules running' line → View → [job stdout or job stderr]

If your job succeeds, note that the Cylc GUI will always clear the job away and you won’t be able to open these files with the mouse any more, however, which I find a bit annoying (the files are still accessible from the command line, though).

JULES From Scratch: Viewing the log files

A more user-friendly way of viewing these logs that works on most systems is to use **Rose Bush** (n.b. not installed on JASMIN):

```
rose slv --name=${RSUITE##*/}
```



- On JASMIN note that Rose Bush is <u>not installed</u>.
- On some systems this command can take 10-15 sec to open: on Monsoon, a workaround is to launch firefox on xcslcX and go to http://localhost/rose-bush/ (see http://cms.ncas.ac.uk/wiki/RoseCylc/Hints#CantviewoutputinRosebush)
- Rose Bush also has a 'history' feature and will show you the logs of previous runs. This is useful, but note that if you ever delete the ~/cylc-run/ directory from your system then you will wipe this history too (I sometimes clear ~/cylc-run/ because it's really just a scratch directory and it sometimes gets very large).

JULES From Scratch: Known bug in *gfortran*

Before we go any further, please be aware of a bug that affects the *gfortran* compiler. This bug affects JULES versions ≥5.2 if you have *gfortran* version after approx. vn4.9.x * (check this using `gfortran --version`). If this is a problem, edit this file:

```
nedit $JULES_ROOT/etc/fcm-make/compiler/gfortran.cfg &
```

On line 17 change the “$fflags_common = -std=f2003 -fall-intrinsics -fmax-identifier-length=63 \” to “$fflags_common = -fall-intrinsics -fmax-identifier-length=63 \” (i.e. just remove the “-std=f2003”). That's all you need to do for this bug.

* In JULES revisions since early 2018 a new compiler flag has been introduced to ensure compatibility with F2003 standards (see https://code.metoffice.gov.uk/trac/jules/wiki/ticket/711/TicketDetails). Unfortunately, this triggers a bug in versions of gfortran after approx. vn4.9.x , meaning that it cannot read namelist files properly, which produces errors like

[FATAL ERROR] …: Error reading namelist JULES_... (IOSTAT=5010 IOMSG=Cannot match namelist object name …)

JULES From Scratch: Help resources

I suggest to bookmark the following webpages if you are ever dealing with JULES:

JULES website: https://jules.jchmr.org/

JULES TRAC: https://code.metoffice.gov.uk/trac/jules/

JULES online manual: http://jules-lsm.github.io/latest/namelists/contents.html

Example Rose suite: https://code.metoffice.gov.uk/trac/roses-u/log/b/p/6/9/6

JULES Tickets: https://code.metoffice.gov.uk/trac/jules/query

JULES Code Branches: https://code.metoffice.gov.uk/trac/jules/browser/main/branches/dev

Finally, **YOU ARE NOT ALONE** in your labours and trials with JULES, Rose, Cylc and FCM ...

Please do use the support email lists/groups or, if you don't feel these are fit for your purposes (too high-brow, too technical, unfriendly, simply scary ...) then create your own and I'll add it to the list on https://jules.jchmr.org/content/getting-started#email-and-support-lists .

Thank you very much!

JULES

Joint UK Land Environment Simulator

JULES From Scratch: Rose stem tests

If you are working at an official SITE (which is just a place where the IT staff have been willing to go through the appropriate steps to set up as a SITE: see *Setting up a new site* on https://code.metoffice.gov.uk/trac/jules/wiki/WaysToRunJules) then KGO data will be installed ("Known Good Output") and if that's there then you can run the *Rose Stem tests* like this:

```
cd jules-vn6.0
rose stem --group=all --source=. --new
```

To save typing, add this line to your ~/.bashrc file and you'll be able to abbreviate it to just "rstem" instead of that command:

```
alias rstem='rose stem --group=all --source=. --new'
```