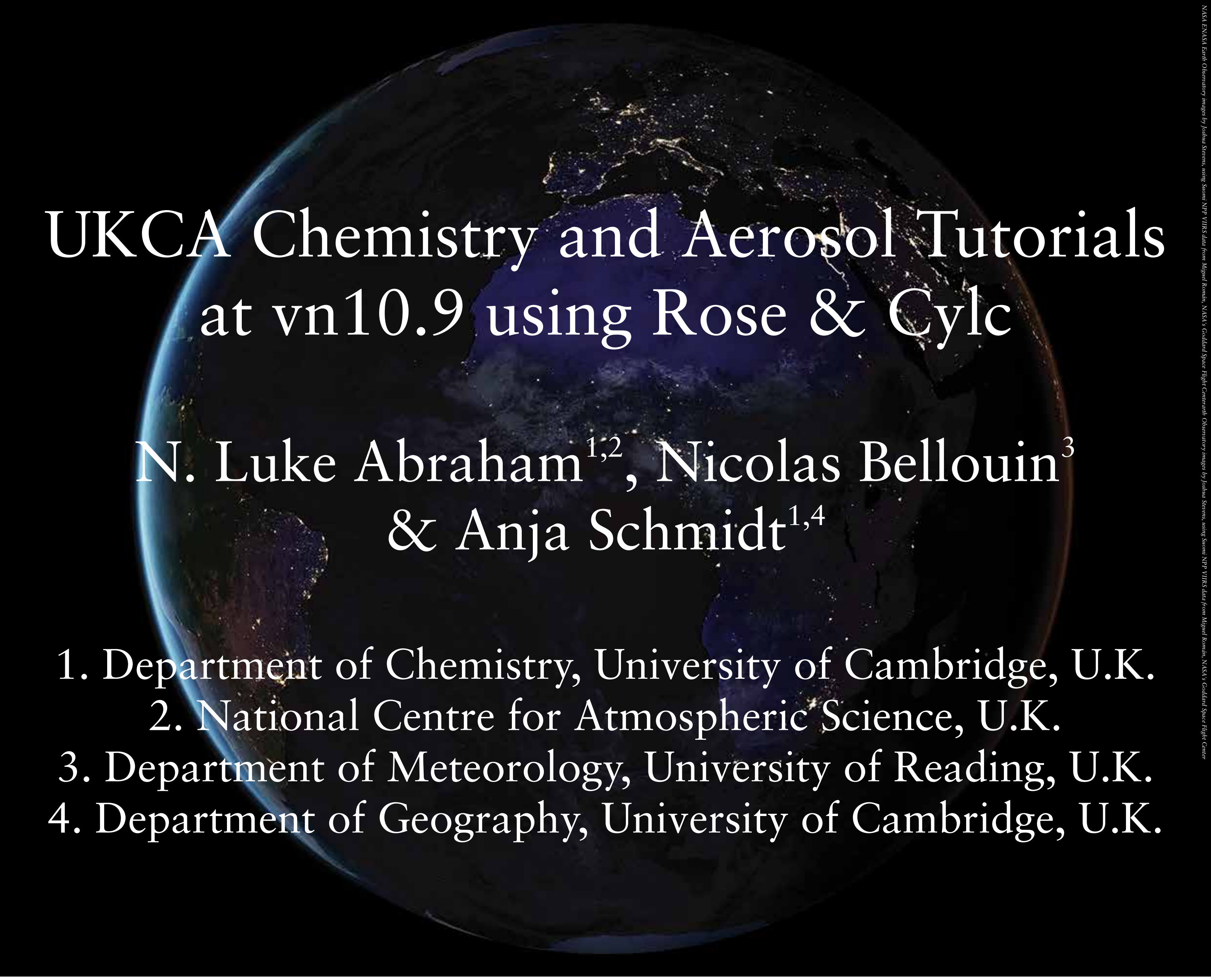


NASA ENASA Earth Observatory images by Joshua Stevens, using Suomi NPP VIIRS data from Miguel Román, NASA's Goddard Space Flight Centerarth Observatory images by Joshua Stevens, using Suomi NPP VIIRS data from Miguel Román, NASA's Goddard Space Flight Center

UKCA Chemistry and Aerosol Tutorials at vn10.9 using Rose & Cylc

N. Luke Abraham[1,2], Nicolas Bellouin[3]
& Anja Schmidt[1,4]

1. Department of Chemistry, University of Cambridge, U.K.
2. National Centre for Atmospheric Science, U.K.
3. Department of Meteorology, University of Reading, U.K.
4. Department of Geography, University of Cambridge, U.K.

https://doi.org/10.17863/CAM.22153

www.ukca.ac.uk/wiki/index.php/UKCA_Chemistry_and_Aerosol_Tutorials_at_vn10.9

November 2017

Acknowledgements
We would like to thank Alex Archibald, Scott Archer-Nicolls, David Wade, Ewa Bednarz, Mohit Dalvi, Paul Cresswell, David Matthews, Ben Johnson, John Pyle, and the team at NCAS Computational Modelling Services (cms.ncas.ac.uk) for their help and support in the creation of these tutorials. This work used the ARCHER UK National Supercomputing Service (www.archer.ac.uk). This work was supported by the NERC Advanced Training Short Courses scheme, grant number NE/P020089/1.

UNIVERSITY OF CAMBRIDGE
Department of Chemistry

NERC SCIENCE OF THE ENVIRONMENT

NERC ADVANCED TRAINING

National Centre for Atmospheric Science
NATURAL ENVIRONMENT RESEARCH COUNCIL

UKCA Chemistry and Aerosol Tutorials at vn10.9

From UKCA

UKCA Chemistry and Aerosol Tutorials at vn10.9 using Rose & Cylc

The following tutorials will teach you how to use various aspects of UKCA, and the parts of Rose (the Unified Model's python-based user interface) that are specific to UKCA.

- ### Things to know before you start

General use of the Rose and UKCA:

1. Running existing UKCA Suite
2. Exploring Rose
3. What is STASH?
4. Adding new chemical tracers
5. Adding new emissions
6. Adding new chemical reactions
7. Adding dry deposition of chemical species
8. Adding wet deposition of chemical species
9. Adding new UKCA chemical diagnostics
10. Examining Aerosol Impacts
11. Worked Example: Developing a change for submission to the trunk

These tutorials are an update and expansion of the tutorials which were at UM8.2, UM8.4, and UM10.4.

This tutorial has been funded by the NERC Advanced Training: Short Courses award scheme (http://www.nerc.ac.uk/funding/available/postgrad/advanced/atsc/) and the ACITES Atmospheric Chemistry Modelling Network (http://www.ncas.ac.uk/index.php/en/what-is-acites) .

Written by Luke Abraham 2018

Retrieved from "http://www.ukca.ac.uk/wiki/index.php?title=UKCA_Chemistry_and_Aerosol_Tutorials_at_vn10.9&oldid=6912"

- This page was last modified on 18 April 2018, at 15:07.

UKCA Chemistry and Aerosol Tutorials: Things to know before you start the vn10.9 practicals

From UKCA

UKCA Chemistry and Aerosol Tutorials at vn10.9

You will need to have the details of your PUMA and ARCHER accounts to hand.

Contents

Before you begin the UKCA Tutorials

These tutorials assume that you are familiar with FCM, and training on this can be found here:

http://cms.ncas.ac.uk/wiki/Fcm

The UM FCM tutorial that is linked from this page uses the old user interface (the *"UMUI"*), so will be less relevant here.

You should also familiarise yourself with Rose & Cylc, and more information on these can be found here:

http://cms.ncas.ac.uk/wiki/RoseCylc

The Met Office provide some on-line UM specific Rose training here:

https://code.metoffice.gov.uk/doc/um/latest/um-training/index.html

Also, if you do not have a PUMA or ARCHER account, you can use the Met Office provided Virtual Machine to run the VM, provided here:

https://github.com/metomi/metomi-vms

You should follow the instructions in UMDP X10 (https://code.metoffice.gov.uk/doc/um/latest/umdp.html) .

It is possible to run a simple UKCA suite in the VM, but note that this requires at least 6GB of RAM to run successfully.

Accounts

To run these tutorials the minimum you will need is

- an account on the Met Office Science Repository Service (https://code.metoffice.gov.uk)

which will then allow you to run the tutorials in a Virtual Machine.

To run these on ARCHER you will need

- a PUMA account
- an ARCHER account

For information on how to get these, please see the Getting Started with UKCA page.

If you are attending the face-to-face training workshop you will be provided with ARCHER training accounts, although you still need your own PUMA account.

Developing code for the UM

During this tutorial you will be creating tickets on the Met Office Science Repository Service (https://code.metoffice.gov.uk) , and making code changes to UM branches. Please note that **this is a *live* system**, used by scientists all over the world to develop their code. However, **please don't worry about breaking anything**, as there are practices in place to prevent this.

For more information about developing for the UM, please see the Working Practices for UM Development with Rose, FCM and trac (https://code.metoffice.gov.uk/trac/um/wiki/working_practices) .

UKCA Training Suite

Machine	UM Version/Configuration	Suite ID	Owner	Comments
ARCHER	vn10.9 N48L38 Intel Compiler	**u-as101**	lukeabraham	May be compiled manually (recommended)
vm	vn10.9 N48L38 GNU Compiler	**u-as159**	lukeabraham	1x2 decomposition
XC40	vn10.9 N48L38 GNU Compiler	**u-au234**	lukeabraham	4x9 decomposition (36-core)

These suites are based on the **`um_n48_ukca_eg`** app, which is available in the MetUM trunk (in `rose-stem/app/`). A ticket giving more details on these (and the code changes) can be found at: um:#3578 (https://code.metoffice.gov.uk/trac/um/ticket/3578) .

To set-up using the UM with Rose & Cylc correctly, go to

- http://cms.ncas.ac.uk/documents/training/December2017/UM_practicals/getting-setup.html

and follow the instructions from **Section 1.2** onwards if you are using Linux/Mac (after opening a terminal).

Note: when entering MOSRS password, a rose window will open behind everything all your other windows. You will also need to enter your username here to complete the process.

When you have completed this, you should launch the Rose browser by typing

```
rosie go &
```

and there you should search for training suite and make a copy of it.

When you have copied the suite, you will need to remember to change the username & tic code in *suite conf* → *Machine Options*.

Additional Requirements (Virtual Machine)

Tutorial Data

To be able to do Tutorials 5 and 10 you will need emissions data and python scripts. These files, as well as worked solutions for all the Rose and code changes can be obtained by running the following command from the `home/` directory of your VM:

```
wget -q --show-progress http://gws-access.ceda.ac.uk/public/ukca/UKCA_Tutorial_vn109.tgz
```

There is 880MB to download, so it might take some time.

Once the download has finished, you can extract the archive by running

```
tar -xzvf UKCA_Tutorial_vn109.tgz
```

This will make a directory called `Tutorial/` containing 1.4GB of files. If you `cd` into this you can check that the files have not been corrupted by running the following command:

```
md5sum --status -c vn109.md5
```

Xconv

You will need to download Xconv (http://cms.ncas.ac.uk/documents/xconv/) (`xconv1.93`) from here:

- http://cms.ncas.ac.uk/documents/xconv/_downloads/xconv1.93_linux_x86_64.tar.gz

You can download this by

```
wget http://cms.ncas.ac.uk/documents/xconv/_downloads/xconv1.93_linux_x86_64.tar.gz
```

Download it to your $HOME/bin on the VM, cd into this directory, `tar -zxvf` the tar-ball, and then

```
ln -s xconv1.93 xconv
ln -s xconv1.93 convsh
```

Iris

There is an **`install-iris`** script provided, but you will need to set-up modules yourself to be able to use it properly. The anaconda install breaks Rose if put in your PATH, however, there is now an alias

```
conda
```

which will open a new terminal with all the anaconda python packages in its $PATH. This will allow you to use Rose in one terminal and Iris in another.

I have found a handy way to use python is to use `ipython` (https://ipython.org/) with the following arguments

```
ipython --pylab --logfile=ipython-`date +\"%Y%m%d-%H%M%S\"`.py"
```

I have aliased this to

```
pylab
```

in my `.bashrc`. The

```
--pylab
```

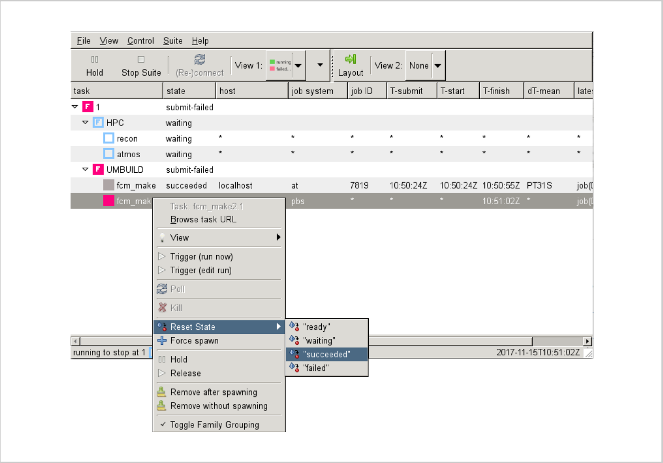
sets-up a MatLab-type environment (numpy, scipy, matplotlib all loaded using standard shortcuts), and

```
--logfile=ipython-`date +\"%Y%m%d-%H%M%S\"`.py
```

means that all commands are saved to a file of the format

```
ipython-YYMMDD-HHMMSS.py
```

Manual Compiling (ARCHER)



task	state	host	job system	job ID	T-submit	T-start	T-finish	dT-mean	lates
F 1	submit-failed								
F HPC	waiting								
recon	waiting	*	*	*	*	*	*	*	*
atmos	waiting	*	*	*	*	*	*	*	*
F UMBUILD	submit-failed								
fcm_make	succeeded	localhost	at	7819	10:50:24Z	10:50:24Z	10:50:55Z	PT31S	job(
fcm_mak			pbs	*	*	*	10:51:02Z	*	job(

running to stop at 1

2017-11-15T10:51:02Z

Gcylc window showing the fcm_make2_um task in the submit-failed state and resetting it.

The UKCA Training Suite has been altered slightly to allow it to be easily manually compiled. This can be selected as an option in **suite conf** ⟶ **UM hosts**. With this set to **true**, when the suite runs the ***fcm_make2*** task will enter the ***submit-failed*** state. Then, on **ARCHER**, you should then change directory into

```
$HOME/cylc-run/[SUITE-ID]/log/job/1/fcm_make2/NN
```

and compile the `um-atmos` and `um-recon` executables by typing

```
./job
```

This will take around 7-12 minutes to compile. When it has finished successfully you will get the following message in the terminal, just before the prompt:

```
JOB SCRIPT EXITING (TASK SUCCEEDED)
```

You can follow the progress of this process in detail by `tail`-ing the following file

```
$HOME/cylc-run/[SUITE-ID]/share/fcm_make/fcm-make2.log
```

Once this step has completed, you can send the rest of the suite off by going back to the Gcylc window and **right-clicking on the *fcm_make2*** task and click **Reset State** ⟶ **"succeeded"**. This will then allow the rest of the tasks to start in sequence.

Note: manually compiling in this way means that you will no longer have the *job.out* or *job.err* files. You should instead look in the *fcm-make2.log* file.

Scripts

There are a number of scripts provided for these tutorials. These can be found on GitHub here:

- https://github.com/theabro/ukca

They are also available on ARCHER.

UKCA Chemistry and Aerosol Tutorials at vn10.9

Written by Luke Abraham 2017.

Retrieved from "http://www.ukca.ac.uk/wiki/index.php?title=UKCA_Chemistry_and_Aerosol_Tutorials:_Things_to_know_before_you_start_the_vn10.9_practicals&oldid=6875"

- This page was last modified on 9 March 2018, at 10:41.

UKCA Chemistry and Aerosol vn10.9 Tutorial 1

From UKCA

UKCA Chemistry and Aerosol Tutorials at vn10.9

Before starting these tutorials you should first make sure that you have completed the setup instructions (http://cms.ncas.ac.uk/documents/training/November2016/UM_practicals/getting-setup.html) . If you are using Linux or macOS you will only need to complete section 2.2 onwards.

Contents

Copying and Running an Existing Rose Suite

Machine	UM Version/Configuration	Suite ID
ARCHER	vn10.9 N48L38 Intel Compiler	**u-as101**
vm	vn10.9 N48L38 GNU Compiler	**u-as159**
XC40	vn10.9 N48L38 GNU Compiler	**u-au234**

If you are using **PUMA & ARCHER**, you will need to login to PUMA, e.g.

```
ssh -Y username@puma.nerc.ac.uk
```

If you are using the **Met Office Virtual Machine**, you will need to login to the VM, e.g.

```
vagrant ssh
```

You should then be asked for your SRS password.

Then launch the UM graphical user interface by:

```
rosie go
```

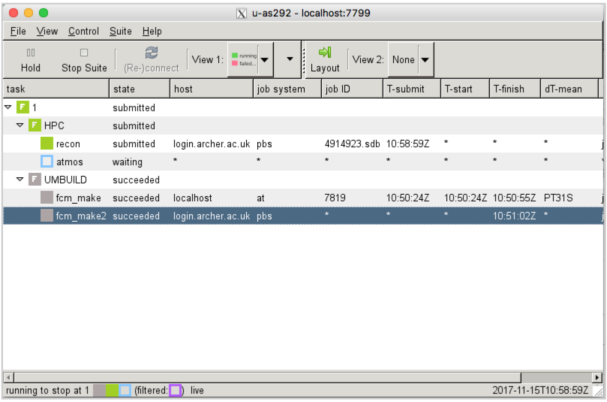
This should then load up a blank interface. Go to **Edit** ⟶ **Data source** and select **u**. The go to the search panel and search for the correct base suite. This will then show all the suites that have this suite-id in it's history or title. You should just select the correctly named suite and not any of the others.

Right-click on the suite and click **copy suite**. A new box will open asking for the project - it is fine to press **Forward** here. On the next panel, it's fine to just press **OK** again as well. You should ***avoid*** the **checkout suite** option, as this will give you a copy of someone else's suite, and it might be possible for you to make changes to it that could affect other users.

The suite will now copy and checkout to your `/home/$USER/roses` directory, and will also appear when running `rosie go`.

You should now right-click ⟶ edit (or double-click) on this new suite, and click **run**, which is symbolised by a *play* symbol (i.e., a large grey arrow-head pointing to the right).

If you are using PUMA/ARCHER, this suite has been configured to **manually compile**, so you should follow the instructions on how to do that, to be sure that you can compile your suite as quickly as possible. It will take about 7 minutes to compile the code (*fcm_make* and *fcm_make2* tasks), followed by about 3 minutes to run the reconfiguration step (*recon*), and then 1 minute to run the UM itself (*atmos*).



task	state	host	job system	job ID	T-submit	T-start	T-finish	dT-mean
1	submitted							
HPC	submitted							
recon	submitted	login.archer.ac.uk	pbs	4914923.sdb	10:58:59Z	*	*	*
atmos	waiting	*	*	*	*	*	*	*
UMBUILD	succeeded							
fcm_make	succeeded	localhost	at	7819	10:50:24Z	10:50:24Z	10:50:55Z	PT31S
fcm_make2	succeeded	login.archer.ac.uk	pbs	*	*	*	10:51:02Z	*

running to stop at 1 (filtered:) live 2017-11-15T10:58:59Z

GCylc running on the VM.

If you are using the VM, times will vary depending on the specifications of the host. It could take about 10 minutes to compile the code (*fcm_make*) for the first time (and about 1-2 minutes when recompiling), followed by about 1 minute to run the reconfiguration step (*recon*), and then about 12 minutes to run the UM itself (*atmos*).

When the suite has finished successfully it will then become blank with the message **stopped with 'succeeded'** in the bottom-left corner.

Version Control

Rose suites are all held under version control, using fcm (http://cms.ncas.ac.uk/wiki/Fcm) . When making changes to a suite, you will need to **save** it before you can **run** the suite. Once you are happy with the settings, you can also **commit** these changes back to the repository - to do this change directory to the

```
/home/$USER/roses/[SUITE-ID]
```

and then type

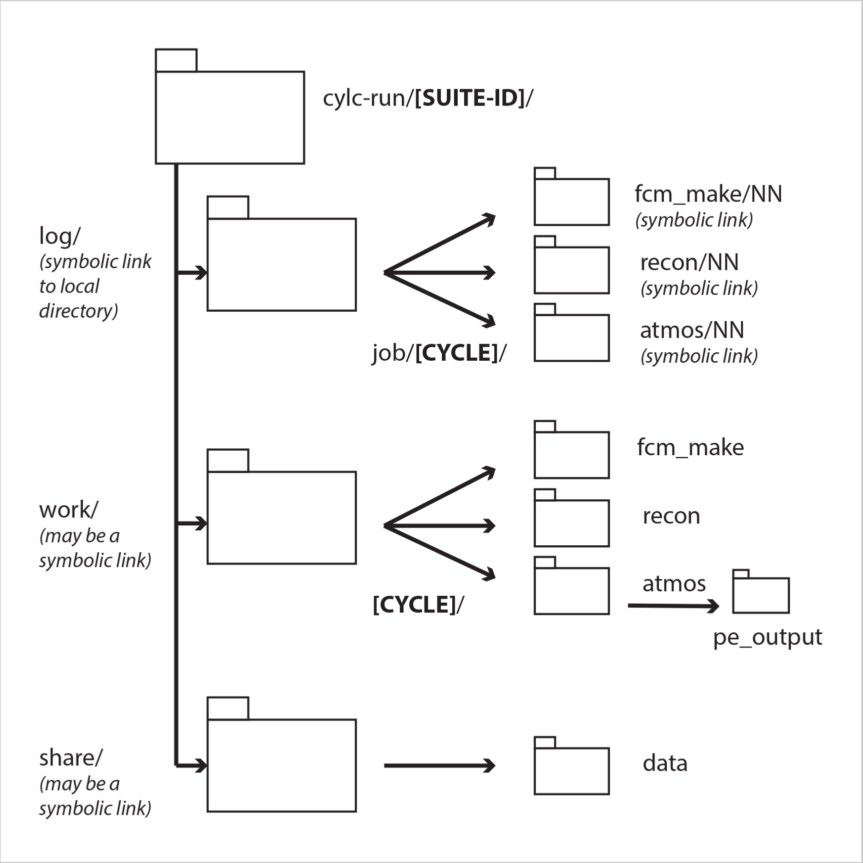
```
fcm commit
```

a text editor will then open, and you should type a short message describing what the changes you have made do. You should then close the editor and type **y** in the terminal. It is recommended that you commit frequently (even on configurations that aren't working) as this protects you against mistakes and accidental deletions etc.

These suites can be viewed on the SRS here: https://code.metoffice.gov.uk/trac/roses-u (password required) (https://code.metoffice.gov.uk/trac/roses-u)

It is recommended that you commit your suites regularly.

Output Directory Structure



Simplified cylc-run directory structure. **[SUITE-ID]** will be e.g. ***u-as101***, and **[CYCLE]** is the cycle-point, e.g. ***1*** or ***19880901T0000Z***.

The output directory structure of rose suites are rather complex. The schematic on the right gives a broad overview of the general structure. Everything can be found within the `[SUITE-ID]` (e.g. *u-as101* etc.) directory, which can be found within your `$HOME/cylc-run` directory. On the VM, everything in held in this single place, but things are more complicated when using PUMA & ARCHER, as some files are held on the ARCHER */home* filesystem, some on the */work* filesystem, and some files are synced back to PUMA as well.

Within this directory there are several directories, including:

- **`log`**: a symbolic link to a directory called something like `log.20171214T141332Z`, which contains all the job.out (containing output from UM `WRITE` statements from the `umPrint` subroutine) and job.err files etc., as well as the script used to run the job.
- **`work`**: this contains the directories used when the job actually runs. Real-time output will be sent to files here (held in a `pe_output` directory. Sometimes model output will also be here. On ARCHER this directory exists on the `/work` filesystem.
- **`share`**: This isn't used by the UKCA Training Suite, but for climate jobs model output will usually be held in this directory. On ARCHER this directory exists on the `/work` filesystem.

Within these directories there will be many sub-directories. Some of these will be named from the **cycle-point** (labelled `[CYCLE]` in the graphic. For the UKCA Training Suite this will be **1**, but under normal climate operation this will be a date string, e.g. **19880901T0000Z**. The sub-directories will (eventually) be named after the **app** (labelled by `[JOB NAME]` in the table below) that the output is from, e.g. `fcm_make`, `recon`, `atmos` etc.

Files	General Path	Example
Most recent *job.out* files	**ARCHER:** `/home/n02/n02/`**`$USER`**`/cylc-run/`**`[SUITE-ID]`**`/log/job/1/`**`[JOB NAME]`**`/NN` **vm:** `/home/vagrant/cylc-run/`**`[SUITE-ID]`**`/log/job/1/`**`[JOB NAME]`**`/NN`	/home/n02/n02/luke/cylc-run/u-as292/log/job/1/atmos/NN /home/vagrant/cylc-run/u-as297/log/job/1/atmos/NN
Processor output (while running)	**ARCHER:** `/work/n02/n02/`**`$USER`**`/cylc-run/`**`[SUITE-ID]`**`/work/1/`**`[JOB NAME]`**`/pe_output` **vm:** `/home/vagrant/cylc-run/`**`[SUITE-ID]`**`/work/1/`**`[JOB NAME]`**`/pe_output`	/work/n02/n02/luke/cylc-run/u-as292/work/1/atmos/pe_output /home/vagrant/cylc-run/u-as297/work/1/atmos/pe_output
Output files in 64-bit fieldsfile format	**ARCHER:** `/work/n02/n02/`**`$USER`**`/cylc-run/`**`[SUITE-ID]`**`/work/1/atmos` **vm:** `/home/vagrant/cylc-run/`**`[SUITE-ID]`**`/work/1/atmos`	/work/n02/n02/luke/cylc-run/u-as292/work/1/atmos /home/vagrant/cylc-run/u-as297/work/1/atmos

Notes

- The `job.out` files can also be viewed through the Gcylc GUI right-click menu from each job.
- A handy command to check the progress of the `atmos` job is

  ARCHER: `tail -1000f atmos.fort6.pe00 | grep Atm_Step`
  vm: `tail -1000f atmos.fort6.pe0 | grep Atm_Step`

- Other suites may put the output fieldsfiles into a different location, e.g.

  `/work/n02/n02/`**`$USER`**`/cylc-run/`**`[SUITE-ID]`**`/share/data/History_Data`

  and there are also post-processing settings that can be used to copy data to the RDF or JASMIN as the suite runs.

- Output fieldsfiles can have various naming conventions. For these suites, they will be **atmosa.pa19810901_00**, but more generally they are likely to be of the form

  `[SHORT SUITE-ID]``a.p`**`[abcdefghijkmsyx]`**`YYYYMMDD`

  e.g. ag308a.pk19880901

  `[SHORT SUITE-ID]``a.p`**`[abcdefghijkm]`**`YYYYmon`

  e.g. ak468a.pe1989oct

- Climate-mean files (**`.pm`**) will often also have seasonal (**`.ps`**), annual (**`.py`**), and decadal (**`.px`**) equivalents.

Whilst this suite is running, take a look at Tutorial 2: Exploring Rose.

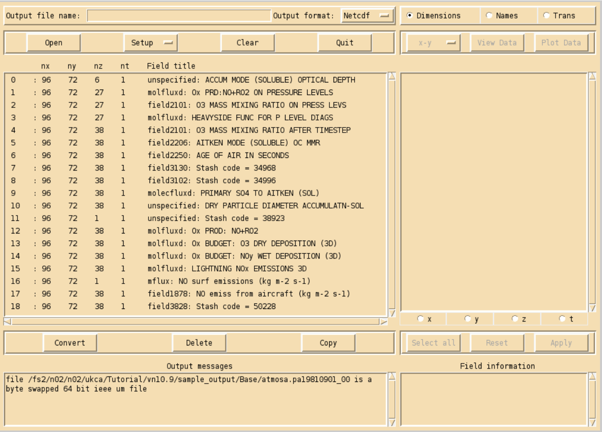
Viewing Output

For more detailed plotting, tools such as the Iris (http://scitools.org.uk/iris/) and cf-python (http://cfpython.bitbucket.org/) libraries can be used to view UM file formats directly. However, for quick viewing, Xconv (http://cms.ncas.ac.uk/documents/xconv/) is a very useful tool. Information on how to install Xconv on the VM can be found here.

To view these files, do

```
xconv atmosa.pa19810901_00
```

As well as viewing files, you can use Xconv to convert these files to netCDF, by filling in the ***Output file name:*** box (e.g. *foo.nc*), and then clicking **convert**. If no path is defined, this will save the file in the same directory that you opened Xconv from.



```
        nx    ny    nz    nt    Field title
0     : 96    72    6     1     unspecified: ACCUM MODE (SOLUBLE) OPTICAL DEPTH
1     : 96    72    27    1     molfluxd: Ox PRD:NO+RO2 ON PRESSURE LEVELS
2     : 96    72    27    1     field2101: O3 MASS MIXING RATIO ON PRESS LEVS
3     : 96    72    27    1     molfluxd: HEAVYSIDE FUNC FOR P LEVEL DIAGS
4     : 96    72    38    1     field2101: O3 MASS MIXING RATIO AFTER TIMESTEP
5     : 96    72    38    1     field2206: AITKEN MODE (SOLUBLE) OC MMR
6     : 96    72    38    1     field2250: AGE OF AIR IN SECONDS
7     : 96    72    38    1     field3130: Stash code = 34968
8     : 96    72    38    1     field3102: Stash code = 34996
9     : 96    72    38    1     molecfluxd: PRIMARY SO4 TO AITKEN (SOL)
10    : 96    72    38    1     unspecified: DRY PARTICLE DIAMETER ACCUMULATN-SOL
11    : 96    72    1     1     unspecified: Stash code = 38923
12    : 96    72    38    1     molfluxd: Ox PROD: NO+RO2
13    : 96    72    38    1     molfluxd: Ox BUDGET: O3 DRY DEPOSITION (3D)
14    : 96    72    38    1     molfluxd: Ox BUDGET: NOy WET DEPOSITION (3D)
15    : 96    72    38    1     molfluxd: LIGHTNING NOx EMISSIONS 3D
16    : 96    72    1     1     mflux: NO surf emissions (kg m-2 s-1)
17    : 96    72    38    1     field1878: NO emiss from aircraft (kg m-2 s-1)
18    : 96    72    38    1     field3828: Stash code = 50228
```

```
file /fs2/n02/n02/ukca/Tutorial/vn10.9/sample_output/Base/atmosa.pa19810901_00 is a
byte swapped 64 bit ieee um file
```

Example output from the UKCA training suite.

Example output from the UKCA training suite can be found at **`/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Base/atmosa.pa19810901_00`** on ARCHER.

Checklist

☐ List suites using `rosie go`

☐ Copy suites using the right-click menu

☐ Run suites using the play button

Tutorial 2

Written by Luke Abraham 2017

Retrieved from "http://www.ukca.ac.uk/wiki/index.php?title=UKCA_Chemistry_and_Aerosol_vn10.9_Tutorial_1&oldid=6755"

- This page was last modified on 18 January 2018, at 12:20.

UKCA Chemistry and Aerosol vn10.9 Tutorial 2

From UKCA

UKCA Chemistry and Aerosol Tutorials at vn10.9

Contents

Exploring Rose

If you are familiar with the UMUI, you will see that Rose is quite different. As well as a complete re-write of the user interface, a lot of the namelists have been streamlined and also other things, such as how STASHmaster files are read (for user diagnostics etc.), have also been changed.

Take a look at your suite, and click through the options. There are 2 key sections that you will need to consider:

fcm_make

Also called often called fcm_make_um.

This section controls which fcm branches are used by the model, and also what compiler settings are used etc. You should go to **fcm_make** ⟶ **env** ⟶ **Sources** to see what branches are being used (although there are none used by the UKCA training suite). During the course of these tutorials you will make a branch and add it here to complete the tutorial tasks. You can add both relative paths to the repository (starting with `branches/`) and absolute paths to a working copy (starting with `/home/`).

um

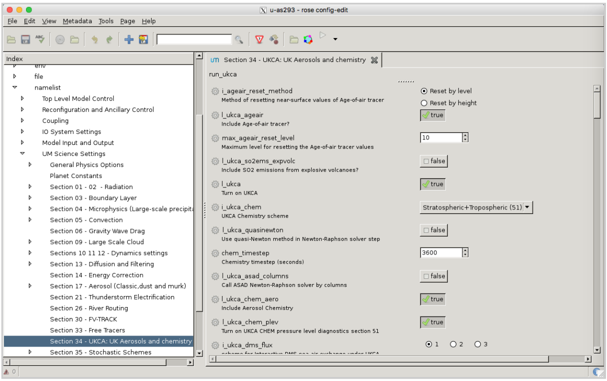
Click the arrow by **um** to view the list of sections. We'll be most interested in the **namelist** section, which contains all the settings that configure the science that will be done in this configuration of the UM, including the UKCA-specific options, and also when adding more output. We will mostly be concerned with 2 sections:

STASH Requests

Go to **um** ⟶ **namelist** ⟶ **Model Input and Output** ⟶ **STASH Requests and Profiles** ⟶ **STASH Requests** to view all the model output that has been requested by this suite. This list can be quite extensive, and we will cover adding more output in Tutorial 3: What is STASH?. During these tutorials this will be know as **the STASH panel**.

Section 34 - UKCA: UK Aerosols and Chemistry

Go to **um** ⟶ **namelist** ⟶ **UM Science Settings**. This contains all the different science sections, and by clicking on each panel you can see the settings for each. If you click on **Section 34 - UKCA: UK Aerosols and Chemistry** you will be able to see the UKCA settings. During these tutorials this will be known as **the UKCA panel**.


u-as293 - rose config-edit
File
Edit
View
Metadata
Tools
Page
Help
Index
env
file
namelist
Top Level Model Control
Reconfiguration and Ancillary Control
Coupling
IO System Settings
Model Input and Output
UM Science Settings
General Physics Options
Planet Constants
Section 01 - 02 - Radiation
Section 03 - Boundary Layer
Section 04 - Microphysics (Large-scale precipita
Section 05 - Convection
Section 06 - Gravity Wave Drag
Section 09 - Large Scale Cloud
Sections 10 11 12 - Dynamics settings
Section 13 - Diffusion and Filtering
Section 14 - Energy Correction
Section 17 - Aerosol (Classic,dust and murk)
Section 21 - Thunderstorm Electrification
Section 26 - River Routing
Section 30 - FV-TRACK
Section 33 - Free Tracers
Section 34 - UKCA: UK Aerosols and chemistry
Section 35 - Stochastic Schemes
um Section 34 - UKCA: UK Aerosols and chemistry
0

run_ukca

i_ageair_reset_method
Method of resetting near-surface values of Age-of-air tracer
● Reset by level ○ Reset by height

l_ukca_ageair
Include Age-of-air tracer?
✓ true

max_ageair_reset_level
Maximum level for resetting the Age-of-air tracer values
10

l_ukca_so2ems_expvolc
Include SO2 emissions from explosive volcanoes?
□ false

l_ukca
Turn on UKCA
✓ true

i_ukca_chem
UKCA Chemistry scheme
Stratospheric+Tropospheric (51)

l_ukca_quasinewton
Use quasi-Newton method in Newton-Raphson solver step
□ false

chem_timestep
Chemistry timestep (seconds)
3600

l_ukca_asad_columns
Call ASAD Newton-Raphson solver by columns
□ false

l_ukca_chem_aero
Include Aerosol Chemistry
✓ true

l_ukca_chem_plev
Turn on UKCA CHEM pressure level diagnostics section 51
✓ true

i_ukca_dms_flux
● 1 ○ 2 ○ 3

Figure 1: The UKCA panel in Rose.

In Rose there is a 1:1 correspondence between the name of the variable in Rose, and the name of the variable in the namelist and hence the UM. This means that it can be easy to find what variable you might be interested in by first finding the appropriate variable in Rose (e.g. l_ukca_ageair). By the name of each variable is a cog - if you click this you can view **info** about the variable, which lists all the information the Rose has been given about it, e.g.

```
Data
name  i_ageair_reset_method
value  1
comments
error
flags
ignored_reason
warning

Metadata
compulsory  true
description  Method of resetting near-surface values of Age-of-air tracer
full_ns  /um/namelist/UM Science Settings/section34
help  There are two options available for resetting the near-surface values
 of the age-of-air tracer to 0.0:
 1. based on maximum level upto which to reset
 2. based on maximum height upto which to reset
 If 'by level' selected,
 recommended value is 10 for consistency
 with older UKCA configurations
 If 'by height' selected,
 the standard value is 2000.0 metres
id  namelist:run_ukca=i_ageair_reset_method
sort-key  a09
trigger  namelist:run_ukca=max_ageair_reset_level: 1 ;
namelist:run_ukca=max_ageair_reset_height: 2 ;
value-titles  Reset by level,Reset by height
values  1,2
```

You can also click **help**, which lists the help text associated with each variable. If you are not sure about a variable, looking at the info can be very informative. Due to the way the change process of the UM works, all variables must have information, such as the above, supplied, and any new science that has been added also needs to be added to each sections documentation paper.

search

Unlike the UMUI, there is a search box in Rose. It works best if you know the name of a variable, although it is still very useful to find options if you're not sure where they are.

Please take some time to browse through the rose suite, and get a feel for where things are. If you previously used the UMUI, you might want to try to locate comparable panels to see how things have changed.

Checklist

- □ Make use of search to find what you need in the suite.

Tutorial 3

Written by Luke Abraham 2017

Retrieved from "http://www.ukca.ac.uk/wiki/index.php?title=UKCA_Chemistry_and_Aerosol_vn10.9_Tutorial_2&oldid=6707"

- This page was last modified on 3 January 2018, at 11:27.

UKCA Chemistry and Aerosol vn10.9 Tutorial 3

From UKCA

UKCA Chemistry and Aerosol Tutorials at vn10.9

When using the UM, it is likely that you will want to add new diagnostics, and we will cover writing out chemical fluxes in a later tutorial. Here we will cover selecting new diagnostics in STASH.

Contents

Task 3.1: add new output

TASK 3.1: Output 3-hourly-mean UKCA CO (section 34 item 010) field to the **UPA** output stream, and then add-in the equivalent diagnostic on pressure levels using the DP27CCM domain.

Hint [hide]
You will need to remember to run the TidyStashTransform macro once you have added the diagnostics in the STASH panel.

In the following page you will learn more about how the Unified Model's output is organised and managed. You should use this information to complete the task above. A worked solution is provided at the end of the tutorial.

What is STASH?

STASH is the Unified Model's storage handling and diagnostic system. It is designed to cope with the many different configurations that the UM can be used in, but still provide output in a consistent and standard way. A full technical description of STASH can be found in **Unified Model Documentation Paper C4** which can be downloaded from the Met Office Science Repository Service (password required) (https://code.metoffice.gov.uk/doc/um/latest/umdp.html) . You may also want to look through the UM STASH help here: https://code.metoffice.gov.uk/doc/um/latest/um-training/stashmaster.html .

Prognostic and Diagnostic Fields

The UM considers variables (or *fields*) to be of two different types, *prognostic* or *diagnostic*. **Prognostic** fields are ones which the model must have values for, prior to each timestep, as e.g. the equations of motion the model solves require these fields (these are fields such as specific humidity or potential temperature), so they must exist in the model start dump. **Diagnostic** fields are all other fields that are derived from prognostic ones, and as such the model does not need to have prior values for these. Ancillary files (such as emissions, SSTs etc) contain prognostic fields. UKCA requires around 137 prognostic fields in the GA7+StratTrop configuration, and could in theory provide almost 2000 diagnostic fields on model levels, many of which have pressure-level equivalents.

From a user's perspective, STASH is used to output fields during the run, and from this point of view it does not matter if these are prognostic or diagnostic fields. However, you will need to consider these differences when you add new chemical tracers.

STASH Sections and Items

Each field that is considered by STASH has a unique address which is given by a **section** and an **item** number. Prognostic fields are mostly held in section 0 (with the exception of tracers) and diagnostics are organised by areas of the code, e.g. short-wave radiation diagnostics are held in section 1, long-wave radiation diagnostics are held in section 2 etc. Some sections will always be on, and some sections will only be on if a certain process is selected, e.g. the interactive land-surface scheme. Each section can hold up to 999 items, where each item is a separate prognostic or diagnostic field, and can be either 2D or 3D.

Each field has its own entry in a **STASHmaster file**. There is a master list of fields which is held in the *STASHmaster_A* file, a copy of which is located in each branch at

```
rose-meta/um-atmos/vnX.Y/etc/stash/STASHmaster/STASHmaster_A
```

If you make changes to this, to add your own diagnostics, you will need to make changes to the

```
rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster_A
```

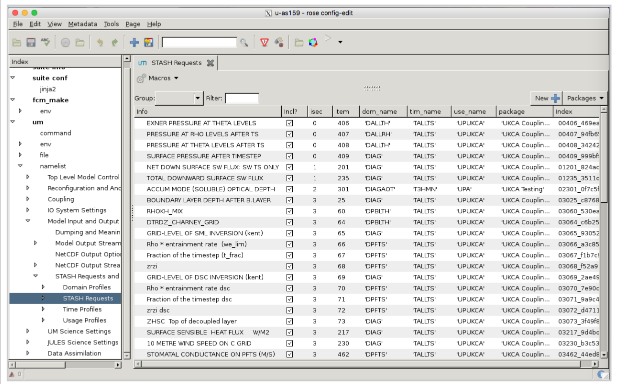
file, and then change where Rose looks for the **metadata** associated with your suite. This will be covered in more detail when you add new chemical tracers.

The term *HEAD* here refers to fact that this is the latest metadata of the branch (c.f. the *"head"* of an FCM branch being the latest version of it). For any new branch at a particular model version (e.g. vn10.9) the STASHmaster_A file in *HEAD* and e.g. *vn10.9* will (initially) be the same. When the branch (at e.g. vnX.Y) is committed to the trunk the changes in *HEAD* are then merged into the trunk to become part of the vnX.Y+0.1 metadata.

STASH Panel

The STASH panel can be found at **um** ⟶ **namelist** ⟶ **Model Input and Output** ⟶ **STASH Requests and Profiles** ⟶ **STASH Requests**.

It is usually intially sorted by section/item numbers, but you can change the ordering by clicking on the items on the bar at the top (e.g. **use_name**). This can be helpful to see which items are going to which output stream.



Info	Incl?	isec	item	dom_name	tim_name	use_name	package	Index
EXNER PRESSURE AT THETA LEVELS	☑	0	406	'DALLTH'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	00406_469ea
PRESSURE AT RHO LEVELS AFTER TS	☑	0	407	'DALLRH'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	00407_94fb6
PRESSURE AT THETA LEVELS AFTER TS	☑	0	408	'DALLTH'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	00408_34242
SURFACE PRESSURE AFTER TIMESTEP	☑	0	409	'DIAG'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	00409_999bf
NET DOWN SURFACE SW FLUX: SW TS ONLY	☑	1	201	'DIAG'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	01201_824ac
TOTAL DOWNWARD SURFACE SW FLUX	☑	1	235	'DIAG'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	01235_3511c
ACCUM MODE (SOLUBLE) OPTICAL DEPTH	☑	2	301	'DIAGAOT'	'T3HMN'	'UPA'	'UKCA Testing'	02301_0f7c5f
BOUNDARY LAYER DEPTH AFTER B.LAYER	☑	3	25	'DIAG'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03025_c8768
RHOKH_MIX	☑	3	60	'DPBLTH'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03060_530ea
DTRDZ_CHARNEY_GRID	☑	3	64	'DPBLTH'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03064_c6b25
GRID-LEVEL OF SML INVERSION (kent)	☑	3	65	'DIAG'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03065_93052
Rho * entrainment rate (we_lim)	☑	3	66	'DPFTS'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03066_a3c85
Fraction of the timestep (t_frac)	☑	3	67	'DPFTS'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03067_f1b7c
zrzi	☑	3	68	'DPFTS'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03068_f52a9
GRID-LEVEL OF DSC INVERSION (kent)	☑	3	69	'DIAG'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03069_2ae4
Rho * entrainment rate dsc	☑	3	70	'DPFTS'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03070_7e90c
Fraction of the timestep dsc	☑	3	71	'DPFTS'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03071_9a9c4
zrzi dsc	☑	3	72	'DPFTS'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03072_d4711
ZHSC Top of decoupled layer	☑	3	73	'DIAG'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03073_3f49f
SURFACE SENSIBLE HEAT FLUX W/M2	☑	3	217	'DIAG'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03217_9d4bc
10 METRE WIND SPEED ON C GRID	☑	3	230	'DIAG'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03230_b3c53
STOMATAL CONDUCTANCE ON PFTS (M/S)	☑	3	462	'DPFTS'	'TALLTS'	'UPUKCA'	'UKCA Couplin...	03462_44ed

Figure 1: The STASH panel in Rose.

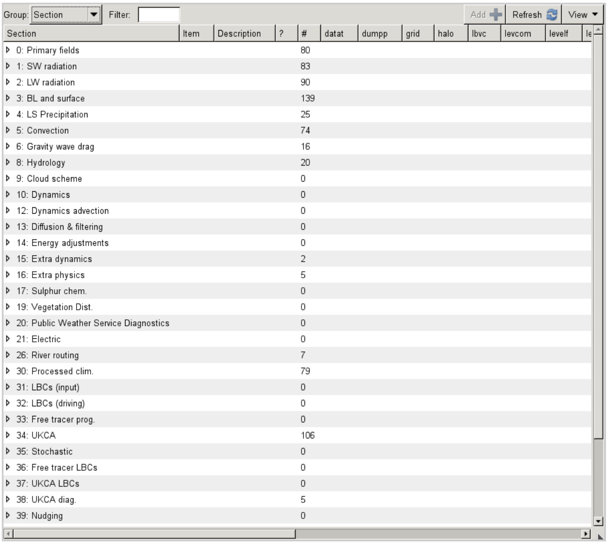
To add new output, click the **New** button (with the large plus symbol on it), and this will open the *Add new STASH requests panel*. This lists all the available STASH output by section. You can scroll down to the section you want, click the small arrow on the left, scroll down to the diagnostic that you want and either right-click on it and click **Add STASH request**, or select it and then click the **Add** button in the top right-hand corner of the panel.

Another option, especially if there is already a STASH request that has similar domain, time, and usage profiles, is to right-click the existing request, and select

Once this has been added to the main STASH panel, you will get a message, similar to **Added request as namelist: streq(1)**. You will also find that a red warning triangle has appeared in the rose editor. This will disappear once you fill-in the *dom_name*, *tim_name*, and *use_name* entries for it. This can be done by clicking next to the red X that is in each column and selecting the correct value from the drop-down menu.

Once you have selected entries for each usage field, you will need to run the **TidyStashTransform** macro. When the STASH entry was created it was given the the namelist value *streq(1)*, with the next as *streq(2)* etc.. This is a default value, and the correct namelist index needs to be generated for each STASH item (each is unique to the choices made, which also prevents duplicate entries being created). To do this, click the **Macros** drop-down menu from the top-left of the panel and select the **stashindices.TidyStashTransform** option. This will process the STASH requests, and any entries where the index doesn't match will be listed. Click **OK**, and this will then apply the correct index to the request, which will be the 5-digit STASH number followed by a string of seemingly-random letters and numbers, e.g. *34010_c4b8f1ad*.

It is also possible to run any Rose macro (including *stashindices.TidyStashTransform*) by going to the **Metadata** ⟶ **um** menu at the top of the Rose GUI.



Section	Item	Description	?	#	datat	dumpp	grid	halo	lbvc	levcom	levelf	le
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								
8: Hydrology				20								
9: Cloud scheme				0								
10: Dynamics				0								
12: Dynamics advection				0								
13: Diffusion & filtering				0								
14: Energy adjustments				0								
15: Extra dynamics				2								
16: Extra physics				5								
17: Sulphur chem.				0								
19: Vegetation Dist.				0								
20: Public Weather Service Diagnostics				0								
21: Electric				0								
26: River routing				7								
30: Processed clim.				79								
31: LBCs (input)				0								
32: LBCs (driving)				0								
33: Free tracer prog.				0								
34: UKCA				106								
35: Stochastic				0								
36: Free tracer LBCs				0								
37: UKCA LBCs				0								
38: UKCA diag.				5								
39: Nudging				0								

Figure 1: The new STASH requests panel in Rose.

UKCA STASH sections

UKCA actually has 6 STASH sections - these are:

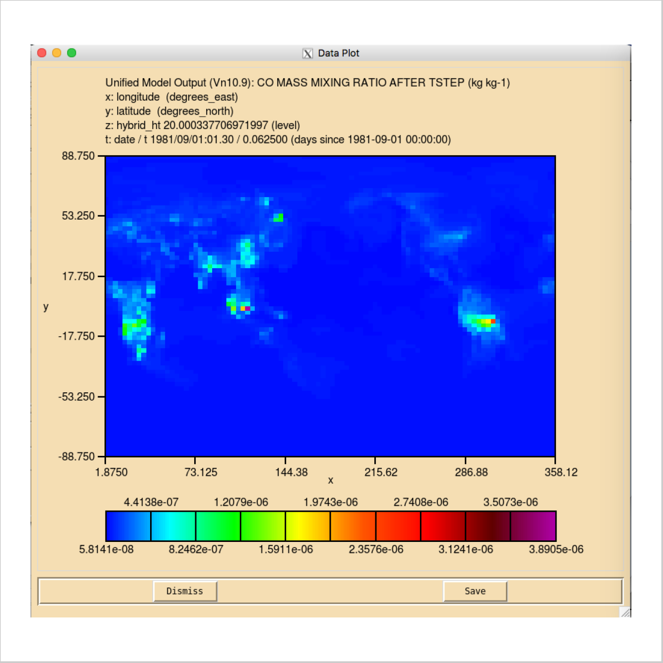
- **Section 34:** This contains the *prognostic* UKCA variables on model theta levels, i.e. all the transported species, and several other fields which are required to be in the restart file.
- **Section 37:** This is for UKCA tracer lateral boundary conditions for use in limited area configurations.
- **Section 38:** This section is for GLOMAP-mode diagnostics on model theta levels.
- **Section 50:** This section is for Chemical diagnostics on model theta levels (or single levels, e.g. surface).
- **Section 51:** This section duplicates all the variables from Section 34, but on pressure levels.

  To use this section `l_ukca_chem_plev` must be set to **true** in the UKCA panel.
  Additionally, s51i999 must also be output, as this is the Heavyside function that is used to mask missing data from the pressure-level fields (as the UM fieldsfile format has missing data set to zero).

- **Section 52:** This section duplicates all the variables from Section 50, but on pressure levels.

  To use this section `l_ukca_asad_plev` must be set to **true** in the UKCA panel.
  Additionally, s51i999 must also be output, as this is the Heavyside function that is used to mask missing data from the pressure-level fields (as the UM fieldsfile format has missing data set to zero).

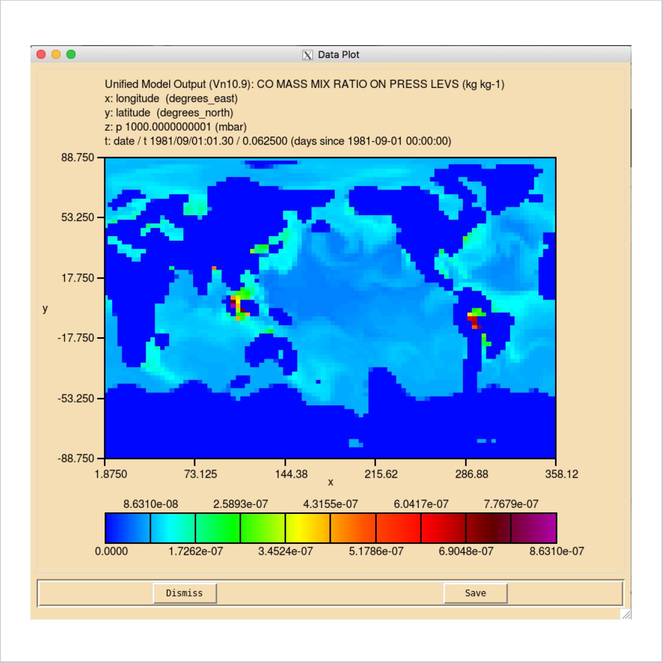
Solution to Task 3.1



Data Plot

Unified Model Output (Vn10.9): CO MASS MIXING RATIO AFTER TSTEP (kg kg-1)
x: longitude (degrees_east)
y: latitude (degrees_north)
z: hybrid_ht 20.000337706971997 (level)
t: date / t 1981/09/01:01.30 / 0.062500 (days since 1981-09-01 00:00:00)

Surface (20m) UKCA CO (x-y)



Data Plot

Unified Model Output (Vn10.9): CO MASS MIX RATIO ON PRESS LEVS (kg kg-1)
x: longitude (degrees_east)
y: latitude (degrees_north)
z: p 1000.0000000001 (mbar)
t: date / t 1981/09/01:01.30 / 0.062500 (days since 1981-09-01 00:00:00)

1000hPa UKCA CO (x-y)

You were given the task

- *Output 3-hourly-mean UKCA CO (section 34 item 010) field to the **UPA** output stream, and then add-in the equivalent diagnostic on pressure levels using the DP27CCM domain*

and were given the hints

- *You will need to remember to run the TidyStashTransform macro once you have added the diagnostics in the STASH panel.*

For a working Rose suite that has completed this task, please see

- **ARCHER:** `u-as292@59777`
- **vm:** `u-as297@59775`

The specific Rose changes made are:

- **ARCHER:** https://code.metoffice.gov.uk/trac/roses-u/changeset/59777/a/s/2/9/2/trunk
- **vm:** https://code.metoffice.gov.uk/trac/roses-u/changeset/59775/a/s/2/9/7/trunk

ARCHER:

```
Index: app/um/rose-app.conf
===================================================================
--- app/um/rose-app.conf        (revision 59739)
+++ app/um/rose-app.conf        (revision 59777)
@@ -3287,6 +3287,14 @@
 tim_name='T3HMN'
 use_name='UPA'

+[namelist:umstash_streq(34010_c4b8f1ad)]
+dom_name='DALLTH'
+isec=34
+item=10
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
 [namelist:umstash_streq(34106_ebb7bb67)]
 dom_name='DALLTH'
 isec=34
@@ -3407,6 +3415,14 @@
 tim_name='T3HMN'
 use_name='UPA'

+[namelist:umstash_streq(51010_5e3d3bef)]
+dom_name='DP27CCM'
+isec=51
+item=10
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
 [namelist:umstash_streq(51999_8b73e62f)]
 dom_name='DP27CCM'
 isec=51
```

vm:

```
Index: app/um/rose-app.conf
===================================================================
--- app/um/rose-app.conf        (revision 59757)
+++ app/um/rose-app.conf        (revision 59775)
@@ -3879,6 +3879,14 @@
 tim_name='T3HMN'
 use_name='UPA'

+[namelist:umstash_streq(34010_c4b8f1ad)]
+dom_name='DALLTH'
+isec=34
+item=10
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
 [namelist:umstash_streq(34106_ebb7bb67)]
 dom_name='DALLTH'
 isec=34
@@ -3999,6 +4007,14 @@
 tim_name='T3HMN'
 use_name='UPA'

+[namelist:umstash_streq(51010_5e3d3bef)]
+dom_name='DP27CCM'
+isec=51
+item=10
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
 [namelist:umstash_streq(51999_8b73e62f)]
 dom_name='DP27CCM'
 isec=51
```

These differences can be found here:

- **PUMA:** `/home/ukca/Tutorial/vn10.9/worked_solutions/Task3.1/Task3.1_rose.patch`

If you open the **.pa** file in Xconv, you should see the following additional fields:

```
3    : 96    72    27    1     CO MASS MIX RATIO ON PRESS LEVS
6    : 96    72    38    1     CO MASS MIXING RATIO
```

Sample output from this task can be found at **`/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task3.1/atmosa.pa19810901_00`** on ARCHER.

Checklist

□ Add new STASH diagnostic requests in: um ⟶ namelist ⟶ Model Input and Output ⟶ STASH Requests and Profiles ⟶ STASH Requests.

□ Select the appropriate domain, time, and usage profiles for the new diagnostic(s).

□ Run the TidyStashTransform macro.

□ If outputting UKCA pressure-level diagnostics, remember to make sure that `l_ukca_chem_plev` and (possibly) `l_ukca_asad_plev` are set to True.

Tutorial 4

Written by Luke Abraham 2017

Retrieved from "http://www.ukca.ac.uk/wiki/index.php?title=UKCA_Chemistry_and_Aerosol_vn10.9_Tutorial_3&oldid=6748"

- This page was last modified on 8 January 2018, at 17:36.

UKCA Chemistry and Aerosol vn10.9 Tutorial 4

From UKCA

UKCA Chemistry and Aerosol Tutorials at vn10.9

Contents

What you will do in this tutorial

In this tutorial you will learn how to make the required changes in Rose and in UKCA to include new chemical tracers. As you learn how to do this you will add two new tracers to the UKCA tutorial suite you have already been running.

Task 4.1: Add new tracers into Rose and UKCA

TASK4.1: Add in two new tracers in to slots 64 and 65 for the StratTrop chemistry scheme. The tracer in slot 64 will be called **ALICE** and the tracer in slot 65 will be called **BOB** and initialise these tracers to `1.000000e-12`. You should also output these two tracers through the **UPA** stream in STASH as 3-hour means. In UKCA, you should set the conversion factor for each of these to 1.0.

Make a branch

As UKCA is a framework, it must have the infrastructure around it to allow it to work, with the infrastructure being the UM itself. In this tutorial you will add in two new chemical tracers. In this past this was done in two steps, but in Rose this must be done in a single step from vn10.6 onwards. Even though these tutorials are at vn10.4, we will follow the same steps as for vn10.6.

For this tutorial you will need to make a new branch in the usual way. You should do this now.

You will have learned how to make UM branches in the NCAS-CMS FCM Tutorial (https://puma.nerc.ac.uk/trac/UM_TUTORIAL) . However, now you should use um.x_tr as the source of your branch.

For example, first you should make a ticket on the Met Office SRS Trac pages (https://code.metoffice.gov.uk/trac/um/newticket) (login required) and then make the branch like so

```
fcm branch-create --type dev -k ticket_number your_branch_name fcm:um.x_tr@vn10.9
```

before checking-out your branch by

```
fcm checkout fcm:um.x_br/dev/userid/vn10.9_your_branch_name
```

More information on FCM can be found at the NCAS-CMS FCM pages (http://cms.ncas.ac.uk/wiki/Fcm) or the FCM User Guide (http://metomi.github.io/fcm/doc/user_guide/)

Metadata

In Rose, the GUI appearance is controlled by *metadata*, and it is possible to edit this metadata to add new variables to panels. We will not do this in this tutorial, but you will need to change where Rose looks for metadata, as the STASHmaster_A is also metadata.

Using your new metadata within Rose

To pick-up the metadata changes, you should edit the **meta** path in **um** to point to

```
um-atmos/HEAD
```

From now on, when you want to open your Rose suite for editing, instead of doing it through rosie, you should instead open the job for editing by going to your `$HOME/roses/[SUITE-ID]` directory, and opening it using the command

```
rose edit -M /path/to/vn10.9_your_branch_name/rose-meta
```

If you don't do this, the STASHmaster changes will not be picked up, and you won't be able to select the tracers.

Rose Changes

Include your branch for code changes

Although we haven't made any code changes yet, you will (at some point) need to include your branch in Rose so that the UKCA code changes can be picked-up and compiled. To do this go to **fcm_make** ⟶ **env** ⟶ **Sources** and add a new branch by clicking the **plus symbol** in the **um_sources** section. You should then put

```
/path/to/vn10.9_your_branch_name
```

if you wish to run from a working copy, and

```
branches/dev/[your MOSRS username]/vn10.9_your_branch_name
```

if you want to run from the repository. You can specify specific revision numbers by putting `@REV` at the end of this line (where **REV** is the revision number, e.g. 12345 etc.).

STASHmaster_A

In Rose-based jobs, adding new UKCA tracers is done through editing the STASHmaster_A file directly. Take a look at the **STASHmaster_A** file, found in

```
/path/to/vn10.9_your_branch_name/rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster_A
```

This is a large file, as it contains all the possible STASH items. Scroll down the file until you get to section 34, and you will see how the tracers are defined. The key entries to consider are highlighted in red. All entried are **fixed-width**, so when editing the file, ensure not to change the spacing, otherwise you will get errors.

```
#===============================================================================
# Section 34 UKCA Chemistry
#===============================================================================
#
1|    1 |   34 |    1 |O3 MASS MIXING RATIO AFTER TIMESTEP |
2|    2 |    0 |    1 |    1 |    2 |   40 |   11 |    0 |    0 |    0 |    0 |
3| 000000000000000000000000111110 | 00000000000000000001 |    1 |
4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
5|    0 | 2101 |    0 |   65 |    0 |    0 |    0 |    0 |    0 |
#
```

The meaning of these entries is:

First Line:

- **34:** The *STASH section*, in this case Section 34, UKCA prognostics.
- **1:** The *STASH item*, in this case, the first item in s34, ozone.
- **O3 MASS MIXING RATIO AFTER TIMESTEP:** The *name* of the field in the STASHmaster file. This can be anything descriptive, but it isn't read by the model, and is only considered in Rose (or e.g. Xconv).
-

Second Line:

- **2:** The *Space Code*, which tells the UM whether the field should be held in the restart dump or not. For prognostics, this should be 2 (which means that they should be stored in the restart dump), but for diagnostics that are only passed to output files, this should be 0.

Third Line:

- **000000000000000000000000111110:** These are *Options Codes* which determine which scheme the STASH field (i.e. tracer in this case) are valid for. These are counted from the **right**, ***n1*** to ***n30***, and for UKCA only the first 8 are used. These have the following meaning:
  - **n1:** Age-of-air tracer (*ageair*) or aerosol tracers
  - **n2:** Standard Tropospheric Chemistry scheme using the Backward Euler solver (*trop*)
  - **n3:** Regional Air Quality Chemistry scheme using the Backward Euler solver (*raq*)
  - **n4:** Tropospheric Chemistry with Isoprene scheme using the Newton-Raphson solver (*tropisop*)
  - **n5:** Stratospheric-Tropospheric Chemistry using the Newton-Raphson solver (*strattrop*, also know as *CheST*)
  - **n6:** Standard Stratospheric Chemistry using the Newton-Raphson solver (*strat*)
  - **n7:** Offline-oxidants scheme (to drive GLOMAP-mode) using the Newton-Raphson solver (*offline*)
  - **n8:** Offline-oxidants scheme (to drive GLOMAP-mode) using the Newton-Raphson solver (*offline_be*)

Therefore, looking at the option code for ozone (s34i001), it is valid for the trop, raq, tropisop, strattrop, and strat chemistry schemes.

Fifth Line:

- **2101:** This is the *PP Field Code* and should be unique to each tracer. Here these begin counting at 2101 (s34i001) and end at 2356 (s34i256).

So to make a STASH entry for a single tracer called ALICE in slot s34i064 that is only valid in the StratTrop scheme, the STASHmaster_A entry would look like

```
#
1|    1 |   34 |   64 |ALICE MASS MIXING RATIO AFTER TSTEP |
2|    2 |    0 |    1 |    1 |    2 |   40 |   11 |    0 |    0 |    0 |    0 |
3| 000000000000000000000000010000 | 00000000000000000001 |    1 |
4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
5|    0 | 2164 |    0 |   65 |    0 |    0 |    0 |    0 |    0 |
#
```

Remember: You need to add **2** tracers, ALICE in s34i064 and BOB in s34i065.

You should make these changes to STASHmaster_A, and then **`fcm commit`** these changes. This is important, otherwise ARCHER will not be able to see the changes you have made once you have included your new STASHmaster_A file in Rose.

Section 51

All STASH entries in Section 34 need a corresponding pressure-level field to be added to section 51. The equivalent STASHmaster_A entry for s51 ozone is:

```
#
#===============================================================================
# Section 51 UKCA Chemistry on pressure levels
#===============================================================================
#
1|    1 |   51 |    1 |O3 MASS MIXING RATIO ON PRESS LEVS   |
2|    0 |    0 |    1 |    1 |    3 |    1 |    2 |    0 |    0 |    0 |    1 |
3| 000000000000000000000000111110 | 00000000000000000001 |    3 |
4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
5|    0 | 2101 |    0 |    8 |    0 |    0 |    0 |    0 |    0 |
#
```

You will notice that there are a number of changes to various codes, and this is to do with defining the grid that the diagnostic is valid on, whether it can be held in the dump, etc. For more information on what these codes mean, please see Appendix C in UMDP C4, which can be obtained from the SRS here (password required) (https://code.metoffice.gov.uk/doc/um/latest/umdp.html) . When making a new entry, the easiest thing to do is copy an existing entry and make the required changes to the item, name, PP field code, and option codes.

Remember: You need to add **2** tracers, ALICE in s34i064 and BOB in s34i065.

Help text

You should also add appropriate help text into the **`vn10.4_your_branch_name/rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster-meta.conf`** file, e.g.

```
[stashmaster:code(34064)]
description=ALICE MASS MIXING RATIO AFTER TIMESTEP
help=UKCA Tutiorial Tracer ALICE Mass Mixing Ratio in kg/kg(Air)
```

and

```
[stashmaster:code(51064)]
description=ALICE  MASS MIX RATIO ON PRESS LEVS
help=UKCA Tutiorial Tracer ALICE Mass Mixing Ratio in kg/kg(Air)
    =on pressure levels
```

Remember: You need to add **2** tracers, ALICE in s34i064 and BOB in s34i065.

Use your new STASHmaster file in Rose

Before you make these changes you should first save and close your suite.

To get Rose to recognise your STASHmaster file, you will need to make a number of changes:

1. Point the **um** metadata to **um-atmos/HEAD** and edit Rose using `rose edit -M /path/to/vn10.9_your_branch_name/rose-meta` (you should already be doing this after following the instructions in the metadata section above.
2. You need to open the `$HOME/roses/`**`[SUITE-ID]`**`/app/um/rose-app.conf` file in a text editor (e.g. vim, emacs, nedit etc.) and add the line `STASHMASTER=STASHmaster` inside the **`[env]`** block
3. You need to open the `$HOME/roses/`**`[SUITE-ID]`**`/rose-suite.conf` file in a text editor and add the following lines at the top of the file:

```
[file:app/um/file/STASHmaster]
source=fcm:um.xm_br/dev/[your MOSRS userid]/vn10.9_your_branch_name/rose-meta/um-atmos/HEAD/etc/stash/STASHmaste
```

You need to ensure that you have committed your STASHmaster_A changes, otherwise your new tracer(s) will not be seen at run-time during the reconfiguration or atmosphere steps.

You may wish to now `fcm commit` your suite changes. You can now re-open your suite.

Initialise your new tracer(s)

Your new tracer(s) is a prognostic variable, as therefore the UM requires prior knowledge of the values it has to be able to run properly. To do this you need to go to **um** ⟶ **namelist** ⟶ **Reconfiguration and Ancillary Control** ⟶ **Configure ancils and initialise dump fields**. Here you will see a list of fields. You can change the order of these by e.g. STASH number, as this often makes it easier to view.

To add new fields, right click anywhere within the table and click **Add new section**. This will make a new entry with index **1**, if you add another entry, it will be given index **2** etc.. Right-click on this and click on **View namelist:items(*X*)** (where ***X*** is the index number). This will open a new tab where you can select the STASH section and item for the field you want to initialise using the drop-down menus. Some UKCA tracers need to be given full 3D initial conditions in ancillary file format. However, for this example we will initialise the tracer to a small number, 1.000000×10^{-12}. To do this, select the radio button for **Set to a specified constant value** and put **1.000000e-12** in the dialog box labelled **user_prog_rconst** at the bottom of the panel.

You will need to do this for each of the tracers you are adding (you can also *clone* existing entries). When you have finished, you will need to go to the **STASH Requests** panel (see the STASH tutorial) and run the **TidyStashTransform** macro to correctly generate the required index value for each tracer initialisation.

Output your new tracers

To output your tracers, go to the **STASH panel** and output your tracers as described in the What is STASH? tutorial.

Remember: You need to add **2** tracers, ALICE in s34i064 and BOB in s34i065.

Required UKCA Code Changes

`ukca_nmspec_mod.F90`

Inside UKCA there is a master mapping array that tells the UKCA code which transported species are placed into a particular item number in section 34. At vn10.9 these are contained in the first 256 slots.

This array is called **`nm_spec`**, and it can be found in `ukca_nmspec_mod.F90`. At vn10.9, it looks like this:

```
nm_spec(1:a_max_ukcavars) = (/                                     &
'O3        ','NO        ','NO3       ','NO2       ','N2O5      ',  &
'HO2NO2    ','HONO2     ','H2O2      ','CH4       ','CO        ',  & !10
'HCHO      ','MeOOH     ','HONO      ','C2H6      ','EtOOH     ',  &
'MeCHO     ','PAN       ','C3H8      ','n-PrOOH   ','i-PrOOH   ',  & !20
'EtCHO     ','Me2CO     ','MeCOCH2OOH','PPAN      ','MeONO2    ',  &
'O3_S      ','C5H8      ','ISOOH     ','ISON      ','MACR      ',  & !30
'MACROOH   ','MPAN      ','HACET     ','MGLY      ','NALD      ',  &
'HCOOH     ','MeCO3H    ','MeCO2H    ','H2O       ','ISO2      ',  & !40
'Cl        ','ClO       ','Cl2O2     ','OClO      ','Br        ',  &
'BrO       ','BrCl      ','BrONO2    ','N2O       ','HCl       ',  & !50
'HOCl      ','HBr       ','HOBr      ','ClONO2    ','CFCl3     ',  &
'CF2Cl2    ','MeBr      ','N         ','O(3P)     ','MACRO2    ',  & !60
'MeCl      ','CF2ClBr   ','CCl4      ','CF2ClCFCl2','CHF2Cl    ',  &
'MeCCl3    ','CF3Br     ','H2OS      ','CH2Br2    ','H2        ',  & !70
'DMS       ','SO2       ','H2SO4     ','MSA       ','DMSO      ',  &
'NH3       ','CS2       ','COS       ','H2S       ','H         ',  & !80
'OH        ','HO2       ','MeOO      ','EtOO      ','MeCO3     ',  &
'n-PrOO    ','i-PrOO    ','EtCO3     ','MeCOCH2OO ','MeOH      ',  & !90
'Monoterp  ','Sec_Org   ','SESQUITERP','SO3       ','AROM      ',  &
'O(3P)_S   ','O(1D)_S   ','NO2       ','BrO       ','HCl       ',  & !100
'Nuc_SOL_ND','Nuc_SOL_SU','Ait_SOL_ND','Ait_SOL_SU','Ait_SOL_BC',  &
'Ait_SOL_OC','Acc_SOL_ND','Acc_SOL_SU','Acc_SOL_BC','Acc_SOL_OC',  & !110
'Acc_SOL_SS','Acc_SOL_DU','Cor_SOL_ND','Cor_SOL_SU','Cor_SOL_BC',  &
'Cor_SOL_OC','Cor_SOL_SS','Cor_SOL_DU','Ait_INS_ND','Ait_INS_BC',  & !120
'Ait_INS_OC','Acc_INS_ND','Acc_INS_DU','Cor_INS_ND','Cor_INS_DU',  &
'Nuc_SOL_OC','Ait_SOL_SS','Nuc_SOL_SO','Ait_SOL_SO','Acc_SOL_SO',  & !130
'Cor_SOL_SO','Nuc_SOL_NH','Ait_SOL_NH','Acc_SOL_NH','Cor_SOL_NH',  &
'Nuc_SOL_NT','Ait_SOL_NT','Acc_SOL_NT','Cor_SOL_NT','XXX       ',  & !140
'Anth_Prec ','Bio_Prec  ','Anth_Cond ','Bio_Cond  ','XXX       ',  &
'XXX       ','XXX       ','XXX       ','PASSIVE O3','AGE OF AIR',  & !150
'RETIRED   ','RETIRED   ','RETIRED   ','RETIRED   ','RETIRED   ',  &
'RETIRED   ','RETIRED   ','RETIRED   ','RETIRED   ','RETIRED   ',  & !160
'RETIRED   ','RETIRED   ','RETIRED   ','RETIRED   ','RETIRED   ',  &
'RETIRED   ','RETIRED   ','RETIRED   ','RETIRED   ','RETIRED   ',  & !170
'RETIRED   ','RETIRED   ','XXX       ','XXX       ','XXX       ',  &
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  & !180
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  &
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  & !190
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  &
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  & !200
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  &
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  & !210
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  &
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  & !220
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  &
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  & !230
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  &
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  & !240
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  &
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  & !250
'XXX       ','XXX       ','XXX       ','XXX       ','XXX       ',  &
'XXX       '                                                        & !256
 /)
```

This array is **case sensitive** and each string is a **fixed length** of **10 characters**, with the whitespace being made up of spaces and not another tpye of white space (e.g. tabs).

When adding new tracers you should not overwrite tracers that are already in use by the chemistry scheme that you are using (e.g. StratTrop etc.), and it is best to also take care and avoid tracers in use by other schemes, if possible. You can see the tracers used by each scheme by looking in the `chch_defs` array at the top of the `ukca_chem_master.F90` module.

Here you should put a new species, `'ALICE     '`, into tracer slot 64 (current specified as `'CF2ClCFCl2'`, which is not in use by any scheme at the moment).

Remember: You need to add **2** tracers, ALICE in s34i064 and BOB in s34i065.

`ukca_chem_master.F90`

We will go through changes that need to be made for the StratTrop chemistry scheme (defined in the file **`ukca_chem_master.F90`**, using the code **`ST`**), but equivalent changes would need to be made for others, e.g. TropIsop or Strat. For this example we will only refer to this scheme.

In this file the list of species used in chemistry is defined by the array **`chch_defs_master`**. The list of species is held in a *derived type*, of the format:

```
chch_t1( N,'SPECIES   ',  X,'TYPE      ','FAMILY    ',  D,  W,  E, SCHEME, QUALIFIER, DISQUALIFIER, VN),  &
```

Where:

- **`N`** is an integer. The value should be incremented for each new **species** that is added.
- **`'SPECIES   '`** is the name of the species, e.g. `'O3        '`
- **`X`** number of odd atoms
- **`'TYPE      '`** describes whether the species is

  a tracer (`'TR        '`)
  a steady-state species (`'SS        '`)
  a constant (`'CT        '`)
  a constant field (`'CF        '`).

  For SS, CT, and CF, special code will need to be added.

- **`'FAMILY    '`** is the family that the species belongs to. This field is not currently used.
- **`D`** is **1** if the species is dry-deposited, and **0** otherwise.
- **`W`** is **1** if the species is wet-deposited, and **0** otherwise.
- **`E`** is **1** if the species is emitted, and **0** otherwise. This field is not currently used.
- **`SCHEME`** defines the chemistry scheme used, e.g.:

  ST = Stratosphere-Troposphere scheme
  T = Troposphere scheme
  S = Stratosphere scheme
  R = RAQ scheme
  OL = Offline-oxidants scheme
  TI = Troposphere-Isoprene scheme

  - If the same species is used in multiple schemes these are added, e.g. `ST+S+TI` etc.
- **`QUALIFIER`** (and **`DISQUALIFIER`**) defines lines to add (or remove) on top of the base chemistry scheme depending on if certain additional options have been chosen, e.g.

  0 = no qualifier/disqualifer used
  A = Aerosol chemistry additions required to run GLOMAP-mode
  TH = Tropospheric heterogeneous reactions
  HP = Heterogeneous PSC chemistry
  ES = Extended stratospheric reactions

  - As with the **`SCHEME`** codes, these are also additive if required, e.g. `A+TH` etc.
- **`VN`** gives the model version the addition is added at. This is future functionality that will allow for multiple rates to be included in the source-code. **This is currently not used, so please set to `107`.**

Therefore to add-in the ALICE species, we should insert a line similar to this one

```
chch_t1(116,'ALICE     ',1,'TR        ','          ',0,0,0,ST,0,0,107),&
```

into **`chch_defs_master`**, remembering to also **increment the size of this array, given by `n_chch_master`**

Remember: You need to add **2** tracers, ALICE in s34i064 and BOB in s34i065.

`ukca_setd1defs.F90`

In **`ukca_setd1defs.F90`** you will find code blocks which set values for various parameters depending on the scheme used. In this example you will need to scroll down to the section which defines the **StratTrop** (also known as *CheST*) chemistry, which is located in the *Stratospheric Chemistry* section and is referenced by using the **`L_ukca_strattrop`** logical.

The IF block tests against `L_ukca_strattrop` and `L_ukca_achem` (which determines whether or not you require the additional chemistry used to drive the GLOMAP-mode aerosol scheme). You should increment the value of **`n_chem_tracers`** by the number of chemical tracers that you are adding in **both** sections of the IF block, as this tracer is not defined to be for aerosol chemistry only. If your additional tracers are aerosol chemistry additions then you should increment the value of **`n_aero_tracers`**.

If you are adding to a different chemistry scheme then you will need to make these changes accordingly.

Remember: You need to add **2** tracers, ALICE in s34i064 and BOB in s34i065.

`ukca_constants.F90`

The unit of the tracers is kg(species)/kg(air) (i.e. mass mixing ratio, or **mmr**), but inside UKCA these species are converted to volume mixing ratio (or **vmr**). To enable UKCA to do this you will need to add the conversion factor(s) for your new tracer(s) into the code. This is done in the **`ukca_constants.F90`** module.

Open this file and add the required conversion factor(s). The naming convention for these is **M_species** is the molecular mass of the new species in g/mol, and **C_species** for the converson factor from vmr to mmr (calculated as M_species/M_air, where M_air=28.97). In actuality, only the C_species listing is always required, although you may need to add the M_species value later if you are emitting into the new species that you are adding.

For example:

```
REAL, PARAMETER :: C_ALICE      = 1.0000
```

Remember: You need to add **2** tracers, ALICE in s34i064 and BOB in s34i065.

`ukca_cspecies.F90`

After you have added in the **C_species** conversion factors, you will need to tell UKCA to use them for your species. To do this you need to edit the **ukca_cspecies.F90** module, which contains code which contructs the **c_species** array of conversion factors for the advected tracers. This contains a subroutine called `ukca_calc_cspecies` which has a long block of code that you need to edit to add an entry like this:

```
WHERE (advt == 'ALICE     ') c_species = C_ALICE
```

The **advt** array is automatically generated by UKCA at run-time from the `chch_defs_scheme` chemistry scheme definition you edited earlier, so your new tracer(s) will exist within it. You need to add in a new line for each of your tracers which sets the value of the **c_species** array to your individual C_species parameter. Add the line(s) in at the end of the block.

Remember: You need to add **2** tracers, ALICE in s34i064 and BOB in s34i065.

Run your suite

Now that you have made the required changes to Rose and in your branch, please run your suite. You may find that it fails on the, `fcm_make2`, `recon`, or `atmos` jobs. To find the errors and see output, you should either right-click the failed task in Gcylc and view the output, or go to the

```
/home/n02/n02/[your ARCHER userid]/cylc-run/[SUITE-ID]
```

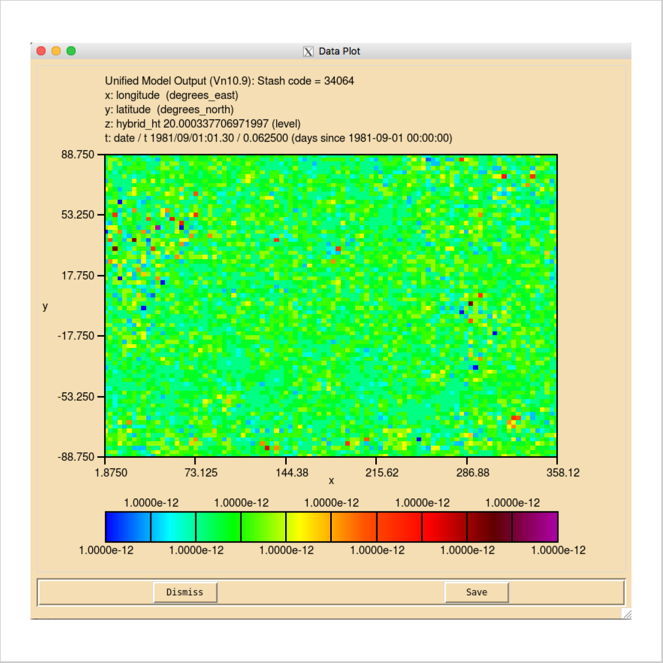
directory on ARCHER. Within this there are a number of directories. To see the log files, cd into

```
log/job/1/[JOB NAME]/NN
```

and view the job.err or job.out files. When the suite successfully runs, the fieldsfile output can usually be found in

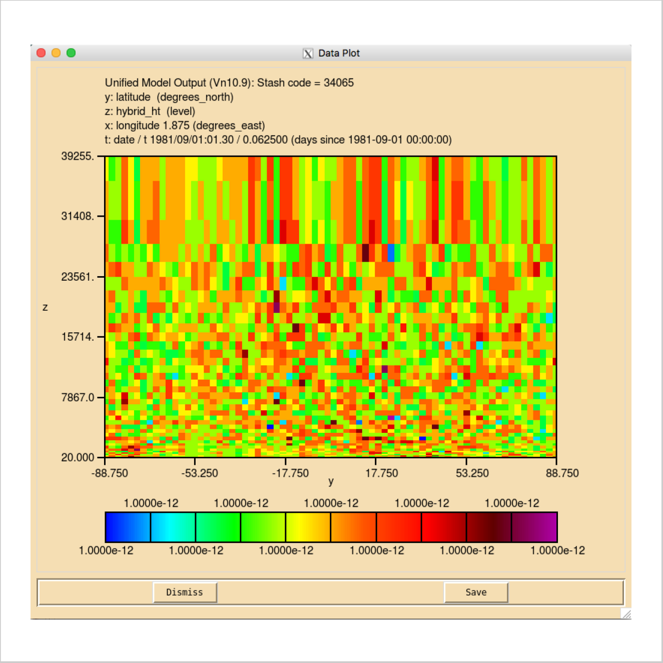
```
/work/n02/n02/[your ARCHER userid]/cylc-run/[SUITE-ID]/work/1/atmos/
```

Numerical Noise



Unified Model Output (Vn10.9): Stash code = 34064
x: longitude (degrees_east)
y: latitude (degrees_north)
z: hybrid_ht 20.000337706971997 (level)
t: date / t 1981/09/01:01.30 / 0.062500 (days since 1981-09-01 00:00:00)

Numerical noise in the ALICE tracer (s34i064)



Unified Model Output (Vn10.9): Stash code = 34065
y: latitude (degrees_north)
z: hybrid_ht (level)
x: longitude 1.875 (degrees_east)
t: date / t 1981/09/01:01.30 / 0.062500 (days since 1981-09-01 00:00:00)

Numerical noise in the BOB tracer (s34i065)

When your suite has run successfully, and you view the fields for your new tracer(s) in Xconv, you may find that they have a speckled appearance, rather than being constant. This is due to numerical noise that has been introduced as the tracer is run through the UKCA chemical solver. Once the field is converted to 32-bit this should disappear. It will also cease to become apparent when emissions and reactions are applied to these tracers.

Solution to Task 4.1

You were given the task

- *Add in two new tracers in to slots 64 and 65 for the StratTrop chemistry scheme. The tracer in slot 64 will be called **ALICE** and the tracer in slot 65 will be called **BOB** and initialise these tracers to `1.000000e-12`. You should also output these two tracers through the **UPA** stream in STASH as 3-hour means. In UKCA, you should set the conversion factor for each of these to 1.0.*

For a working Rose suite that has completed this task, please see

- **ARCHER:** `u-as292@59818`
- **vm:** `u-as297@59815`

The specific Rose changes made are:

- **ARCHER:** https://code.metoffice.gov.uk/trac/roses-u/changeset/59818/a/s/2/9/2/trunk
- **vm:** https://code.metoffice.gov.uk/trac/roses-u/changeset/59815/a/s/2/9/7/trunk

ARCHER:

```
Index: app/um/rose-app.conf
===================================================================
--- app/um/rose-app.conf        (revision 59777)
+++ app/um/rose-app.conf        (revision 59818)
@@ -1,4 +1,4 @@
-meta=um-atmos/vn10.9
+meta=um-atmos/HEAD

 [command]
 default=um-atmos
@@ -24,6 +24,7 @@
 RECON_KEEP_MPP_STDOUT=true
 RUNID=atmos
 SPECTRAL_FILE_DIR=$UM_INSTALL_DIR/vn$VN/ctldata/spectral/ga7
+STASHMASTER=STASHmaster

 [file:$DATAM]
 mode=mkdir
@@ -644,6 +645,18 @@
 !!user_prog_ancil_stash_req=0
 !!user_prog_rconst=0

+[namelist:items(3e9a6939)]
+ancilfilename=
+domain=1
+!!interval=0
+!!netcdf_varname=
+!!period=1
```

```
+source=6
+stash_req=34064
+update_anc=.false.
+!!user_prog_ancil_stash_req=0
+user_prog_rconst=1.000000e-12
+
 [namelist:items(4a4f86c3)]
 ancilfilename='$UM_INSTALL_DIR/ancil/atmos/n48e/orca1/seaice/reynolds/1981_2012_360/v1/qrclim.seaice'
 domain=1
@@ -728,6 +741,18 @@
 !!user_prog_ancil_stash_req=
 user_prog_rconst=1.000000e-12

+[namelist:items(c3e3b7ad)]
+ancilfilename=
+domain=1
+!!interval=0
+!!netcdf_varname=
+!!period=1
+source=6
+stash_req=34065
+update_anc=.false.
+!!user_prog_ancil_stash_req=0
+user_prog_rconst=1.000000e-12
+
 [namelist:items(c54d9b29)]
 ancilfilename='$UM_INSTALL_DIR/ancil/atmos/n48e/orca1/land_sea_mask/etop01/v1/qrparm.mask'
 domain=1
@@ -3295,6 +3320,22 @@
 tim_name='T3HMN'
 use_name='UPA'

+[namelist:umstash_streq(34064_487e535d)]
+dom_name='DALLTH'
+isec=34
+item=64
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
+[namelist:umstash_streq(34065_6dc07351)]
+dom_name='DALLTH'
+isec=34
+item=65
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
 [namelist:umstash_streq(34106_ebb7bb67)]
 dom_name='DALLTH'
 isec=34
Index: app/fcm_make/rose-app.conf
===================================================================
--- app/fcm_make/rose-app.conf  (revision 59777)
+++ app/fcm_make/rose-app.conf  (revision 59818)
@@ -42,4 +42,4 @@
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46535
Index: rose-suite.conf
===================================================================
--- rose-suite.conf     (revision 59777)
+++ rose-suite.conf     (revision 59818)
@@ -1,3 +1,6 @@
+[file:app/um/file/STASHmaster]
+source=fcm:um.xm_br/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns/rose-meta/um-atmos/HEAD/etc/stash/STASHmaster@HE
+
 [jinja2:suite.rc]
 ARCHER_GROUP='n02-chem'
 ARCHER_QUEUE='short'
```

vm:

```

Index: app/um/rose-app.conf
===================================================================
--- app/um/rose-app.conf        (revision 59775)
+++ app/um/rose-app.conf        (revision 59815)
```

```
@@ -1,4 +1,4 @@
-meta=um-atmos/vn10.9
+meta=um-atmos/HEAD

 [command]
 default=um-atmos
@@ -26,6 +26,7 @@
 RECON_STDOUT_FILE=pe_output/atmos.fort6.pe
 RUNID=atmos
 SPECTRAL_FILE_DIR=$UMDIR/vn$VN/ctldata/spectral/ga7
+STASHMASTER=STASHmaster
 UM_THREAD_LEVEL=MULTIPLE

 [file:$DATAM]
@@ -647,6 +648,18 @@
 !!user_prog_ancil_stash_req=0
 !!user_prog_rconst=0

+[namelist:items(3e9a6939)]
+ancilfilename=
+domain=1
+!!interval=0
+!!netcdf_varname=
+!!period=1
+source=6
+stash_req=34064
+update_anc=.false.
+!!user_prog_ancil_stash_req=0
+user_prog_rconst=1.000000e-12
+
 [namelist:items(4a4f86c3)]
 ancilfilename='$UMDIR/ancil/atmos/n48e/orca1/seaice/reynolds/1981_2012_360/v1/qrclim.seaice'
 domain=1
@@ -731,6 +744,18 @@
 !!user_prog_ancil_stash_req=
 user_prog_rconst=1.000000e-12

+[namelist:items(c3e3b7ad)]
+ancilfilename=
+domain=1
+!!interval=0
+!!netcdf_varname=
+!!period=1
+source=6
+stash_req=34065
+update_anc=.false.
+!!user_prog_ancil_stash_req=0
+user_prog_rconst=1.000000e-12
+
 [namelist:items(c54d9b29)]
 ancilfilename='$UMDIR/ancil/atmos/n48e/orca1/land_sea_mask/etop01/v1/qrparm.mask'
 domain=1
@@ -3887,6 +3912,22 @@
 tim_name='T3HMN'
 use_name='UPA'

+[namelist:umstash_streq(34064_487e535d)]
+dom_name='DALLTH'
+isec=34
+item=64
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
+[namelist:umstash_streq(34065_6dc07351)]
+dom_name='DALLTH'
+isec=34
+item=65
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
 [namelist:umstash_streq(34106_ebb7bb67)]
 dom_name='DALLTH'
 isec=34
Index: app/fcm_make/rose-app.conf
===================================================================
--- app/fcm_make/rose-app.conf  (revision 59775)
+++ app/fcm_make/rose-app.conf  (revision 59815)
@@ -42,4 +42,4 @@
```

```
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46535
Index: rose-suite.conf
===================================================================
--- rose-suite.conf     (revision 59775)
+++ rose-suite.conf     (revision 59815)
@@ -1,3 +1,6 @@
+[file:app/um/file/STASHmaster]
+source=fcm:um.xm_br/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns/rose-meta/um-atmos/HEAD/etc/stash/STASHmaster@HE
+
 [jinja2:suite.rc]
 BUILD=true
 OFFLINE=false
```

These ARCHER differences can be found in the file **/home/ukca/Tutorial/vn10.9/worked_solutions/Task4.1/Task4.1_rose.patch** on PUMA.

The specific UM changes made are:

```

Index: src/atmosphere/UKCA/ukca_setd1defs.F90
===================================================================
--- src/atmosphere/UKCA/ukca_setd1defs.F90      (revision 46527)
+++ src/atmosphere/UKCA/ukca_setd1defs.F90      (revision 46535)
@@ -321,7 +321,7 @@
         (/'NO        ','CH4       ','CO        ','HCHO      ', &
           'C2H6      ','C3H8      ','Me2CO     ','MeCHO     ', &
           'C5H8      ','NO_aircrft'/)
-    n_chem_tracers = 71         ! No chem tracers
+    n_chem_tracers = 73         ! No chem tracers
     nr_therm       = 220        ! thermal reactions
     nr_phot        = 55         ! photolytic (ATA)

@@ -337,7 +337,7 @@
           'SO2_high  ','NH3       ','DMS       ','SO2_nat   ', &
           'BC_biomass','OC_biomass','NO_aircrft'/)
     n_aero_tracers = 12
-    n_chem_tracers = 71         ! No chem tracers
+    n_chem_tracers = 73         ! No chem tracers
     IF (L_ukca_trophet) THEN
       nr_therm     = 241        ! thermal reactions
     ELSE
Index: src/atmosphere/UKCA/ukca_chem_master.F90
===================================================================
--- src/atmosphere/UKCA/ukca_chem_master.F90    (revision 46527)
+++ src/atmosphere/UKCA/ukca_chem_master.F90    (revision 46535)
@@ -79,7 +79,7 @@
 ! define below the chemistry schemes

 ! define size of master chemistry
-INTEGER, PARAMETER :: n_chch_master = 148 ! number of known species
+INTEGER, PARAMETER :: n_chch_master = 150 ! number of known species
 INTEGER, PARAMETER :: n_het_master  =  10 ! number of heterogeneous reactions
 INTEGER, PARAMETER :: n_dry_master  =  57 ! number of dry deposition reactions
 INTEGER, PARAMETER :: n_wet_master  =  49 ! number of wet deposition reactions
@@ -376,7 +376,9 @@
 chch_t1(112,'TOLP1     ',1,'SS        ','          ',0,0,0,R,0,0,107),&
 chch_t1(113,'MEMALD    ',1,'TR        ','          ',0,0,0,R,0,0,107),&
 chch_t1(114,'GLY       ',1,'TR        ','          ',0,1,0,R,0,0,107),&
-chch_t1(115,'oXYLENE   ',1,'TR        ','          ',0,0,1,R,0,0,107)/)
+chch_t1(115,'oXYLENE   ',1,'TR        ','          ',0,0,1,R,0,0,107),&
+chch_t1(116,'ALICE     ',1,'TR        ','          ',0,0,0,ST,0,0,107),&
+chch_t1(117,'BOB       ',1,'TR        ','          ',0,0,0,ST,0,0,107)/)

 ! Heterogeneous chemistry
 ! Columns take the following meanings:
Index: src/atmosphere/UKCA/ukca_constants.F90
===================================================================
--- src/atmosphere/UKCA/ukca_constants.F90      (revision 46527)
+++ src/atmosphere/UKCA/ukca_constants.F90      (revision 46535)
@@ -272,6 +272,10 @@
 REAL, PARAMETER :: c_isosvoc2   = 2.3473   ! as C5H8
 REAL, PARAMETER :: c_isosoa     = 4.4874   ! 130.0

+! UKCA Tutorial Tracers:
+REAL, PARAMETER :: c_ALICE      = 1.0000
+REAL, PARAMETER :: c_BOB        = 1.0000
+
 !     molecular masses in g/mol of emitted species,
 !     for budget calculations
 REAL, PARAMETER :: m_ho2     =  33.007
Index: src/atmosphere/UKCA/ukca_nmspec_mod.F90
===================================================================
--- src/atmosphere/UKCA/ukca_nmspec_mod.F90     (revision 46527)
+++ src/atmosphere/UKCA/ukca_nmspec_mod.F90     (revision 46535)
@@ -86,7 +86,7 @@
 'BrO       ','BrCl      ','BrONO2    ','N2O       ','HCl       ',  & !50
 'HOCl      ','HBr       ','HOBr      ','ClONO2    ','CFCl3     ',  &
 'CF2Cl2    ','MeBr      ','N         ','O(3P)     ','MACRO2    ',  & !60
-'MeCl      ','CF2ClBr   ','CCl4      ','CF2ClCFCl2','CHF2Cl    ',  &
+'MeCl      ','CF2ClBr   ','CCl4      ','ALICE     ','BOB       ',  &
 'MeCCl3    ','CF3Br     ','H2OS      ','CH2Br2    ','H2        ',  & !70
 'DMS       ','SO2       ','H2SO4     ','MSA       ','DMSO      ',  &
 'NH3       ','CS2       ','COS       ','H2S       ','H         ',  & !80
Index: src/atmosphere/UKCA/ukca_cspecies.F90
===================================================================
--- src/atmosphere/UKCA/ukca_cspecies.F90       (revision 46527)
+++ src/atmosphere/UKCA/ukca_cspecies.F90       (revision 46535)
@@ -296,6 +296,9 @@
 WHERE (advt == 'ORGNIT    ') c_species = c_orgnit
 WHERE (advt == 'PASSIVE O3') c_species = 1.0
 WHERE (advt == 'AGE OF AIR') c_species = 1.0
+! UKCA Tutorial Tracers:
+WHERE (advt == 'ALICE     ') c_species = c_ALICE
+WHERE (advt == 'BOB       ') c_species = c_BOB

 ! non-advected tracers
 c_na_species=0.0
Index: rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster-meta.conf
===================================================================
--- rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster-meta.conf (revision 46527)
+++ rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster-meta.conf (revision 46535)
@@ -16215,6 +16215,14 @@
 description=O3P MASS MIXING RATIO AFTER TIMESTEP
 help=Oxygen atom (ground state) Mass Mixing Ratio in kg/kg(Air)

+[stashmaster:code(34064)]
+description=ALICE MASS MIXING RATIO AFTER TIMESTEP
+help=UKCA Tutorial Tracer ALICE Mass Mixing Ratio in kg/kg(Air)
+
+[stashmaster:code(34065)]
+description=BOB MASS MIXING RATIO AFTER TIMESTEP
+help=UKCA Tutorial Tracer BOB Mass Mixing Ratio in kg/kg(Air)
+
 [stashmaster:code(34070)]
 description=H2 MASS MIXING RATIO AFTER TIMESTEP
 help=Hydrogen Mass Mixing Ratio in kg/kg(Air)
@@ -22480,6 +22488,16 @@
 description=O3P MASS MIXING RATIO ON PRESS LEVS
 help=Oxygen atom (ground state) Mass Mixing Ratio in kg/kg(Air)

+[stashmaster:code(51064)]
+description=ALICE MASS MIXING RATIO AFTER TIMESTEP
+help=UKCA Tutorial Tracer ALICE Mass Mixing Ratio in kg/kg(Air)
+    =on pressure levels
+
+[stashmaster:code(51065)]
+description=BOB MASS MIXING RATIO AFTER TIMESTEP
+help=UKCA Tutorial Tracer BOB Mass Mixing Ratio in kg/kg(Air)
+    =on pressure levels
+
 [stashmaster:code(51070)]
 description=H2 MASS MIXING RATIO ON PRESS LEVS
 help=Hydrogen Mass Mixing Ratio in kg/kg(Air)
Index: rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster_A
===================================================================
--- rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster_A (revision 46527)
+++ rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster_A (revision 46535)
@@ -15942,6 +15942,18 @@
 4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
 5|    0 | 2160 |    0 |   65 |    0 |    0 |    0 |    0 |    0 |
 #
+1|    1 |   34 |   64 |ALICE MASS MIXING RATIO AFTER TSTEP |
+2|    2 |    0 |    1 |    1 |    2 |   40 |   11 |    0 |    0 |    0 |    0 |
+3| 000000000000000000000000010000 | 00000000000000000001 |    1 |
+4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
+5|    0 | 2164 |    0 |   65 |    0 |    0 |    0 |    0 |    0 |
+#
```

```
+1|     1 |    34 |    65 |BOB MASS MIXING RATIO AFTER TSTEP    |
+2|     2 |     0 |     1 |     1 |     2 |    40 |    11 |     0 |     0 |     0 |     0 |
+3| 00000000000000000000000010000 | 00000000000000000001 |     1 |
+4|     1 |     0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
+5|     0 | 2165 |     0 |    65 |     0 |     0 |     0 |     0 |     0 |
+#
 1|     1 |    34 |    69 |CH3OH MASS MIXING RATIO AFTER TS     |
 2|     2 |     0 |     1 |     1 |     2 |    40 |    11 |     0 |     0 |     0 |     0 |
 3| 00000000000000000000000000100 | 00000000000000000001 |     1 |
@@ -24580,6 +24592,18 @@
 4|     1 |     0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
 5|     0 | 2160 |     0 |     8 |     0 |     0 |     0 |     0 |     0 |
 #
+1|     1 |    51 |    64 |ALICE MMR ON PRESSURE LEVELS         |
+2|     0 |     0 |     1 |     1 |     3 |     1 |     2 |     0 |     0 |     0 |     1 |
+3| 00000000000000000000000010000 | 00000000000000000001 |     3 |
+4|     1 |     0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
+5|     0 | 2164 |     0 |     8 |     0 |     0 |     0 |     0 |     0 |
+#
+1|     1 |    51 |    65 |BOB MMR ON PRESSURE LEVELS           |
+2|     0 |     0 |     1 |     1 |     3 |     1 |     2 |     0 |     0 |     0 |     1 |
+3| 00000000000000000000000010000 | 00000000000000000001 |     3 |
+4|     1 |     0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
+5|     0 | 2165 |     0 |     8 |     0 |     0 |     0 |     0 |     0 |
+#
 1|     1 |    51 |    69 |CH3OH MASS MIX RATIO ON PRESS LEVS   |
 2|     0 |     0 |     1 |     1 |     3 |     1 |     2 |     0 |     0 |     0 |     1 |
 3| 00000000000000000000000000100 | 00000000000000000001 |     3 |
```

These differences can be found in the file **/home/ukca/Tutorial/vn10.9/worked_solutions/Task4.1/Task4.1_code.patch** on PUMA.

If you open the **.pa** file in Xconv, you should see the following fields:

```
7    : 96   72   38    1     field2164: Stash code = 34064
8    : 96   72   38    1     field2165: Stash code = 34065
```

Sample output from this task can be found at **/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task4.1/atmosa.pa19810901_00** on ARCHER.

Checklist

- □ If required, copy an appropriate Rose suite to use to test your changes.
- □ Make a new ticket on the Met Office SRS Trac pages.
- □ Make a branch at the required MetUM version using `fcm branch-create`.
- □ Checkout your branch using `fcm checkout`.
- □ Choose slots for your new tracer(s) from the list in `ukca_set_nmspec.F90`, and add the correct names for them in these slots.
- □ Add the tracer(s) to the `chch_defs_strattrop` array in `ukca_chem_master.F90`.
- □ Increase the value of `n_chem_tracers` in `ukca_setd1defs.F90`.
- □ Add a C_species value for each tracer in `ukca_constants.F90`.
- □ Append the CASE statement in `ukca_cspecies.F90` to add this value to the `c_species` array.
- □ `fcm commit` the changes to your branch.
- □ Make the required changes to branch's STASHmaster_A file, to add the new tracers.
- □ Add help text for your tracers in your branch's STASHmaster-meta.conf file.
- □ Using a text editor, open the `app/um/rose-app.conf` file from your `roses/[SUITE-ID]` directory, and add the line `STASHMASTER=STASHmaster` in the `[env]` block, then save and close the file.
- □ Using a text editor, open the `rose-suite.conf` file from your `roses/[SUITE-ID]` directory, and add the following lines to the top of the file, before saving and closing it:

```
[file:app/um/file/STASHmaster]
source=fcm:um.xm_br/dev/[your MOSRS userid]/vnX.Y_your_branch_name/rose-meta/um-atmos/HEAD/etc/stash/STASHmaster
```

- □ Edit your suite using `rose edit -M /path/to/your/branch/working/copy`.

□ Point the metadata in your suite to um-atmos/HEAD.

□ Include your branch in your suite at: fcm_make ⟶ env ⟶ Sources.

□ Initialise your tracers in your suite at: um ⟶ namelist ⟶ Reconfiguration and Ancillary Control ⟶ Configure ancils and initialise dump fields.

□ Output your tracers in STASH at: um ⟶ namelist ⟶ Model Input and Output ⟶ STASH Requests and Profiles ⟶ STASH Requests.

□ Run the TidyStashTransform transform macro.

□ Save your suite.

□ In the `roses/[SUITE-ID]` directory, run `fcm commit` to commit your changes to the repository.

□ Run your suite.

Tutorial 5

Written by Luke Abraham 2017

Retrieved from "http://www.ukca.ac.uk/wiki/index.php?title=UKCA_Chemistry_and_Aerosol_vn10.9_Tutorial_4&oldid=6940"

- This page was last modified on 26 April 2018, at 10:18.

UKCA Chemistry and Aerosol vn10.9 Tutorial 5

From UKCA

UKCA Chemistry and Aerosol Tutorials at vn10.9

Contents

What you will learn in this Tutorial

During this tutorial you will learn how to make new UKCA netCDF emissions files. Then you will learn how to add new emissions into UKCA so that they emit into one of your new tracers.

At the end of the previous tutorial you will now know how to create new tracers for use by UKCA. However, after completing the tasks, your tracers will still be empty, as nothing has been put into them. This tutorial will teach you how to create an emissions file that the UKCA will read, and that you can then tell UKCA to use and emit into your tracer(s).

This tutorial will go through the steps needed to make an emission into a tracer which UKCA does not currently emit into. The steps in making the netCDF file will be the same for a species which is currently emitted into, although in this simpler case you would not need to make any code changes.

This example only deals with a single field. For a more detailed examples on changing emissions for all species, please see the Emissions for ACSIS webpage.

Task 5.1: Create a new emissions file and use it in your job

Task 5.1: In the

```
/work/n02/n02/ukca/Tutorial/vn10.9/Task5.1
```

directory on ARCHER there is the file **Emissions_of_ALICE.nc** which is a 0.5x0.5 degree resolution surface emission field. You should regrid this file to the **N48 ENDGame grid**, and output it as a netCDF file that has the required netCDF metadata for a climatological surface emission without any diurnal cycle.

The netCDF emissions system

UKCA uses netCDF files to specify the emissions into UKCA species. Using netCDF files means that the **metadata** in the file can be used to specify various options, such as which diurnal cycle to use.

Metadata

In Global Atmosphere 7.0, UKCA emissions are prescribed using netCDF files. The reason for this move away from the UM ancillary file format

Each emission field in the NetCDF files needs to include the following variable metadata attributes (there is no requirement for the netCDF variable names themselves except that they be no longer than 80 characters):

- **`standard_name:`** It should be included if an appropriate name is present in the CF Standard Name Table (see http://cfconventions.org/standard-names.html). An example of a valid ‘standard name’ is:

  `“tendency_of_atmosphere_mass_content_of_nitrogen_monoxide_due_to_emission”` (note that all substrings have to be separated by underscores).

- **`long_name`:** It is recommendable but only compulsory if there is no ‘standard_name’ available for that emission field. There are no specific CF conventions for standard names, but within the UKCA code all substrings should be separated by white space as shown in this example:

  ```
  “tendency of atmosphere mass content of nitrogen monoxide due to emission”.
  ```

  The reason for this is given in the description of the attribute units.

- **`tracer_name`:** This attribute has to be equal to the name of a tracer (or of an emission field such as ‘NO - aircrft’) present in the list of emissions for the given chemical scheme, i.e. `em_chem_spec`. This name is used by the subroutine UKCA_ADD_EMISS to add the emission field to the corresponding tracer. There should be at least one emission field in the NetCDF files (or alternatively an online emission field) with the value of tracer name equal to one of the items in em_chem_spec, otherwise the subroutine UKCA_EMISS_INIT (in module UKCA_EMISS_MOD) will report a missing tracer in the emission files and the model will stop with error.
- **`units`:** According to CF conventions, all emissions should be expressed in $kg\,m^{-2}\,s^{-1}$. As a consequence, all files will contain the attribute

  ```
  units = "kg m-2 s-1"
  ```

  However, under some circumstances emission fields need to be reported as kg of a given species (e.g. nitrogen, carbon, sulfur). When that is the case this needs to be indicated in the standard name attribute if possible (see example below), otherwise in the long name attribute. As an example, in the case of an emission field reported as kg of carbon, the attribute units will be “kg m-2 s-1” while the attribute standard name should contain the substring `“expressed_as_carbon”` if that is accepted by CF conventions for standard names (see http://cf-pcmdi.llnl.gov/documents/cf-standard-names); otherwise the attribute long name should contain the substring `“expressed as carbon”`. The UM subroutines BASE_EMISS_FACTORS and GET_BASE_SCALING will look for such substrings and apply some conversions if needed. It is therefore essential that the substrings in standard_name and long_name are separated by underscores and white spaces, respectively, as indicated above.

- **`hourly_scaling`: (optional)** A character attribute used to apply a diurnal cycle to emissions data with daily or lower frequency. Allowed values:
  - `none` (or attribute not present): no scaling
  - `traffic_uk`: used for UK air quality studies
  - `TNO_MACC_EU_SNAPnn` (where nn=01 to 11): Hourly factors of emissions for Europe. Calculated by TNO for the MACC project.
  - `diurnal_isopems` (for isoprene emissions): use the routine UKCA DIURNAL ISOP EMS to calculate a diurnal cycle using solar zenith angle and latitude to compute the expected number of sunshine hours.
- **`daily_scaling` (optional):** A character attribute used to apply a weekly cycle to emissions weekly or lower frequency. Only allowed when the model is using the Gregorian calendar. Allowed values:
  - `none` (or attribute not present): no scaling
  - `traffic_uk`: used for UK air quality studies
  - `TNO_MACC_GB_SNAPnn` (where nn=01 to 11): Hourly factors of emissions for Great Britain. Calculated by TNO for the MACC project.
- **`vertical_scaling` (optional):** A character attribute defining the vertical distribution of the emission. This allows the user to supply a 2D field to be applied over multiple levels. Allowed values:
  - `surface` (or attribute not present): treat as surface emission
  - `all_levels`, `3D`: field is three-dimensional and is provided on model levels (must have the same number of levels as the model)
  - `high_level`: spread a 2D field over multiple model levels, weighting by model layer thickness to achieve a uniform distribution in height. Must be accompanied by variable attributes lowest level and highest level to indicate the model levels over which to distribute the emission.
  - `step1`: used for air quality simulations, spreads emission over lowest 3 layers of the model; only allowed with 38 model levels.
  - `EMEP_modified_SNAPnn` (where nn=01 to 11): Average vertical profiles for SNAP sectors, similar to implementation for EMEP model. See routine vertical emiss factors for details.
  - `Bieser_modified_SNAPnn` (where nn=01 to 11): Average vertical profiles for SNAP sectors, similar to implementation of Bieser et al. (2011) with the SMOKE model and including fugitive emissions. See routine vertical_emiss_factors for details. In addition, each NetCDF file has to include two global attributes with information that is valid for all emission fields present in the file:
- **`update_freq_in_hours`:** Integer indicating the frequency (in hours) at which all emission fields present in that file should be read to update emissions(:)%values (:,:,:) in the UKCA code. Update points are calculated relative the model’s ancil reftime.
- **`update_type`**: Integer number indicating the times at which the data is provided. The same conventions as for ancillary files have been adopted:
  - 0: Single time
  - 1: Time series
  - 2: Periodic time series

  When emissions are treated as time series (update_type=1) the user is responsible for creating emission fields which cover the whole period for which the model is run; otherwise the model will not find the time registers to do interpolations and will stop with error. When emissions are periodic (update_type=2) each emission field has to include exactly 12 monthly average emissions (Jan, Feb, ..., Dec). Other attributes, in particular some global attributes, as well as some additional fields (e.g. a variable indicating the type of grid mapping) should be present in the files to comply with CF conventions for NetCDF data, but they are not used by the UKCA code.

Using python to regrid your emissions

For this task we will be using the Iris (http://scitools.org.uk/iris/) python library, which is developed by the Met Office and can read Met Office formatted files. The cf-python (http://cfpython.bitbucket.org/) library also does this, but we will not be using this here.

ARCHER

You can use Iris on ARCHER by loading the anaconda (https://www.continuum.io/anaconda-overview) module:

```
module load anaconda/2.2.0-python2
```

Note that this is not the default anaconda module, which does not contain Iris.

If you then

```
conda list
```

you will see that Iris is available:

```
iris                      1.10.0               np19py27_2     https://conda.binstar.org/scitools
```

Virtual Machine

To be able to use Iris on the VM, you should first run the command

```
install-iris
```

which installs the required python packages from anaconda (https://www.continuum.io/anaconda-overview) , and also creates an alias in your `~/.bashrc` file, called **conda**, which opens a terminal with additional paths set. You may need to re-source your `~/.bashrc` by

```
. ~/.bashrc
```

before running the

```
conda
```

command.

Using python

when developing python scripts, a handy command to use can be set by the following alias:

```
alias pylab='ipython --pylab --logfile=ipython-`date +"%Y%m%d-%H%M%S"`.py'
```

This uses ipython (http://ipython.readthedocs.io/en/stable/) , which is an interactive shell for python. The `--pylab` automatically loads some standard python libraries such as scipy (https://www.scipy.org/) , numpy (http://www.numpy.org/) , and matplotlib (http://matplotlib.org/) . The `--logfile` means that all commands will be recorded in a file with the format `ipython-YYYYMMDD-HHMMSS.py`.

You can also execute pre-written python scripts by

- **ARCHER:** `python2.7 -c "execfile('name_of_script.py')"`
- **vm:** `/home/vagrant/miniconda2/bin/python2.7 -c "execfile('name_of_script.py')"`

from the command-line, or running the `execfile` command within ipython.

However, **don't panic**. **You are not expected to know python to complete this tutorial.** You will be provided with an example script to use that you only need to edit a few lines to get to work. You do not need to write a script from scratch, just read-through the provided script and try to understand what it does, and why.

Example python script

In the directory:

```
/work/n02/n02/ukca/Tutorial/vn10.9/Task5.1
```

you will find the file **`regrid_ALICE_N48e.py`**. You should take a copy of this file, and using your output **.pa** file from your suite, you should regrid the emissions into the N48 ENDGame grid. You will later use the resulting netCDF file in your suite.

This file looks like:

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

# This file is part of the UKCA Tutorials:
#  http://www.ukca.ac.uk/wiki/index.php/UKCA_Chemistry_and_Aerosol_Tutorials_at_vn10.9

# Copyright (C) 2017  University of Cambridge

# This is free software: you can redistribute it and/or modify it under the
# terms of the GNU Lesser General Public License as published by the Free Software
# Foundation, either version 3 of the License, or (at your option) any later
# version.

# It is distributed in the hope that it will be useful, but WITHOUT ANY
# WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A
# PARTICULAR PURPOSE.  See the GNU Lesser General Public License for more details.

# You find a copy of the GNU Lesser General Public License at <http://www.gnu.org/licenses/>.

# Written by N. Luke Abraham 2017-11-15 <nla27@cam.ac.uk>

# To use this script on ARCHER you should first
#    module load anaconda/2.2.0-python2
# To be able to use ncdump you should fist
#    module load nco
# preamble
import iris
import numpy
# pre-Iris v1.10, use iris.unit instead of cf_units
import cf_units

# --- CHANGE THINGS BELOW THIS LINE TO WORK WITH YOUR FILES ETC. ---

# name of file containing an ENDGame grid, e.g. your model output
# NOTE: all the fields in the file should be on the same horizontal
#       grid, as the field used MAY NOT be the first in order of STASH
grid_file='/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task4.1/atmosa.pa19810901_00'
# name of emissions file
emissions_file='/work/n02/n02/ukca/Tutorial/vn10.9/Task5.1/Emissions_of_ALICE.nc'

# --- BELOW THIS LINE, NOTHING SHOULD NEED TO BE CHANGED ---

species_name='ALICE'

# this is the grid we want to regrid to, e.g. N48 ENDGame
grd=iris.load_cube(grid_file,iris.AttributeConstraint(STASH='m01s34i010'))
grd.coord(axis='x').guess_bounds()
grd.coord(axis='y').guess_bounds()

# This is the original data
ems=iris.load_cube(emissions_file)
# make intersection between 0 and 360 longitude to ensure that
# the data is regridded correctly
nems = ems.intersection(longitude=(0, 360))

# make sure that we use the same coordinate system, otherwise regrid won't work
nems.coord(axis='x').coord_system=grd.coord_system()
nems.coord(axis='y').coord_system=grd.coord_system()

# now guess the bounds of the new grid prior to regridding
nems.coord(axis='x').guess_bounds()
nems.coord(axis='y').guess_bounds()

# now regrid
ocube=nems.regrid(grd,iris.analysis.AreaWeighted())

# now add correct attributes and names to netCDF file
ocube.var_name='emissions_'+str.strip(species_name)
ocube.long_name=str.strip(species_name)+' surf emissions'
ocube.units=cf_units.Unit('kg m-2 s-1')
ocube.attributes['vertical_scaling']='surface'
ocube.attributes['tracer_name']=str.strip(species_name)
# global attributes, so don't set in local_keys
# NOTE: all these should be strings, including the numbers! This will change at a later UM version.
# basic emissions type
ocube.attributes['emission_type']='2' # periodic
ocube.attributes['update_type']='2'   # same as above
ocube.attributes['update_freq_in_hours']='120' # i.e. 5 days
ocube.attributes['source']='UKCA Tutorial Task 5.1 - creating netCDF emissions'

# rename and set time coord - set to be 0000/01/16:00:00-0000/12/16:00:00
# this bit is annoyingly fiddly
ocube.coord(axis='t').var_name='time'
ocube.coord(axis='t').standard_name='time'
ocube.coords(axis='t')[0].units=cf_units.Unit('hours since 1970-01-01', calendar='360_day')
ocube.coord(axis='t').points=numpy.array([-17020440, -17019720, -17019000, -17018280,
                                          -17017560, -17016840, -17016120, -17015400,
                                          -17014680, -17013960, -17013240, -17012520])
# make z-direction.
zdims=iris.coords.DimCoord(numpy.array([0]),standard_name = 'model_level_number',
                           units='1',attributes={'positive':'up'})
ocube.add_aux_coord(zdims)
ocube=iris.util.new_axis(ocube, zdims)
# now transpose cube to put Z 2nd
ocube.transpose([1,0,2,3])

# make coordinates 64-bit
ocube.coord(axis='x').points=ocube.coord(axis='x').points.astype(dtype='float64')
ocube.coord(axis='y').points=ocube.coord(axis='y').points.astype(dtype='float64')
#ocube.coord(axis='z').points=ocube.coord(axis='z').points.astype(dtype='float64') # integer
ocube.coord(axis='t').points=ocube.coord(axis='t').points.astype(dtype='float64')
# for some reason, longitude_bounds are double, but latitude_bounds are float
ocube.coord('latitude').bounds=ocube.coord('latitude').bounds.astype(dtype='float64')
```

```
# add forecast_period & forecast_reference_time
# forecast_reference_time
frt=numpy.array([-17020080, -17019360, -17018640, -17017920,
                 -17017200, -17016480, -17015760, -17015040,
                 -17014320, -17013600, -17012880, -17012160],dtype='float64')
frt_dims=iris.coords.AuxCoord(frt,standard_name = 'forecast_reference_time',
                           units=cf_units.Unit('hours since 1970-01-01', calendar='360_day'))
ocube.add_aux_coord(frt_dims,data_dims=0)
ocube.coord('forecast_reference_time').guess_bounds()
# forecast_period
fp=numpy.array([-360],dtype='float64')
fp_dims=iris.coords.AuxCoord(fp,standard_name = 'forecast_period',
                           units=cf_units.Unit('hours'),bounds=numpy.array([-720,0],dtype='float64'))
ocube.add_aux_coord(fp_dims,data_dims=None)

# add-in cell_methods
ocube.cell_methods = [iris.coords.CellMethod('mean', 'time')]
# set _FillValue
fillval=1e+20
ocube.data = numpy.ma.array(data=ocube.data, fill_value=fillval, dtype='float32')

# output file name, based on species
outpath='ukca_emiss_'+species_name+'.nc'
# don't want time to be cattable, as is a periodic emissions file
iris.FUTURE.netcdf_no_unlimited=True
# annoying hack to set a missing_value attribute as well as a _FillValue attribute
dict.__setitem__(ocube.attributes, 'missing_value', fillval)
# now write-out to netCDF
saver = iris.fileformats.netcdf.Saver(filename=outpath, netcdf_format='NETCDF3_CLASSIC')
saver.update_global_attributes(Conventions=iris.fileformats.netcdf.CF_CONVENTIONS_VERSION)
saver.write(ocube, local_keys=['vertical_scaling', 'missing_value','um_stash_source','tracer_name'])

# end of script

# Why we are messing around with metadata?
#------------------------------------------
#
# We need to adapt the metadata of the emissions data to
# match what UKCA is expecting.
#  e.g. the metadata of the 'Emissions_of_ALICE.nc file is:
#
#    netcdf Emissions_of_ALICE {
#    dimensions:
#            lon = 720 ;
#            lat = 360 ;
#            date = UNLIMITED ; // (12 currently)
#    variables:
#            float lon(lon) ;
#                    lon:long_name = "Longitude" ;
#                    lon:standard_name = "longitude" ;
#                    lon:units = "degrees_east" ;
#                    lon:point_spacing = "even" ;
#                    lon:modulo = " " ;
#            float lat(lat) ;
#                    lat:long_name = "Latitude" ;
#                    lat:standard_name = "latitude" ;
#                    lat:units = "degrees_north" ;
#                    lat:point_spacing = "even" ;
#            float date(date) ;
#                    date:long_name = "Time" ;
#                    date:units = "days since 1960-01-01" ;
#                    date:time_origin = "01-JAN-1960:00:00:00" ;
#            float ALICE(date, lat, lon) ;
#                    ALICE:source = " " ;
#                    ALICE:name = "ALICE" ;
#                    ALICE:title = "Emissions of ALICE in kg/m^2/s" ;
#                    ALICE:date = "01/01/60" ;
#                    ALICE:time = "00:00" ;
#                    ALICE:long_name = "Emissions of ALICE in kg/m^2/s" ;
#                    ALICE:standard_name = "tendency_of_atmosphere_mass_content_of_ALICE_due_to_emission" ;
#                    ALICE:units = "kg/m2/s" ;
#                    ALICE:missing_value = 2.e+20f ;
#                    ALICE:_FillValue = 2.e+20f ;
#                    ALICE:valid_min = 0.f ;
#                    ALICE:valid_max = 2.60646e-08f ;
#
#    // global attributes:
#                    :history = "Tue Jun 18 14:32:42 BST 2013 - XCONV V1.92 16-February-2006" ;
#    }
#
```

```
#  whereas, the metadata of the
#  /work/n02/n02/hum/ancil/atmos/n48e/ukca_emiss/cmip5/2000/v1/ukca_emiss_CO.nc
#  file is, e.g.:
#
#    netcdf ukca_emiss_CO {
#    dimensions:
#            time = UNLIMITED ; // (12 currently)
#            model_level_number = 1 ;
#            latitude = 72 ;
#            longitude = 96 ;
#            bnds = 2 ;
#    variables:
#            double emissions_CO(time, model_level_number, latitude, longitude) ;
#                    emissions_CO:long_name = "CO surf emissions" ;
#                    emissions_CO:units = "kg m-2 s-1" ;
#                    emissions_CO:um_stash_source = "m01s00i303" ;
#                    emissions_CO:tracer_name = "CO" ;
#                    emissions_CO:vertical_scaling = "surface" ;
#                    emissions_CO:cell_methods = "time: mean" ;
#                    emissions_CO:grid_mapping = "latitude_longitude" ;
#                    emissions_CO:coordinates = "forecast_period forecast_reference_time" ;
#            int latitude_longitude ;
#                    latitude_longitude:grid_mapping_name = "latitude_longitude" ;
#                    latitude_longitude:longitude_of_prime_meridian = 0. ;
#                    latitude_longitude:earth_radius = 6371229. ;
#            double time(time) ;
#                    time:axis = "T" ;
#                    time:bounds = "time_bnds" ;
#                    time:units = "hours since 1970-01-01 00:00:00" ;
#                    time:standard_name = "time" ;
#                    time:calendar = "360_day" ;
#            double time_bnds(time, bnds) ;
#            int model_level_number(model_level_number) ;
#                    model_level_number:axis = "Z" ;
#                    model_level_number:units = "metre" ;
#                    model_level_number:standard_name = "model_level_number" ;
#                    model_level_number:long_name = "height at theta layer midpoint" ;
#                    model_level_number:positive = "up" ;
#            float latitude(latitude) ;
#                    latitude:axis = "Y" ;
#                    latitude:bounds = "latitude_bnds" ;
#                    latitude:units = "degrees_north" ;
#                    latitude:standard_name = "latitude" ;
#            float latitude_bnds(latitude, bnds) ;
#            float longitude(longitude) ;
#                    longitude:axis = "X" ;
#                    longitude:bounds = "longitude_bnds" ;
#                    longitude:units = "degrees_east" ;
#                    longitude:standard_name = "longitude" ;
#            double longitude_bnds(longitude, bnds) ;
#            double forecast_period ;
#                    forecast_period:bounds = "forecast_period_bnds" ;
#                    forecast_period:units = "hours" ;
#                    forecast_period:standard_name = "forecast_period" ;
#            double forecast_period_bnds(bnds) ;
#            double forecast_reference_time(time) ;
#                    forecast_reference_time:units = "hours since 1970-01-01 00:00:00" ;
#                    forecast_reference_time:standard_name = "forecast_reference_time" ;
#                    forecast_reference_time:calendar = "360_day" ;
#
#    // global attributes:
#                    :emission_type = "2" ;
#                    :source = "Data from Met Office Unified Model" ;
#                    :um_version = "7.3" ;
#                    :update_freq_in_hours = "120" ;
#                    :update_type = "2" ;
#                    :Conventions = "CF-1.5" ;
#    }
#
#  so the metadata of our new emissions file needs to be edited to be what UKCA
#  expects.
#
#  After using this script, the resultant netCDF file should look like:
#
#    netcdf ukca_emiss_ALICE {
#    dimensions:
#            time = 12 ;
#            model_level_number = 1 ;
#            latitude = 72 ;
#            longitude = 96 ;
#            bnds = 2 ;
#    variables:
#            float emissions_ALICE(time, model_level_number, latitude, longitude) ;
#                    emissions_ALICE:_FillValue = 1.e+20f ;
#                    emissions_ALICE:long_name = "ALICE surf emissions" ;
#                    emissions_ALICE:units = "kg m-2 s-1" ;
#                    emissions_ALICE:missing_value = 1.e+20 ;
#                    emissions_ALICE:tracer_name = "ALICE" ;
#                    emissions_ALICE:vertical_scaling = "surface" ;
#                    emissions_ALICE:cell_methods = "time: mean" ;
#                    emissions_ALICE:grid_mapping = "latitude_longitude" ;
#                    emissions_ALICE:coordinates = "forecast_period forecast_reference_time" ;
#            int latitude_longitude ;
#                    latitude_longitude:grid_mapping_name = "latitude_longitude" ;
#                    latitude_longitude:longitude_of_prime_meridian = 0. ;
#                    latitude_longitude:earth_radius = 6371229. ;
#            double time(time) ;
#                    time:axis = "T" ;
#                    time:units = "hours since 1970-01-01" ;
#                    time:standard_name = "time" ;
#                    time:long_name = "Time" ;
#                    time:calendar = "360_day" ;
#                    time:time_origin = "01-JAN-1960:00:00:00" ;
#            int model_level_number(model_level_number) ;
#                    model_level_number:axis = "Z" ;
#                    model_level_number:units = "1" ;
#                    model_level_number:standard_name = "model_level_number" ;
#                    model_level_number:positive = "up" ;
#            double latitude(latitude) ;
#                    latitude:axis = "Y" ;
#                    latitude:bounds = "latitude_bnds" ;
#                    latitude:units = "degrees_north" ;
#                    latitude:standard_name = "latitude" ;
#            double latitude_bnds(latitude, bnds) ;
#            double longitude(longitude) ;
#                    longitude:axis = "X" ;
#                    longitude:bounds = "longitude_bnds" ;
#                    longitude:units = "degrees_east" ;
#                    longitude:standard_name = "longitude" ;
#            double longitude_bnds(longitude, bnds) ;
#            double forecast_period ;
#                    forecast_period:bounds = "forecast_period_bnds" ;
#                    forecast_period:units = "hours" ;
#                    forecast_period:standard_name = "forecast_period" ;
#            double forecast_period_bnds(bnds) ;
#            double forecast_reference_time(time) ;
#                    forecast_reference_time:bounds = "forecast_reference_time_bnds" ;
#                    forecast_reference_time:units = "hours since 1970-01-01" ;
#                    forecast_reference_time:standard_name = "forecast_reference_time" ;
#                    forecast_reference_time:calendar = "360_day" ;
#            double forecast_reference_time_bnds(time, bnds) ;
#
#    // global attributes:
#                    :Conventions = "CF-1.5" ;
#                    :date = "01/01/60" ;
#                    :emission_type = "2" ;
#                    :history = "Tue Jun 18 14:32:42 BST 2013 - XCONV V1.92 16-February-2006" ;
#                    :invalid_standard_name = "tendency_of_atmosphere_mass_content_of_ALICE_due_to_emission" ;
#                    :name = "ALICE" ;
#                    :source = "UKCA Tutorial Task 5.1 - creating netCDF emissions" ;
#                    :time = "00:00" ;
#                    :title = "Emissions of ALICE in kg/m^2/s" ;
#                    :update_freq_in_hours = "120" ;
#                    :update_type = "2" ;
#                    :valid_max = 2.60646e-08f ;
#                    :valid_min = 0.f ;
#    }
```

All scripts provided for these tutorials can also be found on GitHub (https://github.com/theabro/ukca) .

Solution to Task5.1

You were given the task:

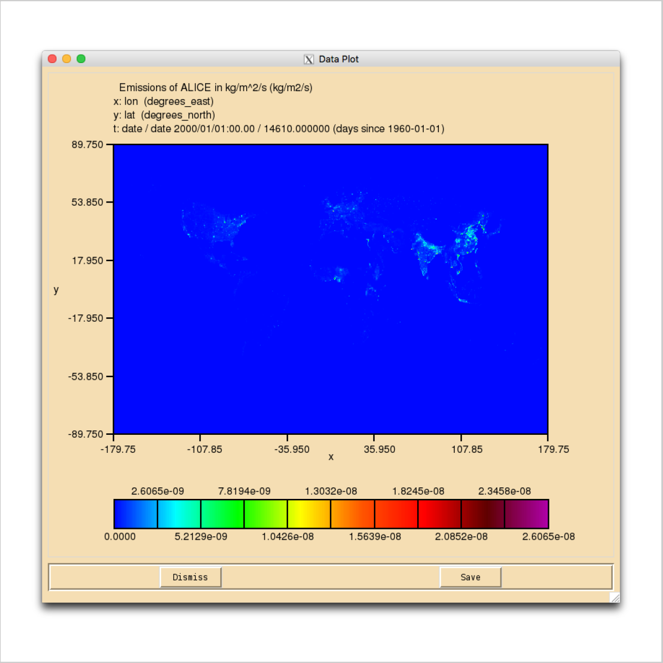
- *In the*

  ```
  /work/n02/n02/ukca/Tutorial/vn10.9/Task5.1
  ```

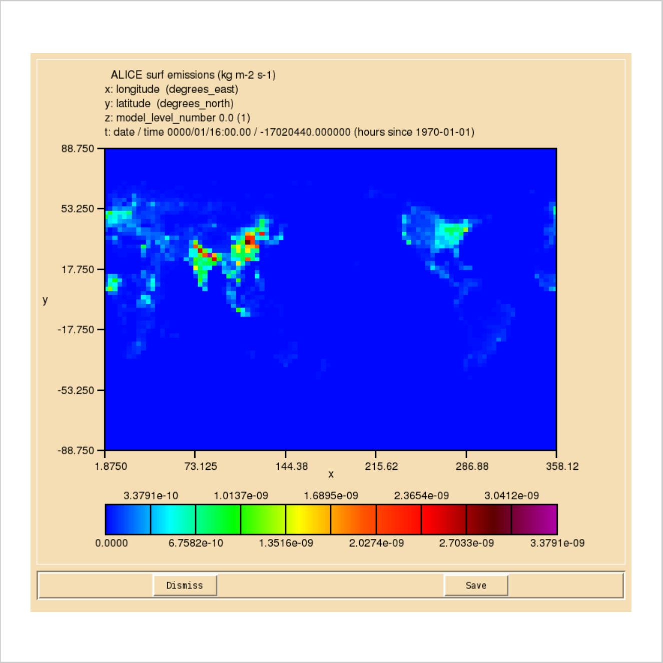
> *directory on ARCHER there is the file **Emissions_of_ALICE.nc** which is a 0.5x0.5 degree resolution surface emission field. You should regrid this file to the **N48 ENDGame grid**, and output it as a netCDF file that has the required netCDF metadata for a climatological surface emission without any diurnal cycle.*

It should be straight-forward to point to your **.pa** file to use in the script. You should read through it and have an understanding of how it works.

Once the file has been regridded, you should see the output looking like this:



ALICE emissions prior to regridding.



ALICE emissions after regridding. Note the shift of the grid.

A file that has been produced by the above script can be found at

```
/work/n02/n02/ukca/Tutorial/vn10.9/Task5.1/solution/ukca_emiss_ALICE.nc
```

Task 5.2: make the required code changes to add your emission into UKCA

TASK 5.2: You should now make the UKCA code changes to add your emission into the **ALICE** tracer. Use STASH code `s50i304` for the diagnosic.

Hint [hide]
You will need to add-in the molar mass of ALICE. You can calculate this from the mass of air and the conversion factor defined in Task 4.1. You will need to add code to allow a diagnostic of the emission to be output.

Rose changes

There are several ways that the location of emissions files can be defined within the UKCA panel. These are:

1. With a full path to the file
2. Using environment variables set using an **ancil_versions** file
3. A mixture of (1) and (2)
4. By specifying a top-level directory in the **`ukca_em_dir`** Rose variable, and then paths to the required files within **`ukca_em_files`**

Within Rose you may also find that the files are in separate boxes (one for each file) or in a single long list.

The UKCA training suite uses option **4** above, specifying both **`ukca_em_dir`** and **`ukca_em_files`**.

ARCHER

As **`ukca_em_dir`** is within `UM_INSTALL_DIR`, you will need to either copy or symbolically-link the directories from here to your own space on **`/work`**.

`UM_INSTALL_DIR` is set in the `suite.rc` file to be `UMDIR`, so you can make the required changes like this:

```
mkdir ukca_emiss
cd ukca_emiss
ln -s /work/n02/n02/hum/ancil/atmos/n48e/ukca_emiss/andres_kasgnoc
ln -s /work/n02/n02/hum/ancil/atmos/n48e/ukca_emiss/cmip5
ln -s /work/n02/n02/hum/ancil/atmos/n48e/ukca_emiss/gfed3.1
```

You can then either copy or symbolically link your newly-created emissions file into this directory as well.

This directory will need to be on `/work` as the batch queues are unable to see `/home`.

Virtual Machine

On the VM you do have write permissions for UMDIR, which is located at /home/vagrant/umdir, so you could either copy your new emissions file there, or follow the ARCHER instructions above.

Make your Rose changes

Once you have created the required directory structure you should change the value of **ukca_em_dir** to point to your new directory, and then go to the **ukca_em_files** list. Click the **blue plus symbol** to add an extra line, and put the path to the new file, relative to this directory.

If your emissions files are defined with the full path, and/or in a single long list, make the changes accordingly.

Code changes

ukca_chem_master.F90

You can choose to put a **1** in the **E** column of the **chch_defs** array. This is just for completeness however, as this column is not currently used.

ukca_setd1defs.F90

You will need to edit the **em_chem_spec** array for the scheme that you are using to include the new species that you are emitting into. By convention, these are ordered as 2D fields followed by 3D fields, although this order is unimportant other than the fact that the **'NO_aircrft'** entry **MUST** be last in the list.

If the tracer is valid for both with and without aerosol chemistry (& GLOMAP) you will need to put this in twice. Also remember to increment this size of the **em_chem_spec** array, given by **n_chem_emissions** variable.

ukca_constants.F90

You will need to define the **M_species** for the emitted species. This should be consistent with the **C_species** value set in the previous tutorial.

get_molmass_mod.F90

You will need to add to the **species_name** CASE statement, to include a line such as this

```
CASE ('ALICE     ')
  get_molmass = m_ALICE
```

for each new species that you are emitting into.

Emissions diagnostics

By default, when emitting into a new species, the code requires you to create a diagnostic for these emissions as well. This is done by considering the following file and routines:

STASHmaster_A and **STASHmaster-meta.conf**

You should add a diagnostic in a similar way to the section 34 tracers, but in this case to **section 50**. It is best to copy an existing 2D (or 3D) emissions diagnostic specification, and make only the required edits to that.

Once you have made these code changes you should **commit your branch** to ensure that these changes are picked-up by your suite correctly.

get_emdiag_stash_mod.F90

Once you have decided on the STASH code to use, you need to add the species and 3-digit **item** code to this routine in the CASE statement.

ukca_emdiags_struct_mod.F90

This routine contains the derived type that holds the emissions fluxes for all species. You will need to add a logical (used to determine whether the output has been requested in STASH) and a pointer to a 2D or 3D array that will hold the emissions flux.

ukca_update_emdiagstruct_mod.F90

In this routine you will need to add the initialisation of the logical added in ukca_emdiags_struct_mod.F90, as well as a small block of code that copies the 2D/3D em_diags data into the newly created pointer in the derived type. The logical will also need to be set to **.TRUE.** if STASH output has been requested (determined by the sf(item,section) look-up array).

Here is it easiest to copy an existing code-block and make the necessary changes for your new emission.

ukca_emiss_diags_mod.F90

In this routine you will add a unique code-block for this diagnostic that copies the pointer in the derived type into the stashwork array using the copydiag copydiag_3d routines. The stashwork array contains all the fields that will be output through STASH.

Here is it easiest to copy an existing code-block and make the necessary changes for your new emission.

ukca_emiss_ctl_mod.F90

The call to **ukca_emiss_diags** is protected by an IF statement, and you'll need to add a check to the entry in the **sf** array to see if the diagnostic is on. This should be of the format **sf(N,section)**, where **N** is the item number of your new emissions diagnostic.

Solution to Task 5.2

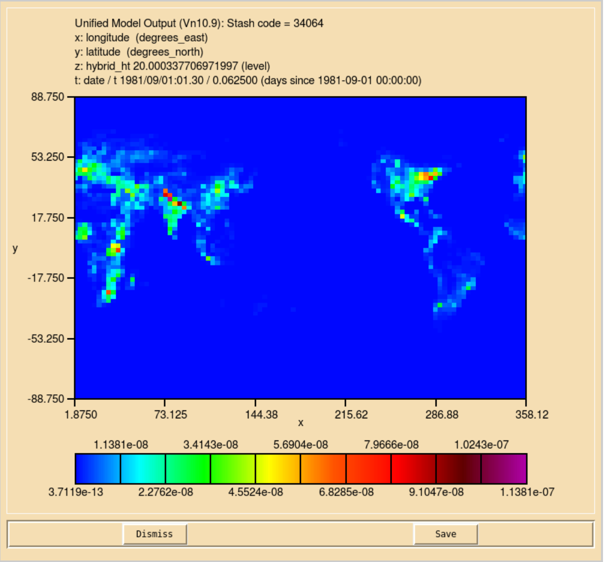
You were given the task

- *You should now make the UKCA code changes to add your emission into the ALICE tracer. Use STASH code s50i304 for the diagnosic.*

You were given the hints

- *You will need to add-in the molar mass of ALICE. You can calculate this from the mass of air and the conversion factor defined in Task 4.1.*
- *You will need to add code to allow a diagnostic of the emission to be output.*

Once you have made the required code and Rose changes, you should now see that you ALICE tracer (s34i064) no longer contains noise, but now looks a bit like the emissions field you created in Task 5.1.


Unified Model Output (Vn10.9): Stash code = 34064
x: longitude (degrees_east)
y: latitude (degrees_north)
z: hybrid_ht 20.000337706971997 (level)
t: date / t 1981/09/01:01.30 / 0.062500 (days since 1981-09-01 00:00:00)
88.750
53.250
17.750
-17.750
-53.250
-88.750
y
1.8750
73.125
144.38
215.62
286.88
358.12
x
1.1381e-08
3.4143e-08
5.6904e-08
7.9666e-08
1.0243e-07
3.7119e-13
2.2762e-08
4.5524e-08
6.8285e-08
9.1047e-08
1.1381e-07
Dismiss
Save

Surface level (20m) concentrations (kg/kg) of ALICE after emissions have been applied.

For a working Rose suite that has completed this task, please see

- **ARCHER:** `u-as292@59922`
- **vm:** `u-as297@59923`

The specific Rose changes made are:

- **ARCHER:** https://code.metoffice.gov.uk/trac/roses-u/changeset/59922/a/s/2/9/2/trunk
- **vm:**
  - https://code.metoffice.gov.uk/trac/roses-u/changeset/59921/a/s/2/9/7/trunk
  - https://code.metoffice.gov.uk/trac/roses-u/changeset/59923/a/s/2/9/7/trunk

The specific Rose changes made are:

ARCHER:

```
Index: app/um/rose-app.conf
==============================================================
--- app/um/rose-app.conf        (revision 59818)
+++ app/um/rose-app.conf        (revision 59922)
@@ -2638,7 +2638,7 @@
 ukca_cfc115mmr=4.584e-11
 ukca_ch2br2mmr=1.802e-11
 ukca_cosmmr=1.000e-9
-ukca_em_dir='$UM_INSTALL_DIR/ancil/atmos/n48e/ukca_emiss'
+ukca_em_dir='/work/n02/n02/ukca/Tutorial/vn10.9/ukca_emiss'
 ukca_em_files='gfed3.1/clim_2002_2011/v2/ukca_emiss_BC_biomas
             ='gfed3.1/clim_2002_2011/v2/ukca_emiss_OC_biomas
             ='andres_kasgnoc/v1/ukca_emiss_SO2_nat_L38.nc',
@@ -2661,7 +2661,8 @@
             ='cmip5/2000/v1/ukca_emiss_Me2CO.nc',
             ='cmip5/2000/v1/ukca_emiss_C5H8.nc',
             ='cmip5/2000/v1/ukca_emiss_NVOC.nc',
-            ='cmip5/2000/v1/ukca_emiss_CO.nc'
+            ='cmip5/2000/v1/ukca_emiss_CO.nc',
+            ='Task5.1/solution/ukca_emiss_ALICE.nc'
 ukca_h1202mmr=3.788e-13
 ukca_h1211mmr=2.225e-11
 ukca_h1301mmr=1.363e-11
Index: app/fcm_make/rose-app.conf
==============================================================
--- app/fcm_make/rose-app.conf  (revision 59818)
+++ app/fcm_make/rose-app.conf  (revision 59922)
@@ -42,4 +42,4 @@
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solr
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solr
```

These differences can be found in the file **`/home/ukca/Tutorial/vn10.9/worked_solutions/Task5.2/Task5.2_rose.patch`** on PUMA.

vm:

```
Index: app/um/rose-app.conf
===================================================================
--- app/um/rose-app.conf        (revision 59815)
+++ app/um/rose-app.conf        (revision 59923)
@@ -2641,7 +2641,7 @@
 ukca_cfc115mmr=4.584e-11
 ukca_ch2br2mmr=1.802e-11
 ukca_cosmmr=1.000e-9
-ukca_em_dir='$UMDIR/ancil/atmos/n48e/ukca_emiss'
+ukca_em_dir='/home/vagrant/ukca_emiss'
 ukca_em_files='gfed3.1/clim_2002_2011/v2/ukca_emiss_BC_biomass.nc',
             ='gfed3.1/clim_2002_2011/v2/ukca_emiss_OC_biomass.nc',
             ='andres_kasgnoc/v1/ukca_emiss_SO2_nat.nc',
@@ -2664,7 +2664,8 @@
             ='cmip5/2000/v1/ukca_emiss_Me2CO.nc',
              ='cmip5/2000/v1/ukca_emiss_C5H8.nc',
              ='cmip5/2000/v1/ukca_emiss_NVOC.nc',
-             ='cmip5/2000/v1/ukca_emiss_CO.nc'
+             ='cmip5/2000/v1/ukca_emiss_CO.nc',
+             ='Task5.1/solution/ukca_emiss_ALICE.nc'
 ukca_h1202mmr=3.788e-13
 ukca_h1211mmr=2.225e-11
 ukca_h1301mmr=1.363e-11
Index: app/fcm_make/rose-app.conf
===================================================================
--- app/fcm_make/rose-app.conf  (revision 59815)
+++ app/fcm_make/rose-app.conf  (revision 59923)
@@ -42,4 +42,4 @@
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46535
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46574
```

For a working UM branch that has completed this task, please see

The specific UM changes made are:

```

Index: src/atmosphere/UKCA/ukca_setd1defs.F90
===================================================================
--- src/atmosphere/UKCA/ukca_setd1defs.F90      (revision 46535)
+++ src/atmosphere/UKCA/ukca_setd1defs.F90      (revision 46574)
@@ -313,7 +313,7 @@
       nr_phot         = 38
     END IF
   ELSE IF (L_ukca_strattrop .AND. .NOT. L_ukca_achem) THEN
-    n_chem_emissions = 9
+    n_chem_emissions = 10
     n_3d_emissions = 1       ! aircraft NOX
     n_aero_tracers =  0
     ALLOCATE(em_chem_spec(n_chem_emissions+n_3d_emissions))
@@ -320,13 +320,13 @@
     em_chem_spec =                                           &
         (/'NO        ','CH4       ','CO        ','HCHO      ', &
           'C2H6      ','C3H8      ','Me2CO     ','MeCHO     ', &
-          'C5H8      ','NO_aircrft'/)
+          'C5H8      ','ALICE     ','NO_aircrft'/)
     n_chem_tracers = 73         ! No chem tracers
     nr_therm       = 220        ! thermal reactions
     nr_phot        = 55         ! photolytic (ATA)

   ELSE IF (L_ukca_strattrop .AND. L_ukca_achem) THEN
-    n_chem_emissions = 19      ! em_chem_spec below
+    n_chem_emissions = 20      ! em_chem_spec below
     n_3d_emissions   = 4       ! BC, OC, volc SO2 & aircraft NOX
     ALLOCATE(em_chem_spec(n_chem_emissions+n_3d_emissions))
     em_chem_spec =                                           &
@@ -335,7 +335,7 @@
           'C5H8      ','BC_fossil ','BC_biofuel','OC_fossil ', &
           'OC_biofuel','Monoterp  ','NVOC      ','SO2_low   ', &
           'SO2_high  ','NH3       ','DMS       ','SO2_nat   ', &
-          'BC_biomass','OC_biomass','NO_aircrft'/)
+          'BC_biomass','OC_biomass','ALICE     ','NO_aircrft'/)
     n_aero_tracers = 12
     n_chem_tracers = 73         ! No chem tracers
     IF (L_ukca_trophet) THEN
Index: src/atmosphere/UKCA/ukca_chem_master.F90
===================================================================
--- src/atmosphere/UKCA/ukca_chem_master.F90    (revision 46535)
+++ src/atmosphere/UKCA/ukca_chem_master.F90    (revision 46574)
@@ -377,7 +377,7 @@
 chch_t1(113,'MEMALD    ',1,'TR        ','          ',0,0,0,R,0,0,107),&
 chch_t1(114,'GLY       ',1,'TR        ','          ',0,1,0,R,0,0,107),&
 chch_t1(115,'oXYLENE   ',1,'TR        ','          ',0,0,1,R,0,0,107),&
-chch_t1(116,'ALICE     ',1,'TR        ','          ',0,0,0,ST,0,0,107),&
+chch_t1(116,'ALICE     ',1,'TR        ','          ',0,0,1,ST,0,0,107),&
 chch_t1(117,'BOB       ',1,'TR        ','          ',0,0,0,ST,0,0,107)/)

 ! Heterogeneous chemistry
Index: src/atmosphere/UKCA/get_molmass_mod.F90
===================================================================
--- src/atmosphere/UKCA/get_molmass_mod.F90     (revision 46535)
+++ src/atmosphere/UKCA/get_molmass_mod.F90     (revision 46574)
@@ -243,6 +243,11 @@
```

```
   get_molmass = m_brcl

   ! -----------------------------------------
+  !   UKCA Tutorial Tracer
+CASE ('ALICE     ')
+  get_molmass = m_ALICE
+
+  ! -----------------------------------------
   !   Others (report warning)
 CASE ('AGE       ')
   get_molmass =  1.0
Index: src/atmosphere/UKCA/ukca_constants.F90
===================================================================
--- src/atmosphere/UKCA/ukca_constants.F90      (revision 46535)
+++ src/atmosphere/UKCA/ukca_constants.F90      (revision 46574)
@@ -370,6 +370,8 @@
 REAL, PARAMETER :: m_mgly    = 72.0
 REAL, PARAMETER :: m_hacet  = 74.0

+!     UKCA Tutorial tracers - only ALICE is emitted
+REAL, PARAMETER :: m_ALICE    = 28.97

 !     Extra masses for EXTTC chemistry
 REAL, PARAMETER :: m_apin     =  136.0
Index: src/atmosphere/UKCA/ukca_emiss_diags_mod.F90
===================================================================
--- src/atmosphere/UKCA/ukca_emiss_diags_mod.F90        (revision 46535)
+++ src/atmosphere/UKCA/ukca_emiss_diags_mod.F90        (revision 46574)
@@ -584,6 +584,23 @@
   END IF
 END IF

+!----------------------------------------------------------------------
+! Sec 50, item 304: ALICE surface emissions
+item = get_emdiag_stash ('ALICE     ')
+IF (sf(item, section)) THEN
+
+  ! DEPENDS ON: copydiag
+  CALL copydiag (stashwork (si(item,section,im_index)),              &
+                 emdiags%em_ALICE (:,:),                             &
+                 row_length, rows, 0,0,0,0, at_extremity,            &
+                 atmos_im, section, item, icode, cmessage)
+
+  IF (icode >  0) THEN
+    errcode = section*1000+item
+    CALL ereport ('UKCA_EMISS_DIAGS', errcode, cmessage)
+  END IF
+END IF
+
 IF (lhook) CALL dr_hook(ModuleName//':'//RoutineName,zhook_out,zhook_handle)

 RETURN
Index: src/atmosphere/UKCA/ukca_emiss_ctl_mod.F90
===================================================================
--- src/atmosphere/UKCA/ukca_emiss_ctl_mod.F90  (revision 46535)
+++ src/atmosphere/UKCA/ukca_emiss_ctl_mod.F90  (revision 46574)
@@ -788,7 +788,8 @@
     sf(168,section) .OR. sf(169,section) .OR. sf(170,section) .OR.      &
     sf(171,section) .OR. sf(172,section) .OR. sf(211,section) .OR.      &
     sf(212,section) .OR. sf(213,section) .OR. sf(214,section) .OR.      &
-    sf(215,section) .OR. sf(216,section) .OR. sf(217,section) ) THEN
+    sf(215,section) .OR. sf(216,section) .OR. sf(217,section) .OR.      &
+    sf(304,section)) THEN

   CALL ukca_emiss_diags (row_length, rows, model_levels,                &
     len_stashwork50, stashwork50)
Index: src/atmosphere/UKCA/get_emdiag_stash_mod.F90
===================================================================
--- src/atmosphere/UKCA/get_emdiag_stash_mod.F90        (revision 46535)
+++ src/atmosphere/UKCA/get_emdiag_stash_mod.F90        (revision 46574)
@@ -142,6 +142,9 @@
 CASE ('SO2_nat   ')
   get_emdiag_stash = 217

+CASE ('ALICE     ')
+  get_emdiag_stash = 304
+
 CASE DEFAULT
   ! Report error unless this is an aerosol emission (BC_fossil:, etc)
   ! No diagnostics for these source emissions because the resulting
```

```
Index: src/atmosphere/UKCA/ukca_emdiags_struct_mod.F90
===================================================================
--- src/atmosphere/UKCA/ukca_emdiags_struct_mod.F90     (revision 46535)
+++ src/atmosphere/UKCA/ukca_emdiags_struct_mod.F90     (revision 46574)
@@ -53,6 +53,7 @@
   LOGICAL :: l_em_so2low
   LOGICAL :: l_em_so2hi
   LOGICAL :: l_em_so2nat
+  LOGICAL :: l_em_ALICE

   ! Pointers to hold emission diagnostics
   REAL, POINTER :: em_no     (:,:)
@@ -79,6 +80,7 @@
   REAL, POINTER :: em_so2low (:,:)
   REAL, POINTER :: em_so2hi  (:,:)
   REAL, POINTER :: em_so2nat (:,:,:)
+  REAL, POINTER :: em_ALICE  (:,:)

 END TYPE emdiags_struct

Index: src/atmosphere/UKCA/ukca_update_emdiagstruct_mod.F90
===================================================================
--- src/atmosphere/UKCA/ukca_update_emdiagstruct_mod.F90        (revision 46535)
+++ src/atmosphere/UKCA/ukca_update_emdiagstruct_mod.F90        (revision 46574)
@@ -111,6 +111,7 @@
   emdiags%l_em_so2low = .FALSE.
   emdiags%l_em_so2hi  = .FALSE.
   emdiags%l_em_so2nat = .FALSE.
+  emdiags%l_em_ALICE  = .FALSE.

   l_first             = .FALSE.
 END IF
@@ -399,6 +400,17 @@
     END IF
   END IF

+CASE ('ALICE       ')
+  IF (emdiags%l_em_ALICE) THEN
+    emdiags%em_ALICE (:,:) = em_diags (:,:,1)
+  ELSE
+    IF (sf(item,section)) THEN
+      ALLOCATE (emdiags%em_ALICE (row_length, rows))
+      emdiags%em_ALICE (:,:) = em_diags (:,:,1)
+      emdiags%l_em_ALICE     = .TRUE.
+    END IF
+  END IF
+
 CASE DEFAULT
     ! Report error unless this is an aerosol emission (BC_fossil:, etc)
   IF (.NOT. ANY(aero_ems_species == diag_name))  ierr = 1
Index: rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster-meta.conf
===================================================================
--- rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster-meta.conf (revision 46535)
+++ rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster-meta.conf (revision 46574)
@@ -22211,6 +22211,10 @@
     =
     =Available if chemistry is turned on in UKCA

+[stashmaster:code(50304)]
+description=alice emission flux (kg m-2 s-1)
+help=Diagnostic to check the total ALICE surface emission flux
+
 [stashmaster:code(51001)]
 description=O3 MASS MIXING RATIO ON PRESS LEVS
 help=Ozone Mass Mixing Ratio in kg/kg(Air)
Index: rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster_A
===================================================================
--- rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster_A (revision 46535)
+++ rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster_A (revision 46574)
@@ -24236,6 +24236,12 @@
 4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
 5|    0 | 3834 |    0 |  129 |    0 |    0 |    0 |    0 |    0 |
 #
+1|    1 |   50 |  304 |ALICE surf emissions (kg m-2 s-1)    |
+2|    0 |    0 |    1 |    1 |    5 |   -1 |   -1 |    0 |    0 |    0 |    0 |
+3| 000000000000000000000000010000 | 00000000000000000001 |    3 |
+4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
+5|    0 | 1876 |    0 |  129 |    0 |    0 |    0 |    0 |    0 |
+#
 #==============================================================================
 # Section 50 Item 999 reserved.
 #==============================================================================
```

These differences can be found in the file **`/home/ukca/Tutorial/vn10.9/worked_solutions/Task5.2/Task5.2_code.patch`** on PUMA.

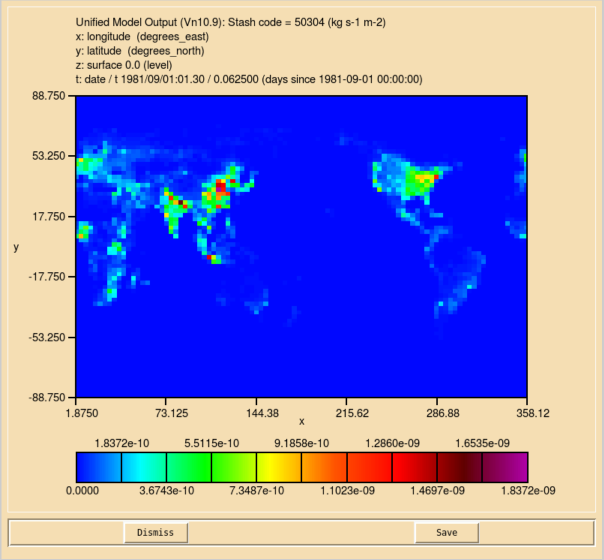
Sample output from this task can be found at **`/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task5.2/atmosa.pa19810901_00`** on ARCHER.

Task 5.3: Output the diagnostic of your new emission

Task 5.3: Now that you are able to run the model with your new emission file, output this as a diagnostic to the **UPA** stream as a 3-hour mean.

You should reference the what is STASH tutorial for information on how to do this

Solution to Task 5.3



ALICE emission flux diagnostic output.

You were given the task:

- *Now that you are able to run the model with your new emission file, output this as a diagnostic to the **UPA** stream as a 3-hour mean.*

This achieved by outputting s50i304 to the UPA stream in STASH.

You will now find the additional field in your output file:

```
23   : 96    72    1     1     mflux: Stash code 50304
```

For a working Rose suite that has completed this task, please see

- **ARCHER:** `u-as292@59937`
- **vm:** `u-as297@59931`

The specific Rose changes made are:

- **ARCHER:** https://code.metoffice.gov.uk/trac/roses-u/changeset/59937/a/s/2/9/2/trunk
- **vm:** https://code.metoffice.gov.uk/trac/roses-u/changeset/59931/a/s/2/9/7/trunk

The specific Rose changes made are:

ARCHER:

```
Index: app/um/rose-app.conf
===================================================================
--- app/um/rose-app.conf        (revision 59922)
+++ app/um/rose-app.conf        (revision 59937)
@@ -3449,6 +3449,14 @@
 tim_name='T3HMN'
 use_name='UPA'

+[namelist:umstash_streq(50304_f1d61e7a)]
+dom_name='DIAG'
+isec=50
+item=304
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
 [namelist:umstash_streq(51001_3e6241a4)]
 dom_name='DP27CCM'
 isec=51
```

These differences can be found in the file **`/home/ukca/Tutorial/vn10.9/worked_solutions/Task5.3/Task5.3_rose.patch`** on PUMA.

vm:

```
Index: app/um/rose-app.conf
===================================================================
--- app/um/rose-app.conf        (revision 59923)
+++ app/um/rose-app.conf        (revision 59931)
@@ -4041,6 +4041,14 @@
 tim_name='T3HMN'
 use_name='UPA'

+[namelist:umstash_streq(50304_f1d61e7a)]
+dom_name='DIAG'
```

```
+isec=50
+item=304
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
 [namelist:umstash_streq(51001_3e6241a4)]
 dom_name='DP27CCM'
 isec=51
```

Sample output from this task can be found at **/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task5.3/atmosa.pa19810901_00** on ARCHER.

Checklist

□ Obtain emissions data for the species of interest.

□ Regrid the emissions to the correct MetUM resolution that you are using.

□ Save these emissions as netCDF, including the required metadata that UKCA requires.

□ In your suite, include this file in the `ukca_em_files` variable in the UKCA panel: um ⟶ namelist ⟶ UM Science Settings ⟶ Section 34 - UKCA: UK Aerosols and Chemistry.

□ (Optional) Put a 1 in the E column of the `chch_defs_master` in `ukca_chem_master.F90`.

□ Put the species being emitted into the `em_chem_spec` array in `ukca_setd1defs.F90`, and increment `n_chem_emissions`.

□ Put the correct M_species value in `ukca_constants.F90`.

□ In `get_molmass_mod.F90`, append the CASE statement with your new species.

□ Make diagnostics output slots for your new emissions in `STASHmaster_A` and add help-text to `STASHmaster-meta.conf`.

□ In `get_emdiag_stash_mod.F90` add the species and 3-digit item code to the CASE statement.

□ In `ukca_emdiags_struct_mod.F90` add a logical and array pointer to the derived type for your new species.

□ In `ukca_update_emdiagstruct_mod.F90` initialise your new derived type entries, and add a block of code to copy the emissions field into the diagnostic pointer.

□ In `ukca_emiss_diags_mod.F90` add a code block to copy the field to the `stashwork` array using `copydiag/copydiag_3d`.

□ In `ukca_emiss_ctl_mod.F90` add the `sf(ITEM,section)` check in the IF statement protecting the call to `ukca_emiss_diags`.

□ `fcm commit` the changes to your branch.

□ Output your emissions diagnostic in: um ⟶ namelist ⟶ Model Input and Output ⟶ STASH Requests and Profiles ⟶ STASH Requests.

□ Run the TidyStashTransform transform macro.

□ Save your suite.

□ In the `roses/[SUITE-ID]` directory, run `fcm commit` to commit your changes to the repository.

□ Run your suite.

Tutorial 6

Written by Luke Abraham 2017. Many thanks to Alistair Sellar for the notes on netCDF metadata.

Retrieved from "http://www.ukca.ac.uk/wiki/index.php?title=UKCA_Chemistry_and_Aerosol_vn10.9_Tutorial_5&oldid=6749"

- This page was last modified on 9 January 2018, at 15:42.

UKCA Chemistry and Aerosol vn10.9 Tutorial 6

From UKCA

UKCA Chemistry and Aerosol Tutorials at vn10.9

Contents

What you will learn in this tutorial

During this tutorial you will learn how UKCA specifies different chemical reactions. You will then add a new reaction involving the new tracers that you have added.

Task 6.1: Add a bimolecular reaction

TASK 6.1: You should now add in the bimolecular reaction of **ALICE** with **OH** to form **BOB** and a ***secondary organic compound*** (labelled in UKCA as **Sec_Org**). This reaction is given by:

$$\mathrm{ALICE} + \mathrm{OH} \longrightarrow \mathrm{BOB} + \mathrm{Sec\_Org}$$

Parameter	Value
k_0	2.70E-11
α	0.00
β	-390.00

Adding new Chemical Reactions

UKCA currently uses two different methods of defining the chemical reactions solved in the model. The first is a backward Euler solver, and is used for the *RAQ* and *StdTrop* chemistry schemes where the solver itself is created by a code-writer. The second makes use of the ASAD chemical integration software package (http://www.atm.ch.cam.ac.uk/acmsu/asad/) , and is used for the *CheT/TropIsop*, *CheS/Strat*, and *CheST/StratTrop* chemistry schemes. ASAD can use many different solvers, although currently it uses a symbolic Newton-Raphson solver. In this tutorial we will only consider the ASAD framework, as this is easily extended by a user.

ASAD considers four different types of chemical reactions: bimolecular reactions, termolecular reactions, heterogeneous reactions, and photolysis reactions. To make changes and add reactions you will need to make changes to the UKCA source code which can be found in

```
vn10.9_your_branch_name/src/atmosphere/UKCA
```

During this tutorial you will be tasked with adding a new reaction into your branch.

Biomolecular Reactions

For most bimolecular reactions, it is sufficient to provide the k_0, α, and β coefficients that are used to compute the rate coefficient k from the Arrhenius expression

$$k = k_0 \left(\frac{T}{300}\right)^{\alpha} \exp\left(\frac{-\beta}{T}\right)$$

Bimolecular Reaction Definition

The bimolecular reactions are defined in the **`ukca_chem_master.F90`** module using the **`ratb_t1`** Fortran type specification, and are held in arrays. At the end of this routine the **`ratb_defs_master`** array is created from these, and if that scheme is selected in UKCA these reactions are copied across into the master **`ratb_defs`** array.

The format of this **`ratb_t1`** type (defined in **`ukca_chem_defs_mod.F90`**) is

```
ratb_t1(N, 'Reactant 1','Reactant 2','Product 1 ','Product 2 ','Product 3 ',&
'Product 4 ',  k0,  α,  β, Fraction of Product 1 produced, Fraction of Product 2 produced,
Fraction of Product 3 produced, Fraction of Product 4 produced, SCHEME, QUALIFIER, DISQUALIFIER, VN), &
```

If fractional products are not required for a reaction, then the *fraction of each product* formed should be set to 0.00. If fractional products are required for any one of the products then the fraction of each product formed should be set to its correct value.

The settings for `N`, `SCHEME`, `QUALIFIER`, `DISQUALIFIER`, and `VN` are the same as in the adding new tracers tutorial, although here **`N`** should be incremented for each new **reaction**, where there might be the same reaction specified several times with changes to reaction rates or even species.

The specifications of the individual reactions are done as, e.g.

```
! B041 IUPAC2005
ratb_t1(41,'EtOO      ','NO        ','MeCHO     ','HO2       ','NO2       ',&
'          ',2.55e-12,  0.00, -380.00, 0.00, 0.00, 0.00, 0.00,ST+R,0,0,107),&
ratb_t1(41,'EtOO      ','NO        ','MeCHO     ','HO2       ','NO2       ',&
'          ',2.60e-12,  0.00, -380.00, 0.00, 0.00, 0.00, 0.00,TI+T,0,0,107),&
! B060 JPL2011
ratb_t1(60,'HO2       ','NO        ','OH        ','NO2       ','          ',&
'          ',3.30e-12,  0.00, -270.00, 0.00, 0.00, 0.00, 0.00,ST+R,0,0,107),&
ratb_t1(60,'HO2       ','NO        ','OH        ','NO2       ','          ',&
'          ',3.50e-12,  0.00, -250.00, 0.00, 0.00, 0.00, 0.00,S,0,0,107),&
ratb_t1(60,'HO2       ','NO        ','OH        ','NO2       ','          ',&
'          ',3.60e-12,  0.00, -270.00, 0.00, 0.00, 0.00, 0.00,TI+T,0,0,107),&
! B132 JPL2011
! Not in TI/R scheme
ratb_t1(132,'O(3P)     ','OH        ','O2        ','HO2           ','           ',&
'          ',1.80e-11,  0.00, -180.00, 0.00, 0.00, 0.00, 0.00,T,0,0,107),&
ratb_t1(132,'O(3P)     ','OH        ','O2        ','H         ','           ',&
'          ',1.80e-11,  0.00, -180.00, 0.00, 0.00, 0.00, 0.00,ST,0,0,107),&
ratb_t1(132,'O(3P)     ','OH        ','O2        ','H         ','           ',&
'          ',2.40e-11,  0.00, -110.00, 0.00, 0.00, 0.00, 0.00,S,0,0,107),&
```

The first reaction in these examples takes its kinetic data from IUPAC (http://www.iupac-kinetic.ch.cam.ac.uk/) . Going to this website, this reaction is defined here (http://www.iupac-kinetic.ch.cam.ac.uk/datasheets/xhtml/HOx_VOC8_HO_CH2C%28CH3%29CHCH2%28isoprene%29.xhtml_mathml.xml) . The second and third reaction above take their kinetic data from NASA's Jet Propulsion Laboratory (http://jpldataeval.jpl.nasa.gov/) . The rate for this can be found on page 1-19 of the JPL2011 document (http://jpldataeval.jpl.nasa.gov/pdf/JPL%2010-6%20Final%2015June2011.pdf) . You can see that in the instances above, different chemistry schemes use slightly different rates or species, as may be required by the scheme and species considered.

When adding new reactions you will need to increment the size of the array holding the `ratb_t1` type.

To add new bimolecular reactions you will need to append equivalent lines for the new reactions to the end of the **`ratb_defs_master`** array (increasing the array sizes accordingly). If there is a reaction that is an exception to the general Arrhenius equation then special code needs to be placed in the **`asad_bimol.F90`** routine, which is held in the `UKCA/` source-code directory.

Termolecular Reactions

As well as defining reactions involving a third body, the termolecular rate definition can also be used to define unimolecular reactions.

The pressure and temperature dependent rate, k, of a termolecular reaction is given by

$$k = \left(\frac{k_0\,[M]}{1 + k_0\,[M]\,/k_\infty} \right) F_c^{\left(1+\left[\log_{10}\left(\frac{k_0[M]}{k_\infty}\right)\right]^2\right)^{-1}}$$

where the low pressure rate constant k_0 is given by

$$k_0 = k_1 \left(\frac{T}{300}\right)^{\alpha_1} \exp\left(\frac{-\beta_1}{T}\right)$$

and the high pressure rate constant k_∞ is given by

$$k_\infty = k_2 \left(\frac{T}{300}\right)^{\alpha_2} \exp\left(\frac{-\beta_2}{T}\right)$$

Termolecular Reaction Definition

The termolecular reactions are defined in the **`ukca_chem_master.F90`** module using the **`ratt_t1`** Fortran type specification and are held in the **`ratt_defs_master`** array.

To format of this **`ratt_t1`** type is

```
ratt_t(N,'Reactant 1','Reactant 2','Product 1 ','Product 2 ', $f$, &
$k_1$,  $\alpha_1$,  $\beta_1$, $k_2$,  $\alpha_2$,  $\beta_2$, Fraction of Product 1 produced, Fraction of Product 2 produced,SCHEME,QUALIFIER,
DISQUALIFIER,VN), &
```

and as in `rabt_t1`, where the fraction of a product should be set to 0.0 if this functionality does not need to be used.

The settings for `N`, `SCHEME`, `QUALIFIER`, `DISQUALIFIER`, and `VN` are the same as in the adding new tracers tutorial, although here **`N`** should be incremented for each new **reaction**, where there might be the same reaction specified several times with changes to reaction rates or even species.

The f value is used to define the F_c value by

> If $f < 1.0$ then $F_c = f$
>
> else $$F_c = \exp\left(-T/f\right)$$

as F_c may or may not be highly temperature dependent.

Examples of these reactions are

```
! T024 IUPAC 2001
ratt_t1(24,'NO        ','NO        ','NO2       ','NO2       ',     0.0,&
  3.30e-39,  0.00,-530.,  0.00e+00,  0.0, 0.0, 0.0, 0.0, T+ST,0,0,107),&
! B; not in TI/TI scheme
ratt_t1(24,'NO        ','NO        ','NO2       ','NO2       ',     0.0,&
  6.93e-40,  0.00,-530.,  0.00e+00,  0.0, 0.0, 0.0, 0.0, S,0,0,107),&
ratt_t1(25,'SO2       ','OH        ','SO3       ','HO2       ',     0.6,&
  3.00e-31, -3.30, 0.00,  1.50e-12,  0.0, 0.0, 0.0, 0.0, ST+S,A,0,107),&
ratt_t1(25,'SO2       ','OH        ','HO2       ','H2SO4     ',     0.6,&
  3.00e-31, -3.30, 0.00,  1.50e-12,  0.0, 0.0, 0.0, 0.0, TI,A,0,107),&
ratt_t1(25,'SO2       ','OH        ','H2SO4     ','          ',     0.6,&
  3.00e-31, -3.30, 0.00,  1.50e-12,  0.0, 0.0, 0.0, 0.0, OL,A,0,107),&
```

To add new termolecular reactions you will need to append equivalent lines for the new reactions to the end of the **`ratt_defs_master`** array (increasing the array sizes accordingly). If there is any special code that needs to be added, this should be placed in the **`asad_trimol.F90`** routine, which is held in the `UKCA/` source-code directory.

Heterogeneous Reactions

Heterogeneous reactions are those that occur on aerosol surfaces. There is no functional form defined for these reactions, with special code needed to be added for each case.

Heterogeneous Reaction Definition

The heterogeneous reactions are defined in the **`ukca_chem_master.F90`** module using the **`rath_t1`** Fortran type specification, usually in one array (**`rath_defs_master`**).

To format of this **`rath_t1`** type is

```
rath_t(N,'Reactant 1','Reactant 2','Product 1 ','Product 2 ','Product 3 ',&
'Product 4 ', Fraction of Product 1 produced, Fraction of Product 2 produced, Fraction of Product 3 produced,
Fraction of Product 4 produced,SCHEME,QUALIFIER,DISQUALIFIER,VN), &
```

i.e. there is no rate information provided. For reactions on PSCs special code has been added to the routines in **`ukca_hetero_mod.F90`**, and for other reactions there is code in **`asad_hetero.F90`**.

The settings for `N`, `SCHEME`, `QUALIFIER`, `DISQUALIFIER`, and `VN` are the same as in the adding new tracers tutorial, although here **`N`** should be incremented for each new **reaction**, where there might be the same reaction specified several times with changes to reaction rates or even species.

Examples of this type are

```
nrath_t1(2,'ClONO2    ','HCl       ','Cl        ','Cl        ','HONO2     ', &
'          ', 0.000, 0.000, 0.000, 0.000, S+ST,HP,0,107), &
!HSO3+H2O2(aq)
rath_t1(6,'SO2       ','H2O2      ','NULL0     ','          ','          ', &
'          ', 0.000, 0.000, 0.000, 0.000, TI+S+ST+OL+R,A,0,107),            &
```

To add new heterogeneous reactions you will need to append equivalent lines for the new reactions to the end of the array (increasing the array sizes accordingly), before adding code to either **`ukca_hetero_mod.F90`** (for stratospheric reactions) or **`asad_hetero.F90`** (for tropospheric reactions).

Photolysis Reactions

These define a reaction where a chemical compound is broken down by photons. There is no functional form defined for this type of reaction. Instead, either (in the troposphere) input files are used to define the reaction rates for each species, while (in the stratosphere) on-line look-up tables are generated for the rates for each species, or a separate photolysis code, **Fast-JX**, is used to interactively calculate the rate of reaction throughout the the whole atmosphere (for Fast-JX). These interactive schemes are preferred as they take the effect of aerosols or clouds into account at each timestep, allowing for more feedbacks to be investigated. In the upper stratosphere there are some wavelength regions that Fast-JX does not consider, and so the 3D on-line look-up tables are also used for these regions.

Tropospheric Off-Line Photolysis

If Fast-JX is not being used, then the off-line two-dimensional (zonally average) tropospheric photolysis is used (for all schemes). It is based on the work of Hough (1988)[1] and Law *et al* (1998)[2].

This scheme makes use of datafiles which define the reaction rate for a particular species (e.g. H2O2), or if no rate is known, a **nil** rate can be used. For vn10.4 these files can be found in

```
$UMDIR/vn10.4/ctldata/UKCA/tropdata/photol
```

To use this scheme set the value of `i_ukca_photol` by clicking **2D Photolysis Scheme**. You will then need to give the location of the files (above). The code controlling this scheme is held in **`ukca_phot2d.F90`**.

It is advised that this scheme is no longer used, and Fast-JX interactive photolysis should be used instead.

References

1. Hough, A. M.: The calculation of photolysis rates for use in global modelling studies, Tech. rep., UK Atomic Energy Authority, Harwell, Oxon., UK, 1988
2. Law, K., Plantevin, P., Shallcross, D., Rogers, H., Pyle, J., Grouhel, C., Thouret, V., and Marenco, A.: Evaluation of modeled O3 using Measurement of Ozone by Airbus In-Service Aircraft (MOZAIC) data, J. Geophys. Res., 103, 25721–25737, 1998

Stratospheric Look-Up Table Photolysis

In a chemistry scheme which has stratospheric chemistry, such as *CheS/Strat* and *CheST/StratTrop*, if interactive photolysis is not used, then above 300hPa the look-up table approach of Lary and Pyle (1991)[1] is used (below 300hPa the tropospheric scheme described above is used). To use this scheme set the value of `i_ukca_photol` by clicking **2D Photolysis Scheme**. The code for this scheme is held in **`ukca_strat_update.F90`**.

References

1. Lary, D. and Pyle, J.: Diffuse-radiation, twilight, and photochemistry, J. Atmos. Chem., 13, 393–406, 1991.

Interactive Photolysis

The original Fast-J scheme (Wild *et al*, 2000)[1] uses 7 different wavelength bins appropriate for the troposphere, and the updated Fast-JX scheme (Neu et al, 2007)[2] adds up to an extra 11 bins allowing use in the stratosphere. At vn10.4 only Fast-JX is available, although previous UM version used Fast-J as well.

To use this scheme set the value of **`i_ukca_photol`** by clicking **FastJ-X**. You will then need to give the location of several input data files used by this scheme.

Further details on how the the Fast-JX scheme is used in UKCA, can be found in Telford *et al* (2013) (http://www.geosci-model-dev.net/6/161/2013/gmd-6-161-2013.html) [3].

The Fast-JX data files are held in

```
$UMDIR/vn10.9/ctldata/UKCA/fastj
```

on ARCHER.

References

1. Wild, O., Zhu, X., and Prather, M.: Fast-J: accurate simulation of in- and below-cloud photolysis in tropospheric chemical models, J. Atmos. Chem., 37, 245–282, doi:10.1023/A:1006415919030, 2000
2. Neu, J., Prather, M., and Penner, J.: Global atmospheric chemistry: integrating over fractional cloud cover, J. Geophys. Res., 112, D11306, 12 pp., doi:10.1029/2006JD008007, 2007
3. Telford, P. J., Abraham, N. L., Archibald, A. T., Braesicke, P., Dalvi, M., Morgenstern, O., O'Connor, F. M., Richards, N. A. D., and Pyle, J. A.: Implementation of the Fast-JX Photolysis scheme (v6.4) into the UKCA component of the MetUM chemistry-climate model (v7.3), Geosci. Model Dev., 6, 161-177, doi:10.5194/gmd-6-161-2013, 2013.

Photolysis Reaction Definition

The photolysis reactions are defined in the **`ukca_chem_master.F90`** module using the **`ratj_t1`** Fortran type specification and held in the **`ratj_defs_master`** array.

To format of this **`ratj_t1`** type is

```
ratj_t1(N,'Reactant 1','Reactant 2','Product 1 ','Product 2 ','Product 3 ',&
'Product 4 ', Fraction of Product 1 produced, Fraction of Product 2 produced, Fraction of Product 3 produced,
Fraction of Product 4 produced, Quantum Yield, Look-up Label,SCHEME,QUALIFIER,DISQUALIFIER,VN), &
```

The **Look-Up Label** is used to define the file used for the 2D photolysis, and is used by Fast-JX to find the correct values for each species in the input data files. This is a 10-character string, although only the first **7** characters are read by Fast-JX. *Reactant 2* will always be *PHOTON*.

The settings for `N`, `SCHEME`, `QUALIFIER`, `DISQUALIFIER`, and `VN` are the same as in the adding new tracers tutorial, although here **`N`** should be incremented for each new **reaction**, where there might be the same reaction specified several times with changes to reaction rates or even species.

Examples of this type are

```
! 3
! This should produce H+ CHO -> H + HO2 + CO in ST scheme.
ratj_t1(3,'HCHO      ','PHOTON    ','HO2       ','HO2       ','CO        ',&
     '          ', 0.0,0.0,0.0,0.0, 100.000,'jhchoa    ',TI+T+ST+R,0,0,107) ,&
ratj_t1(3,'HCHO      ','PHOTON    ','H         ','CO        ','HO2       ',&
     '          ', 0.0,0.0,0.0,0.0, 100.000,'jhchoa    ',S,0,0,107) ,&
```

```
! 4
ratj_t1(4,'HCHO      ','PHOTON    ','H2        ','CO        ','          ',&
     '          ', 0.0,0.0,0.0,0.0, 100.000,'jhchob    ',TI+S+T+ST+R,0,0,107) ,&
```

Solution to Task 6.1: Add a bimolecular reaction

You were given the task

- *You should now add in the bimolecular reaction of **ALICE** with **OH** to form **BOB** and a* **secondary organic compound** *(labelled in UKCA as **Sec_Org**). This reaction is given by:*

$$\mathrm{ALICE} + \mathrm{OH} \longrightarrow \mathrm{BOB} + \mathrm{Sec\_Org}$$

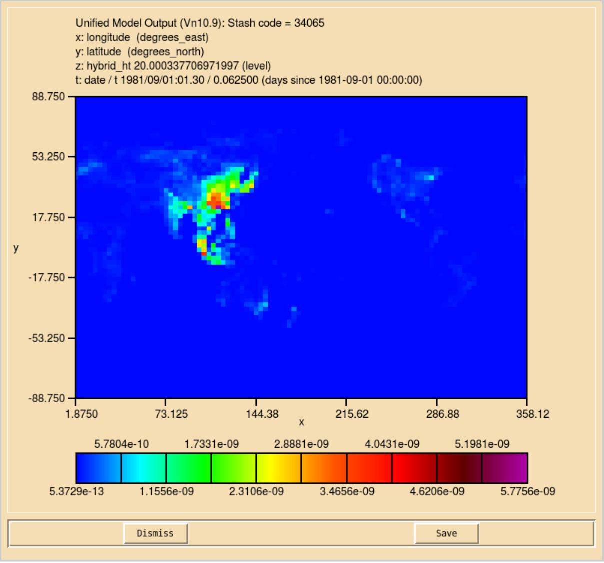
Parameter	Value
k_0	2.70E-11
α	0.00
β	-390.00

For a working Rose suite that has completed this task, please see

- **ARCHER:** `u-as292@60067`
- **vm:** `u-as297@59986`

The specific Rose changes made are:

- **ARCHER:** https://code.metoffice.gov.uk/trac/roses-u/changeset/60067/a/s/2/9/2/trunk
- **vm:** https://code.metoffice.gov.uk/trac/roses-u/changeset/59986/a/s/2/9/7/trunk



Surface level concentrations (kg/kg) of BOB after the reaction has been applied.

The specific Rose changes made are:

ARCHER:

```
Index: app/fcm_make/rose-app.conf
===================================================================
--- app/fcm_make/rose-app.conf  (revision 59937)
+++ app/fcm_make/rose-app.conf  (revision 60067)
@@ -42,4 +42,4 @@
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46574
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46619
```

These differences can be found in the file **`/home/ukca/Tutorial/vn10.9/worked_solutions/Task6.1/Task6.1_rose.patch`** on PUMA.

vm:

```
Index: app/fcm_make/rose-app.conf
===================================================================
--- app/fcm_make/rose-app.conf  (revision 59931)
+++ app/fcm_make/rose-app.conf  (revision 59986)
@@ -42,4 +42,4 @@
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46574
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46619
```

The specific UM changes made are:

```
Index: src/atmosphere/UKCA/ukca_chem_master.F90
===================================================================
--- src/atmosphere/UKCA/ukca_chem_master.F90    (revision 46574)
+++ src/atmosphere/UKCA/ukca_chem_master.F90    (revision 46619)
@@ -83,7 +83,7 @@
 INTEGER, PARAMETER :: n_het_master   =  10 ! number of heterogeneous reactions
 INTEGER, PARAMETER :: n_dry_master   =  57 ! number of dry deposition reactions
 INTEGER, PARAMETER :: n_wet_master   =  49 ! number of wet deposition reactions
-INTEGER, PARAMETER :: n_bimol_master= 400 ! number of bimolecular reactions
+INTEGER, PARAMETER :: n_bimol_master= 401 ! number of bimolecular reactions
 INTEGER, PARAMETER :: n_ratj_master =  76 ! number of photolysis reactions
 INTEGER, PARAMETER :: n_ratt_master =  49 ! number of termolecular reactions

@@ -2152,7 +2152,9 @@
```

```
 ratb_t1(276,'MACRO2    ','MeOO      ','MGLY      ','HACET     ','MeCO3     ',&
 'HCHO      ',1.00e-12,  0.00,    0.00, 1.00, 0.75, 0.25, 2.75, TI,0,0,107),&
 ratb_t1(277,'MACRO2    ','MeOO      ','HO2       ','CO        ','          ',&
-'          ',1.00e-12,  0.00,    0.00, 1.17, 0.25, 0.00, 0.00, TI,0,0,107) /)
+'          ',1.00e-12,  0.00,    0.00, 1.17, 0.25, 0.00, 0.00, TI,0,0,107),&
+ratb_t1(278,'ALICE     ','OH        ','BOB       ','Sec_Org   ','          ',&
+'          ',2.70E-11,  0.00, -390.00, 0.00, 0.00, 0.00, 0.00, ST,0,0,107) /)

 !----------------------------------------------------------------------
 ! NOTES: CheST Bimolecular Reactions
```

These differences can be found in the file **/home/ukca/Tutorial/vn10.9/worked_solutions/Task6.1/Task6.1_code.patch** on PUMA.

Sample output from this task can be found at **/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task6.1/atmosa.pa19810901_00** on ARCHER.

Checklist

□ Add the new reaction into the correct reaction type array in `ukca_chem_master.F90`, incrementing the size of the array accordingly.

□ If required, add special code to the `asad_bimol.F90`, `asad_trimol.F90`, `ukca_hetero_mod.F90`, or `asad_hetero.F90` routines.

□ For photolysis reactions, further work is required to calculate new cross sections. Code will also need to be added to `ukca_strat_update.F90`.

Tutorial 7

Written by Luke Abraham 2017

Retrieved from "http://www.ukca.ac.uk/wiki/index.php?title=UKCA_Chemistry_and_Aerosol_vn10.9_Tutorial_6&oldid=6716"

- This page was last modified on 3 January 2018, at 11:45.

UKCA Chemistry and Aerosol vn10.9 Tutorial 7

From UKCA

UKCA Chemistry and Aerosol Tutorials at vn10.9

Contents

What you will learn in this Tutorial

In this tutorial you will learn how the two UKCA dry deposition schemes are implemented. You will then make changes to allow one of your new tracers to be dry-deposited.

Task 7.1: adding new dry deposition values

TASK 7.1: You should now add in the dry deposition of **ALICE**. This species deposits in a similar way to **CO**. The values for **depvel_defs_strattrop** are:

Surface Type	Summer (day)	Summer (night)	Summer (24h ave)	Winter (day)	Winter (night)	Winter (24h ave)
Water	0.00	0.00	0.00	0.00	0.00	0.00
Forest	0.03	0.03	0.03	0.03	0.03	0.03
Grass	0.03	0.03	0.03	0.03	0.03	0.03
Desert	0.03	0.03	0.03	0.03	0.03	0.03
Ice	0.00	0.00	0.00	0.00	0.00	0.00

i.e. the same as for CO.

Hint [hide]
You will need to make changes for both dry deposition schemes. The changes to **ukca_surfddr.F90** can be made very easily by adding **ALICE** to the **CO** block in the `CASE` statement.

Adding Dry Deposition

UKCA uses two different dry-deposition schemes:

- A simple 2D parameterisation described by Giannakopoulos (1999)[1], Ganzeveld and Lelieveld (1995)[2], and Sander and Crutzen (1996)[3].
- A more detailed interactive parameterisation, based on the Wesely scheme (Wesely, 1989; Sanderson 2007)[4,5]

The default scheme is the interactive scheme, which is chosen by setting `l_ukca_intdd` to **true** in the UKCA panel.

Note: If you are using the interactive scheme and wish to add new values to it, you will also need to add values to the 2D scheme as well, otherwise you will get an error.

During this tutorial you will be tasked with adding the dry deposition of one of your new tracers.

References

1. Giannakopoulos, C., M. P. Chipperfield, K. S. Law, and J. A. Pyle (1999), Validation and intercomparison of wet and dry deposition schemes using 210Pb in a global three-dimensional off-line chemical transport model, J. Geophys. Res., 104(D19), 23761–23784, doi:10.1029/1999JD900392.
2. Ganzeveld, L., and J. Lelieveld (1995), Dry deposition parameterization in a chemistry general circulation model and its influence on the distribution of reactive trace gases, J. Geophys. Res., 100(D10), 20999–21012, doi:10.1029/95JD02266.
3. Sander, R., and P. J. Crutzen (1996), Model study indicating halogen activation and ozone destruction in polluted air masses transported to the sea, J. Geophys. Res., 101(D4), 9121–9138, doi:10.1029/95JD03793.
4. M.L. Wesely, Parameterization of surface resistances to gaseous dry deposition in regional-scale numerical models, Atmospheric Environment (1967), Volume 23, Issue 6, 1989, Pages 1293-1304, ISSN 0004-6981, http://dx.doi.org/10.1016/0004-6981(89)90153-4.
5. Sanderson, M. G., Collins, W. J., Hemming, D. L. and Betts, R. A. (2007), Stomatal conductance changes due to increasing carbon dioxide levels: Projected impact on surface ozone levels. Tellus B, 59: 404–411. doi: 10.1111/j.1600-0889.2007.00277.x

Chemistry Scheme Specification

The default is to use the 2D scheme, although it is advisable to use the interactive scheme. Within the UKCA code, whether a species is dry deposited or not is controlled in the **`ukca_chem_master.F90`** file. In the **`chch_defs_master`** array there are lines like

```
!  10 DD: 7,WD: 4,
chch_t1(10,'HONO2     ',1,'TR        ','          ',1,1,0,TI+S+T+ST+R,0,0,107),&
!  11 DD: 8,WD: 5,
chch_t1(11,'H2O2      ',1,'TR        ','          ',1,1,0,TI+S+T+ST+OL+R,0,0,&
   107),&
```

Where the **1** in the 6th column turns on dry deposition of that species (being 0 otherwise). You will need to change the 0 to a **1** for the species that you wish to now dry deposit.

2D Dry Deposition Scheme

The deposition velocities for the 2D scheme are defined in the **`depvel_defs_master`** array, which is held in the **`ukca_chem_master.F90`** module. This is a large derived type containing a length 30 array, usually formatted to be made up of size (6,5) blocks (for ease of reading), of the format

`N`	`'SPECIES  '`				
Summer (day) velocity over water	Summer (night) velocity over water	Summer (24h ave.) velocity over water	Winter (day) velocity over water	Winter (night) velocity over water	Winter (24h ave.) velocity over water
Summer (day) velocity over forest	Summer (night) velocity over forest	Summer (24h ave.) velocity over forest	Winter (day) velocity over forest	Winter (night) velocity over forest	Winter (24h ave.) velocity over forest
Summer (day) velocity over grass	Summer (night) velocity over grass	Summer (24h ave.) velocity over grass	Winter (day) velocity over grass	Winter (night) velocity over grass	Winter (24h ave.) velocity over grass
Summer (day) velocity over desert	Summer (night) velocity over desert	Summer (24h ave.) velocity over desert	Winter (day) velocity over desert	Winter (night) velocity over desert	Winter (24h ave.) velocity over desert
Summer (day) velocity over ice	Summer (night) velocity over ice	Summer (24h ave.) velocity over ice	Winter (day) velocity over ice	Winter (night) velocity over ice	Winter (24h ave.) velocity over ice
`SCHEME`	**`QUALIFIER`**	**`DISQUALIFIER`**	**`VN`**		

and are in cm/s. The *desert* category is not used, and only the *day* and *night* values are considered in the calculation of the dry-deposition velocities. Examples of these values are

```
! 7
! R and T are at older revision than S and ST. Make consistent
depvel_t(7,'O3        ',& ! (Ganzeveld& Lelieveld (1995) note 1
                          ! (modified to be the same as Guang's version)
(/0.05,  0.05,  0.05,  0.05,  0.05,  0.05,& !      1.1
  0.85,  0.30,  0.65,  0.65,  0.25,  0.45,& !      1.2
  0.65,  0.25,  0.45,  0.65,  0.25,  0.45,& !      1.3
  0.18,  0.18,  0.18,  0.18,  0.18,  0.18,& !      1.4
  0.05,  0.05,  0.05,  0.05,  0.05,  0.05/),& !      1.5
  TI+ST,0,0,107),&
depvel_t(7,'O3        ',&
(/0.05,  0.05,  0.05,  0.05,  0.05,  0.05,              &
  1.00,  0.11,  0.56,  0.26,  0.11,  0.19,              &
  1.00,  0.37,  0.69,  0.59,  0.46,  0.53,              &
  0.26,  0.26,  0.26,  0.26,  0.26,  0.26,              &
  0.05,  0.05,  0.05,  0.05,  0.05,  0.05/),              &
  T+R,0,0,107),&
depvel_t(7,'O3        ',&
! O3 (Ganzeveld& Lelieveld (1995) - note 1)
(/0.07,  0.07,  0.07,  0.07,  0.07,  0.07,&
  1.00,  0.11,  0.56,  0.26,  0.11,  0.19,&
  1.00,  0.37,  0.69,  0.59,  0.46,  0.53,&
  0.26,  0.26,  0.26,  0.26,  0.26,  0.26,&
  0.07,  0.07,  0.07,  0.07,  0.07,  0.07/),&
  S,0,0,107),&
! 9
! No DD of NO in R scheme
depvel_t(8,'NO        ',& ! (inferred from NO2 - see Giannakopoulos (1998))
(/0.00,  0.00,  0.00,  0.00,  0.00,  0.00,& !      2.1
  0.14,  0.01,  0.07,  0.01,  0.01,  0.01,& !      2.2
  0.10,  0.01,  0.06,  0.01,  0.01,  0.01,& !      2.3
  0.01,  0.01,  0.01,  0.01,  0.01,  0.01,& !      2.4
  0.00,  0.00,  0.00,  0.00,  0.00,  0.00/),& !      2.5
  TI+T+S+ST,0,0,107),&
```

Note: as you can see above, this definition makes use of the `N`, `SCHEME`, `QUALIFIER`, `DISQUALIFIER`, and `VN` format, and the settings for these are the same as in the adding new tracers tutorial, although here **`N`** should be incremented for each new **deposition**, where there might be the same species specified several times with changes to deposition velocities.

This scheme is controlled in **ukca_ddeprt.F90**. The deposition only occurs in the bottom (i.e. 'surface') layer.

Interactive Dry Deposition Scheme

Adding in new species to the interactive scheme is slightly more involved than for the 2D scheme. This scheme is controlled from the **ukca_ddepctl.F90** routine which is called from *ukca_chemistry_ctl.F90*. The two routines **ukca_aerod.F90** and **ukca_surfddr.F90** contain species specific information, and it is these routines that need to be altered to add in values for a new species. Further details on this scheme can be found in the The UKCA UM documentation paper.

When using this scheme, dry deposition occurs throughout the boundary layer, rather than just in the lowest model (i.e. surface) layer.

Changes to ukca_aerod.F90

This routine calculates the aerodynamic and quasi-laminar surface resistances. The species dependant information that is needed is the diffusion coefficient, **d0** (in units of $m^2 s^{-1}$). By default this is set to -1 if the species is not deposited. If it is deposited, and there are no values for this coefficient in the literature, it is suggested that $d_{0,\mathrm{species}}$ is calculated as

$$d_{0,\mathrm{species}} = d_{0,H_2O} \sqrt{M_{H_2O}/M_{\mathrm{species}}}$$

Where M_{H_2O} is the relative molecular mass of H2O, and M_{species} is the relative molecular mass of the species being deposited, and d_{0,H_2O} is the diffusion coefficient for H2O (2.08E-5 $m^2 s^{-1}$).

You should add in an appropriate value for the new species that you are depositing in the **CASE** statement in this routine. Examples of how this is already done are

```
          CASE ('O3        ','NO2       ','O3S       ','NO3       ')
            d0(j) = 1.4e-5
          CASE ('HONO      ')
            d0(j) = d_h2o * SQRT(m_h2o / m_hono)
```

Note: If you have not yet defined a **M_species** value for your new species, you will need to do this in **`ukca_constants.F90`**.

Changes to ukca_surfddr.F90

The Wesely scheme considers 9, 13, 17, or 27 different surface types:

9 Surface Types	13 Surface Types	17 Surface Types	27 Surface Types
5 Plant Functional Types	***9*** Plant Functional Types	***13*** Plant Functional Types	***13*** Plant Functional Types
1. *Broadleaved trees* 2. *Needleleaf trees* 3. *C3 Grass* 4. *C4 Grass* 5. *Shrub* 6. Urban 7. Water 8. Bare Soil 9. Ice	1. *Broadleaved deciduous trees* 2. *Broadleaved evergreen tropical trees* 3. *Broadleaved evergreen temperate trees* 4. *Needleleaf deciduous trees* 5. *Needleleaf evergreen trees* 6. *C3 Grass* 7. *C4 Grass* 8. *Shrub deciduous* 9. *Shrub evergreen* 10. Urban 11. Water 12. Bare Soil 13. Ice	1. *Broadleaved deciduous trees* 2. *Broadleaved evergreen tropical trees* 3. *Broadleaved evergreen temperate trees* 4. *Needleleaf deciduous trees* 5. *Needleleaf evergreen trees* 6. *C3 Grass* 7. *C3 Crop* 8. *C3 Pasture* 9. *C4 Grass* 10. *C4 Crop* 11. *C4 Pasture* 12. *Shrub deciduous* 13. *Shrub evergreen* 14. Urban 15. Water 16. Bare Soil 17. Ice	1. *Broadleaved deciduous trees* 2. *Broadleaved evergreen tropical trees* 3. *Broadleaved evergreen temperate trees* 4. *Needleleaf deciduous trees* 5. *Needleleaf evergreen trees* 6. *C3 Grass* 7. *C3 Crop* 8. *C3 Pasture* 9. *C4 Grass* 10. *C4 Crop* 11. *C4 Pasture* 12. *Shrub deciduous* 13. *Shrub evergreen* 14. Urban 15. Water 16. Bare Soil 17. Ice 18. Elevated ice level 1 19. Elevated ice level 2 20. Elevated ice level 3 21. Elevated ice level 4 22. Elevated ice level 5 23. Elevated ice level 6 24. Elevated ice level 7 25. Elevated ice level 8 26. Elevated ice level 9 27. Elevated ice level 10

The examples below are given for 9 surface types, but you will need to make changes for both options.

ukca_surfddr.F90 sets the surface resistance (in sm^{-1}) for each of the species dry-deposited (**rsurf**)). If a species is not deposited onto a particular type of surface (but is deposited onto other types) then its resistance on this type can be set to a very large value (**r_null**). Often many species are assigned the same values. You will need to add in appropriate values for your species into the **CASE** statement within this routine.

Examples of how this is already done are

```
          CASE ('NO2       ','NO3       ')
      rsurf(:,n)=(/225.,225.,400.,400.,600.,1200.,2600.,1200.,       &
        3500. /)
          CASE ('CO        ')
      rsurf(:,n)=(/3700.,7300.,4550.,1960.,4550.0,r_null,r_null,     &
        4550.0,r_null /)  ! Shrub+bare soil set to C3 grass (guess)
```

Remember to make the changes for ***ALL*** the case statements associated with the different numbers of surface types.

Solution to Task 7.1: adding new dry deposition values

You were given the task

- *You should now add in the dry deposition of **ALICE**. This species deposits in a similar way to **CO**. The values for **depvel_defs_strattrop** are:*

Surface Type	Summer (day)	Summer (night)	Summer (24h ave)	Winter (day)	Winter (night)	Winter (24h ave)
Water	0.00	0.00	0.00	0.00	0.00	0.00
Forest	0.03	0.03	0.03	0.03	0.03	0.03
Grass	0.03	0.03	0.03	0.03	0.03	0.03
Desert	0.03	0.03	0.03	0.03	0.03	0.03
Ice	0.00	0.00	0.00	0.00	0.00	0.00

i.e. the same as for CO.

You were given the hint:

- *You will need to make changes for both dry deposition schemes. The changes to **ukca_surfddr.F90** can be made very easily by adding **ALICE** to the **CO** block in the `CASE` statement.*

For a working Rose suite that has completed this task, please see

- **ARCHER:** `u-as292@60203`
- **vm:** `u-as297@60160`

The specific Rose changes made are:

- **ARCHER:** https://code.metoffice.gov.uk/trac/roses-u/changeset/60203/a/s/2/9/2/trunk
- **vm:** https://code.metoffice.gov.uk/trac/roses-u/changeset/60160/a/s/2/9/7/trunk

The specific Rose changes made are:

ARCHER:

```
Index: app/fcm_make/rose-app.conf
===================================================================
--- app/fcm_make/rose-app.conf  (revision 60067)
+++ app/fcm_make/rose-app.conf  (revision 60203)
@@ -42,4 +42,4 @@
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46619
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46683
```

These differences can be found in the file **`/home/ukca/Tutorial/vn10.9/worked_solutions/Task7.1/Task7.1_rose.patch`** on PUMA.

vm:

```
Index: app/fcm_make/rose-app.conf
===================================================================
--- app/fcm_make/rose-app.conf  (revision 59986)
+++ app/fcm_make/rose-app.conf  (revision 60160)
@@ -42,4 +42,4 @@
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46619
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46683
```

The specific UM changes made are:

```
Index: src/atmosphere/UKCA/ukca_chem_master.F90
===================================================================
--- src/atmosphere/UKCA/ukca_chem_master.F90    (revision 46619)
+++ src/atmosphere/UKCA/ukca_chem_master.F90    (revision 46683)
@@ -81,7 +81,7 @@
 ! define size of master chemistry
 INTEGER, PARAMETER :: n_chch_master = 150 ! number of known species
 INTEGER, PARAMETER :: n_het_master  =  10 ! number of heterogeneous reactions
-INTEGER, PARAMETER :: n_dry_master  =  57 ! number of dry deposition reactions
+INTEGER, PARAMETER :: n_dry_master  =  58 ! number of dry deposition reactions
 INTEGER, PARAMETER :: n_wet_master  =  49 ! number of wet deposition reactions
 INTEGER, PARAMETER :: n_bimol_master= 401 ! number of bimolecular reactions
 INTEGER, PARAMETER :: n_ratj_master =  76 ! number of photolysis reactions
@@ -377,7 +377,7 @@
 chch_t1(113,'MEMALD    ',1,'TR        ','          ',0,0,0,R,0,0,107),&
 chch_t1(114,'GLY       ',1,'TR        ','          ',0,1,0,R,0,0,107),&
 chch_t1(115,'oXYLENE   ',1,'TR        ','          ',0,0,1,R,0,0,107),&
-chch_t1(116,'ALICE     ',1,'TR        ','          ',0,0,1,ST,0,0,107),&
+chch_t1(116,'ALICE     ',1,'TR        ','          ',1,0,1,ST,0,0,107),&
 chch_t1(117,'BOB       ',1,'TR        ','          ',0,0,0,ST,0,0,107)/)
 ! Heterogeneous chemistry
@@ -3329,7 +3329,15 @@
   0.50,  0.50,  0.50,  0.50,  0.50,  0.50,&
   0.50,  0.50,  0.50,  0.50,  0.50,  0.50,&
   0.50,  0.50,  0.50,  0.50,  0.50,  0.50/),&
-  TI,A,0,107) /)
+  TI,A,0,107),&
+! UKCA Tutorial Task7.1
+depvel_t(53,'ALICE     ',&
+(/0.00,  0.00,  0.00,  0.00,  0.00,  0.00,&
+  0.03,  0.03,  0.03,  0.03,  0.03,  0.03,&
+  0.03,  0.03,  0.03,  0.03,  0.03,  0.03,&
+  0.03,  0.03,  0.03,  0.03,  0.03,  0.03,&
+  0.00,  0.00,  0.00,  0.00,  0.00,  0.00/),&
+  ST,0,0,107) /)

 ! determine which chemistry is to be used. Test here that only one scheme is
 ! selected.
Index: src/atmosphere/UKCA/ukca_surfddr.F90
===================================================================
--- src/atmosphere/UKCA/ukca_surfddr.F90        (revision 46619)
+++ src/atmosphere/UKCA/ukca_surfddr.F90        (revision 46683)
@@ -433,7 +433,7 @@
                      r_null,12500.0, 500.0,12500.0 /)
       CASE ('NH3       ')
         rsurf(:,n)=tenpointzero
-      CASE ('CO        ')
+      CASE ('CO        ','ALICE     ')
         rsurf(:,n)=(/ 3700.0, 7300.0, 4550.0, 1960.0, 4550.0,                 &
                       r_null, r_null, 4550.0, r_null /)
         ! Shrub+bare soil set to C3 grass (guess)
@@ -517,7 +517,7 @@
         rsurf(:,n)=rooh
       CASE ('NH3       ')
         rsurf(1:6,n)=(/ 137.0,111.1,111.9,131.3,130.4,209.8 /)
-      CASE ('CO        ')
+      CASE ('CO        ','ALICE     ')
         rsurf(1:6,n)=(/ 3700.0,3700.0,3700.0,7300.0,7300.0,4550.0 /)
         ! Shrub+bare soil set to C3 grass (guess)
       CASE ('CH4       ')
@@ -594,7 +594,7 @@
         rsurf(7:13,n)=(/ 618.6,648.6,784.3,888.9,4000.0,1290.3,4000.0 /)
       CASE ('NH3       ')
         rsurf(7:13,n)=(/ 196.1,185.8,196.1,180.7,148.9,213.5,215.1 /)
-      CASE ('CO        ')
+      CASE ('CO        ','ALICE     ')
         rsurf(7:13,n)=(/ 1960.0,4550.0,4550.0,r_null,r_null,4550.0,r_null /)
           ! Shrub+bare soil set to C3 grass (guess)
       CASE ('HCHO      ')
@@ -648,7 +648,7 @@
       CASE ('NH3       ')
         rsurf(7:17,n)=(/ 209.8,209.8,196.1,196.1,196.1,                       &
                          185.8,196.1,180.7,148.9,213.5,215.1 /)
-      CASE ('CO        ')
+      CASE ('CO        ','ALICE     ')
        rsurf(7:17,n)=(/ 4550.0,4550.0,1960.0,1960.0,1960.0,                  &
                          4550.0,4550.0,r_null,r_null,4550.0,r_null /)
          ! Shrub+bare soil set to C3 grass (guess)
@@ -706,7 +706,7 @@
       CASE ('NH3       ')
         rsurf(18:27,n)=(/ 215.1,215.1,215.1,215.1,215.1,                      &
                           215.1,215.1,215.1,215.1,215.1 /)
-      CASE ('CO        ')
+      CASE ('CO        ','ALICE     ')
         rsurf(18:27,n)=(/ r_null,r_null,r_null,r_null,r_null,                 &
                           r_null,r_null,r_null,r_null,r_null /)
           ! Shrub+bare soil set to C3 grass (guess)
Index: src/atmosphere/UKCA/ukca_aerod.F90
===================================================================
--- src/atmosphere/UKCA/ukca_aerod.F90  (revision 46619)
+++ src/atmosphere/UKCA/ukca_aerod.F90  (revision 46683)
@@ -231,6 +231,8 @@
       d0(j) = d_h2o * SQRT(m_h2o / m_meoh)
     CASE ('Monoterp  ')
       d0(j) = d_h2o * SQRT(m_h2o / m_monoterp)
+    CASE ('ALICE     ')
+      d0(j) = d_h2o * SQRT(m_h2o / m_ALICE)
     END SELECT

```

```
  END DO
  !
```

These differences can be found in the file **/home/ukca/Tutorial/vn10.9/worked_solutions/Task7.1/Task7.1_code.patch** on PUMA.

Sample output from this task can be found at **/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task7.1/atmosa.pa19810901_00** on ARCHER.

Checklist

- □ Put a 1 in the D column of the `chch_defs_master` in `ukca_chem_master.F90`.
- □ Append deposition velocity values to the `depvel_defs_master` array, and increment `n_dry_master`.
- □ Put the correct M_species value in `ukca_constants.F90`.
- □ In `ukca_aerod.F90`, append the CASE statement with your new species to set a value for `d0`.
- □ In `ukca_surfddr.F90`, append all the CASE statements (for the various different numbers of surface types) with the values for your new species.

Tutorial 8

Written by Luke Abraham 2017

Retrieved from "http://www.ukca.ac.uk/wiki/index.php?title=UKCA_Chemistry_and_Aerosol_vn10.9_Tutorial_7&oldid=6460"

- This page was last modified on 11 December 2017, at 09:57.

UKCA Chemistry and Aerosol vn10.9 Tutorial 8

From UKCA

UKCA Chemistry and Aerosol Tutorials at vn10.9

Contents

What you will learn in this Tutorial

In this tutorial you will learn how the wet deposition of chemical species is handelled in UKCA. You will then add-in the wet deposition of one of your new tracers.

Task 8.1: Add wet deposition of a species

Task 8.1: Add in wet deposition for **BOB**, using the following values:

$k(298)$	$-(\Delta H/R)$	$k(298)$ **for the 1st dissociation**	$-(\Delta H/R)$ **for the 1st dissociation**	$k(298)$ **for the 2nd dissociation**	$-(\Delta H/R)$ **for the 2nd dissociation**
$0.21 \times 10^{+06}$	$0.87 \times 10^{+04}$	$0.2 \times 10^{+02}$	0.0	0.0	0.0

Adding Wet Deposition

The formulation used in UKCA is described in Giannakopoulos (1999)[1]. This scheme uses the following formula to calculate the effective Henry's Law coefficient

$$H_{eff} = k\,(298)\exp\left(-\frac{\Delta H}{R}\left[\left(\frac{1}{T}\right)-\left(\frac{1}{298}\right)\right]\right)$$

where $k\,(298)$ is the rate constant at 298K.

During this tutorial you will be tasked with adding the wet deposition of one of your new tracers.

References

1. Giannakopoulos, C., M. P. Chipperfield, K. S. Law, and J. A. Pyle (1999), Validation and intercomparison of wet and dry deposition schemes using 210Pb in a global three-dimensional off-line chemical transport model, J. Geophys. Res., 104(D19), 23761–23784, doi:10.1029/1999JD900392.

Turning on Wet Deposition for a Species

Chemistry Scheme Specification

Within the UKCA code, whether a species is wet deposited or not is controlled in the **ukca_chem_master.F90** module. In the **chch_defs_master** array there are lines like

```
!  10 DD: 7,WD: 4,
chch_t1(10,'HONO2     ',1,'TR        ','          ',1,1,0,TI+S+T+ST+R,0,0,107),&
!  11 DD: 8,WD: 5,
chch_t1(11,'H2O2      ',1,'TR        ','          ',1,1,0,TI+S+T+ST+OL+R,0,0,&
   107),&
```

Where the 1 in the 7th column turns on wet deposition of that species (being 0 otherwise). You will need to change the 0 to a **1** for the species that you wish to now wet deposit.

Setting Henry's Law values

In the **ukca_chem_master.F90** module the parameters required to calculate H_{eff} are held in the **henry_defs_master** array (of defined size **n_wet_master**), and has format

N	**'SPECIES '**				
$k(298)$	$-(\Delta H/R)$	$k(298)$ for the 1st dissociation	$-(\Delta H/R)$ for the 1st dissociation	$k(298)$ for the 2nd dissociation	$-(\Delta H/R)$ for the 2nd dissociation
SCHEME	**QUALIFIER**	**DISQUALIFIER**	**VN**		

Columns 3 and 4 are used if the species dissociates in the aqueous phase. In this case, H_{eff} is further multiplied by a factor of

$$1+\frac{k(aq)}{H^+}$$

where

$$k(aq)=k\left(298\right)\exp\left(-\frac{\Delta H}{R}\left[\left(\frac{1}{T}\right)-\left(\frac{1}{298}\right)\right]\right)$$

and column 3 contains the values of $k(298)$ and column 4 contains the value of $-\Delta H / R$. Similarly, if the species dissociates a second time then a further factor of $1 + k(aq) / H^+$ is applied, where this value of $k(aq)$ is calculated from the values of $k(298)$ and $-\Delta H / R$ in columns 5 and 6.

Note: As with the 2D dry deposition values in **`depvel_defs_master`**, the order of **`henry_defs_master`** also assumes that the values are in the same order as the species (that wet deposit) in the **`chch_defs_master`** array.

Examples for this array are

```
!    4
wetdep(4,'HONO2     ',&
(/0.21e+06,0.87e+04,0.20e+02,0.00e+00,0.00e+00,0.00e+00/),TI+T+ST+R,0,0,107),&
wetdep(4,'HONO2     ',&
(/0.21e+06,0.87e+04,0.157e+02,0.00e+00,0.00e+00,0.00e+00/),S,0,0,107),&
!    5
wetdep(5,'H2O2      ',&
(/0.83e+05,0.74e+04,0.24e-11,-0.373e+04,0.e+00,0.e+00/),TI+T+ST+OL+R,0,0,107),&
wetdep(5,'H2O2      ',&
(/0.83e+05,0.74e+04,0.22e-11,-0.373e+04,0.00e+00,0.00e+00/),S,0,0,107),&
```

Solution to Task 8.1: Add wet deposition of a species

You were given the task

- *Add in wet deposition for **BOB**, using the following values:*

$k(298)$	$-\left(\Delta H/R\right)$	$k(298)$ **for the 1st dissociation**	$-\left(\Delta H/R\right)$ **for the 1st dissociation**	$k(298)$ **for the 2nd dissociation**	$-\left(\Delta H/R\right)$ **for the 2nd dissociation**
$0.21\times10^{+06}$	$0.87\times10^{+04}$	$0.2\times10^{+02}$	0.0	0.0	0.0

For a working Rose suite that has completed this task, please see

- **ARCHER:** `u-as292@60205`
- **vm:** `u-as297@60206`

The specific Rose changes made are:

- **ARCHER:** https://code.metoffice.gov.uk/trac/roses-u/changeset/60205/a/s/2/9/2/trunk
- **vm:** https://code.metoffice.gov.uk/trac/roses-u/changeset/60206/a/s/2/9/7/trunk

The specific Rose changes made are:

ARCHER:

```
Index: app/fcm_make/rose-app.conf
===================================================================
--- app/fcm_make/rose-app.conf  (revision 60203)
+++ app/fcm_make/rose-app.conf  (revision 60205)
@@ -42,4 +42,4 @@
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46683
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46696
```

These differences can be found in the file **`/home/ukca/Tutorial/vn10.9/worked_solutions/Task8.1/Task8.1_rose.patch`** on PUMA.

vm:

```
 Index: app/fcm_make/rose-app.conf
===================================================================
--- app/fcm_make/rose-app.conf  (revision 60160)
+++ app/fcm_make/rose-app.conf  (revision 60206)
@@ -42,4 +42,4 @@
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46683
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46696
```

The specific UM changes made are:

```

Index: src/atmosphere/UKCA/ukca_chem_master.F90
===================================================================
--- src/atmosphere/UKCA/ukca_chem_master.F90    (revision 46683)
+++ src/atmosphere/UKCA/ukca_chem_master.F90    (revision 46696)
@@ -82,7 +82,7 @@
 INTEGER, PARAMETER :: n_chch_master = 150 ! number of known species
 INTEGER, PARAMETER :: n_het_master  =  10 ! number of heterogeneous reactions
 INTEGER, PARAMETER :: n_dry_master  =  58 ! number of dry deposition reactions
-INTEGER, PARAMETER :: n_wet_master  =  49 ! number of wet deposition reactions
+INTEGER, PARAMETER :: n_wet_master  =  50 ! number of wet deposition reactions
 INTEGER, PARAMETER :: n_bimol_master= 401 ! number of bimolecular reactions
 INTEGER, PARAMETER :: n_ratj_master =  76 ! number of photolysis reactions
 INTEGER, PARAMETER :: n_ratt_master =  49 ! number of termolecular reactions
@@ -378,7 +378,7 @@
 chch_t1(114,'GLY       ',1,'TR        ','          ',0,1,0,R,0,0,107),&
 chch_t1(115,'oXYLENE   ',1,'TR        ','          ',0,0,1,R,0,0,107),&
 chch_t1(116,'ALICE     ',1,'TR        ','          ',1,0,1,ST,0,0,107),&
-chch_t1(117,'BOB       ',1,'TR        ','          ',0,0,0,ST,0,0,107)/)
+chch_t1(117,'BOB       ',1,'TR        ','          ',0,1,0,ST,0,0,107)/)

 ! Heterogeneous chemistry
 ! Columns take the following meanings:
@@ -1039,7 +1039,10 @@
 wetdep(38,'s-BuOOH   ',&
 (/0.34e+03,0.57e+04,0.00e+00,0.00e+00,0.00e+00,0.00e+00/),R,0,0,107), &
 wetdep(39,'GLY       ',&
-(/0.36e+06,0.00e+00,0.00e+00,0.00e+00,0.00e+00,0.00e+00/),R,0,0,107) /)
+(/0.36e+06,0.00e+00,0.00e+00,0.00e+00,0.00e+00,0.00e+00/),R,0,0,107),&
+!    UKCA Tutorial: wet deposition of BOB
+wetdep(40,'BOB       ',&
+(/0.21e+06,0.87e+04,0.20e+02,0.00e+00,0.00e+00,0.00e+00/),ST,0,0,107) /)

 ! Bimolecular reactions are too many to define here in one statement.
 TYPE(ratb_t1) :: ratb_defs_master(1:n_bimol_master)
```

These differences can be found in the file **/home/ukca/Tutorial/vn10.9/worked_solutions/Task8.1/Task8.1_code.patch** on PUMA.

Sample output from this task can be found at **/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task8.1/atmosa.pa19810901_00** on ARCHER.

Checklist

- □ Put a 1 in the W column of the `chch_defs_master` in `ukca_chem_master.F90`.
- □ Append the Henry's Law parameter values in the `depvel_defs_master` array, and increment `n_wet_master`.

Tutorial 9

Written by Luke Abraham 2017

Retrieved from "http://www.ukca.ac.uk/wiki/index.php?title=UKCA_Chemistry_and_Aerosol_vn10.9_Tutorial_8&oldid=6463"

- This page was last modified on 11 December 2017, at 09:58.

UKCA Chemistry and Aerosol vn10.9 Tutorial 9

From UKCA

UKCA Chemistry and Aerosol Tutorials at vn10.9

Contents

What you will learn in this Tutorial

In this tutorial you will learn about the UKCA diagnostics package and the different diagnostics that you can output using it. You will also learn how to add new diagnostics from the new reactions and deposition that you have added.

Task 9.1: Output new diagnostics

TASK 9.1: Output diagnostics of the reaction $\mathrm{ALICE} + \mathrm{OH} \longrightarrow \mathrm{BOB} + \mathrm{SEC\_ORG}$ to STASH code **50134**, the dry deposition of **ALICE** to STASH code **50135**, and the wet deposition of **BOB** to **50136**. They should be outputted as a **3-hour mean** to the **pa/UPA** stream.

Adding New UKCA Diagnostics

If you are using one of the chemistry schemes that uses ASAD (e.g. *CheT/TropIsop*, *CheS/Strat*, *CheST/StratTrop*) then you can make use of the ASAD Reaction Flux Diagnostics module (held in **asad_chem_flux_diags.F90**). These allow you to straight-forwardly output new reaction and deposition fluxes.

To output new diagnostics you will first need to define them in the **asad_flux_dat.F90** module, and then create new STASHmaster file specifications for them.

During this tutorial you will be tasked with outputting the reaction and deposition fluxes that you have added in to your branch.

Flux Definitions in asad_flux_dat.F90

Within the **asad_flux_dat.F90** module the diagnostics are defined in blocks with the format

```
asad_flux_defn('Diagnostic type',STASH code,'Diagnostic specification',Mask,Reaction number,Number of species, &
(/'Species/Reactant 1','Reactant 2'/),                                                                         &
(/'Product 1','Product 2','Product 3','Product 4'/)),                                                          &
```

Which have the following meaning:

Diagnostic Type

This is a **three character string** which defines what type of diagnostic is being requested. This can take the values

- **RXN** to output the flux through a reaction (in moles/gridcell/s)
- **DEP** to output the deposition flux of a species (in moles/gridcell/s)
- **NET** to output the net chemical tendency of a species (in moles/gridcell/s)
- **STE** to output the net dynamical tendency of a species (in moles/gridcell/s)
- **MAS** to output the mass of the atmosphere (in kg/gridcell)
- **PSC** to output polar stratospheric cloud diagnostics (1 when the gridcell contains a PSC, 0 otherwise - monthly mean field will be a fraction in range 0 → 1)
- **TPM** to output the tropospheric mask (1 for troposphere, 0 otherwise - monthly mean field will be a fraction in range 0 → 1)
- **OUT** to output a tracer in mmr. Only really useful if the field is masked to give the tropospheric concentration only (see the discussion of the **Mask** option)

STASH Code

This is a **5 digit integer** defining the STASH code that the diagnostic will be outputted to (e.g. 50001). Currently this must be in **section 50**.

Diagnostic Specification

This is a **one character string** which is needed to further define what diagnostic is required. If it isn't needed then it should just be set to **X** or left blank.

- *RXN*
  - **B** to output the flux through a bimolecular reaction
  - **T** to output the flux through a termolecular reaction
  - **H** to output the flux through a heterogeneous reaction
  - **J** to output the flux through a photolysis reaction
- *DEP*
  - **D** to output the dry deposition flux
  - **W** to output the wet deposition flux
- *PSC*
  - **1** to output the fraction of Type 1 PSCs
  - **2** to output the fraction of Type 2 PSCs

Mask

This is a **logical** which defines whether only the tropospheric values of the diagnostic are outputted (**.TRUE.**) or not (**.FALSE.**). It is calculated every timestep.

For the *STE* diagnostic this is required if you wish to output the diagnosed stratosphere-troposphere exchange of a species. For the *OUT* diagnostic this can be used to output only the tropospheric concentration of a tracer. This is also used in the calculation of the of the *TPM* diagnostic.

Reaction number

This is an **integer**, and should only be used in the special case of there being two (or more) reactions with the exactly the same reactants and products, but with different rate coefficients. In this case the first reaction in the list would be given number **1** and the second **2** etc. If this is not needed then it should be set to **0** (which will be usual for most reactions).

Number of Species

This is an **integer**, and should give the total number of species, so this will be **1** for diagnostics such as *DEP*, *STE*, *NET* etc., which only consider a single species, and the **total number of reactants and products** for diagnostics *RXN* and *RTE*.

Species

This is a **10-character string** giving the exact name of the species that the diagnostic should be considered for (including capitalisation). This is only used for the *DEP*, *NET*, *STE*, and *OUT*. For the *RXN* and *RTE* diagnostics the full list of reactants and products should be given (see below). For the *MAS*, *PSC*, and *TPM* diagnostics this isn't needed and could either be set to **XXX** or left blank. If it is needed the other reactant/product slots should be left blank.

Reactants and Products

These are **10-character strings**, and should be as the reaction is defined in the **`ukca_chem_master.F90`** module.

Summing Diagnostics

If you define more than one diagnostic to be output to the same STASH code, then the diagnostic routines will sum these diagnostics together. This can be useful (e.g., if you wanted to output the sum of all NO+RO2 reactions to one STASH item), but can be problematic if you accidentally output two fields to the same STASH code, as this will give strange results!

Changes to `asad_flux_dat.F90`

After you have defined your new diagnostics at the top of this module, you will need to make sure that they have been added correctly to the **`asad_chemical_fluxes`** array, which is defined in the **`asad_load_default_fluxes`** subroutine held in the **`asad_flux_dat.F90`** module. You will need to increment the size of this array, given by the **`n_chemical_fluxes`** parameter.

Changes to `asad_chem_flux_diags.F90` (and other UKCA routines)

You will need to edit **`asad_chem_flux_diags.F90`** module if you want to output a new type of diagnostic. This can be quite involved, but you can look at existing routines to see how things are done. You will also need to add code into the main UKCA routines the pass the data through, e.g.

```
  ! 3D flux diagnostics
  IF (L_asad_use_chem_diags .AND.                              &
       ((L_asad_use_flux_rxns .OR. L_asad_use_rxn_rates) .OR.   &
       (L_asad_use_wetdep .OR. L_asad_use_drydep)))             &
       CALL asad_chemical_diagnostics(row_length,rows,          &
       model_levels,ix,jy,k,secs_per_step,volume,ierr)
```

STASHmaster file

While the diagnostics are defined in **asad_flux_dat.F90** they are turned on by requesting the item through STASH. To do this you will need to edit the STASHmaster_A file in your branch.

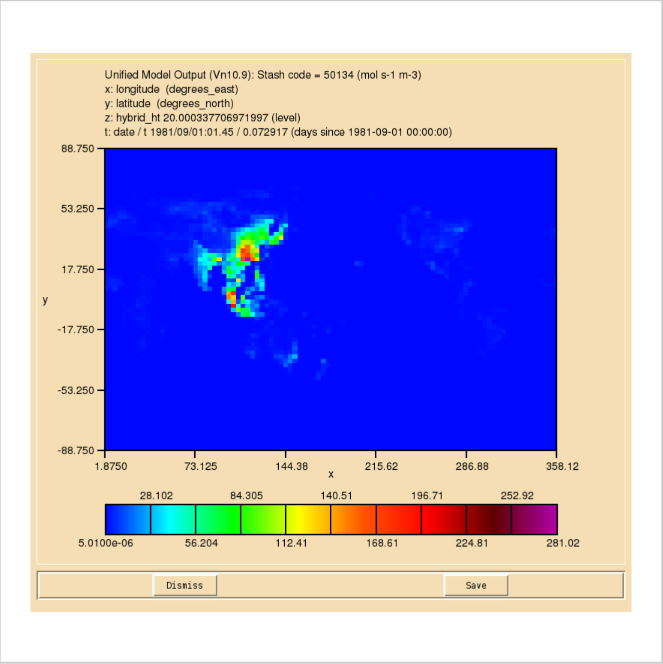
Most UKCA diagnostics are 3D, although some, such as emissions (which are outputted in a different way), are 2D. You should take care with the STASH settings in STASHmaster_A between these types, as there some differences that will need to be considered.

Rose Changes

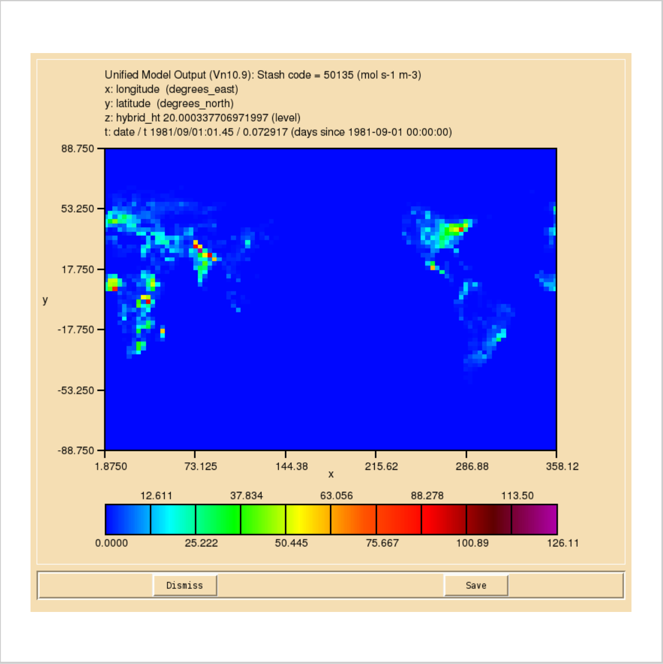
If you have not done so, you will also need to make sure that you use your new STASHmaster_A file in Rose, as is explained in detail in Tutorial 4.

After you have made your STASmaster_A file changes, you will need to add these diagnostics into STASH, as per Tutorial 3. Remember to run the TidyStashTransform macro. As you will have included your branch's STASHmaster_A file using *@HEAD*, you won't need to make any further changes to Rose, but you will need to make sure that you have committed your UM branch prior to running. Because the file is taken from ***`fcm:um.xm_br`*** you will need to make sure that the revision has synced to the PUMA mirror (i.e. by waiting a few minutes).

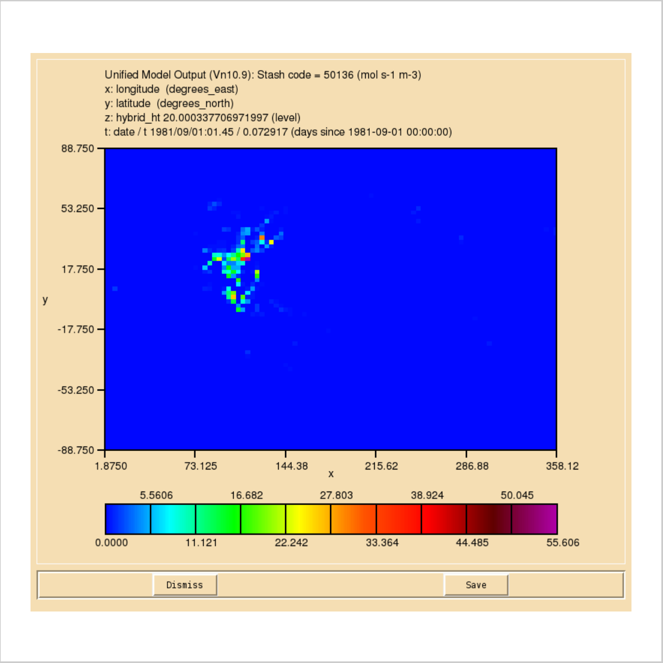
Solution to Task 9.1: Output new diagnostics


Unified Model Output (Vn10.9): Stash code = 50134 (mol s-1 m-3)
x: longitude (degrees_east)
y: latitude (degrees_north)
z: hybrid_ht 20.000337706971997 (level)
t: date / t 1981/09/01:01.45 / 0.072917 (days since 1981-09-01 00:00:00)
88.750
53.250
17.750
y
-17.750
-53.250
-88.750
1.8750
73.125
144.38
x
215.62
286.88
358.12
28.102
84.305
140.51
196.71
252.92
5.0100e-06
56.204
112.41
168.61
224.81
281.02
Dismiss
Save

ALICE + OH → BOB + Sec_Org (s50i230) reaction flux (mol/s) at the surface (20m)


Unified Model Output (Vn10.9): Stash code = 50135 (mol s-1 m-3)
x: longitude (degrees_east)
y: latitude (degrees_north)
z: hybrid_ht 20.000337706971997 (level)
t: date / t 1981/09/01:01.45 / 0.072917 (days since 1981-09-01 00:00:00)
88.750
53.250
17.750
y
-17.750
-53.250
-88.750
1.8750
73.125
144.38
x
215.62
286.88
358.12
12.611
37.834
63.056
88.278
113.50
0.0000
25.222
50.445
75.667
100.89
126.11
Dismiss
Save

ALICE dry deposition flux (s50i231) (mol/s) at the surface (20m)


Unified Model Output (Vn10.9): Stash code = 50136 (mol s-1 m-3)
x: longitude (degrees_east)
y: latitude (degrees_north)
z: hybrid_ht 20.000337706971997 (level)
t: date / t 1981/09/01:01.45 / 0.072917 (days since 1981-09-01 00:00:00)
88.750
53.250
17.750
y
-17.750
-53.250
-88.750
1.8750
73.125
144.38
x
215.62
286.88
358.12
5.5606
16.682
27.803
38.924
50.045
0.0000
11.121
22.242
33.364
44.485
55.606
Dismiss
Save

BOB wet deposition flux (s50i232) (mol/s) at the surface (20m)

You were given the task

- *Output diagnostics of the reaction* $\mathrm{ALICE} + \mathrm{OH} \longrightarrow \mathrm{BOB} + \mathrm{SEC\_ORG}$ *to STASH code* ***50134****, the dry deposition of* ***ALICE*** *to STASH code* ***50135****, and the wet deposition of* ***BOB*** *to* ***50136****. They should be outputted as a* ***3-hour mean*** *to the* ***pa/UPA*** *stream.*

For a working Rose suite that has completed this task, please see

- **ARCHER:** `u-as292@60289`
- **vm:** `u-as297@60286`

The specific Rose changes made are:

- **ARCHER:** https://code.metoffice.gov.uk/trac/roses-u/changeset/60289/a/s/2/9/2/trunk
- **vm:** https://code.metoffice.gov.uk/trac/roses-u/changeset/60286/a/s/2/9/7/trunk

The specific Rose changes made are:

ARCHER:

```
Index: app/um/rose-app.conf
===================================================================
--- app/um/rose-app.conf        (revision 60205)
+++ app/um/rose-app.conf        (revision 60289)
@@ -3425,6 +3425,30 @@
 tim_name='T3HMN'
 use_name='UPA'
 
+[namelist:umstash_streq(50134_de3d001f)]
+dom_name='DALLTH'
+isec=50
+item=134
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
+[namelist:umstash_streq(50135_8528080c)]
+dom_name='DALLTH'
+isec=50
+item=135
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
+[namelist:umstash_streq(50136_fb2dacd3)]
+dom_name='DALLTH'
+isec=50
+item=136
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
 [namelist:umstash_streq(50156_4ce4e1d4)]
 dom_name='DIAG'
 isec=50
Index: app/fcm_make/rose-app.conf
===================================================================
--- app/fcm_make/rose-app.conf  (revision 60205)
+++ app/fcm_make/rose-app.conf  (revision 60289)
@@ -42,4 +42,4 @@
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46696
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46718
```

These differences can be found in the file **/home/ukca/Tutorial/vn10.9/worked_solutions/Task9.1/Task9.1_rose.patch** on PUMA.

vm:

```
Index: app/um/rose-app.conf
===================================================================
--- app/um/rose-app.conf        (revision 60206)
+++ app/um/rose-app.conf        (revision 60286)
@@ -4017,6 +4017,30 @@
 tim_name='T3HMN'
 use_name='UPA'
 
+[namelist:umstash_streq(50134_de3d001f)]
+dom_name='DALLTH'
+isec=50
+item=134
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
+[namelist:umstash_streq(50135_8528080c)]
+dom_name='DALLTH'
+isec=50
+item=135
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
+[namelist:umstash_streq(50136_fb2dacd3)]
+dom_name='DALLTH'
+isec=50
+item=136
```

```
+package=
+tim_name='T3HMN'
+use_name='UPA'
+
 [namelist:umstash_streq(50156_4ce4e1d4)]
 dom_name='DIAG'
 isec=50
Index: app/fcm_make/rose-app.conf
===================================================================
--- app/fcm_make/rose-app.conf  (revision 60206)
+++ app/fcm_make/rose-app.conf  (revision 60286)
@@ -42,4 +42,4 @@
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46696
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46718
```

The specific UM changes made are:

```

Index: src/atmosphere/UKCA/asad_flux_dat.F90
===================================================================
--- src/atmosphere/UKCA/asad_flux_dat.F90       (revision 46696)
+++ src/atmosphere/UKCA/asad_flux_dat.F90       (revision 46718)
@@ -97,7 +97,7 @@
 CHARACTER(LEN=10) :: blank0 = '          '   ! Defines null product

 ! Number of chemical fluxes defined below
-INTEGER, PARAMETER :: n_chemical_fluxes = 310
+INTEGER, PARAMETER :: n_chemical_fluxes = 313

 TYPE(asad_flux_defn), ALLOCATABLE, SAVE :: asad_chemical_fluxes(:)

@@ -1285,6 +1285,19 @@
 (/'          ','          ','          ','          '/))         &
 /)

+TYPE(asad_flux_defn), PARAMETER, PUBLIC ::                       &
+                                    ukca_tutorial_fluxes(3) = (/ &
+asad_flux_defn('RXN',50134,'B',.FALSE.,0,4,                      &
+(/'ALICE     ','OH        '/),                                   &
+(/'BOB       ','Sec_Org   ','          ','          '/)),        &
+asad_flux_defn('DEP',50135,'D',.FALSE.,0,1,                      &
+(/'ALICE     ','          '/),                                   &
+(/'          ','          ','          ','          '/)),        &
+asad_flux_defn('DEP',50136,'W',.FALSE.,0,1,                      &
+(/'BOB       ','          '/),                                   &
+(/'          ','          ','          ','          '/))         &
+/)
+
 TYPE(asad_flux_defn), PUBLIC :: asad_aerosol_chem(16)

 PUBLIC :: asad_load_default_fluxes
@@ -1367,7 +1380,8 @@
    asad_ro2ro2_reacn,          & ! 8  263
    asad_ch4_oxidn,             & ! 6  269
    asad_o1d_prod,              & ! 3  272
-   asad_h2o_budget             & ! 38 307
+   asad_h2o_budget,            & ! 38 307
+   ukca_tutorial_fluxes        & ! 3
 /)

 IF (printstatus > PrStatus_Normal) THEN
Index: rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster-meta.conf
===================================================================
--- rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster-meta.conf (revision 46696)
+++ rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster-meta.conf (revision 46718)
@@ -21102,6 +21102,21 @@
     =number of moles of Ox being produced by this reaction per second in the
     =whole model.

+[stashmaster:code(50134)]
+description=RXN FLUX: ALICE+OH->BOB+Sec_Org
+help=Flux through ALICE+OH->BOB+Sec_Org reaction
+    =moles/s
+
+[stashmaster:code(50135)]
+description=DRY DEP FLUX: ALICE (3D)
+help=Dry Deposition flux of ALICE (3D)
```

```
+    =moles/s
+
+[stashmaster:code(50136)]
+description=WET DEP FLUX: BOB (3D)
+help=Wet Deposition flux of BOB (3D)
+    =moles/s
+
 [stashmaster:code(50140)]
 description=DMS + OH => SO2 + MeOO + HCHO
 help=Chemical reaction flux for DMS + OH => SO2 + MeOO + HCHO
@@ -24136,6 +24151,21 @@
     =number of moles of Ox being produced by this reaction per second in the
     =whole model.

+[stashmaster:code(52134)]
+description=RXN FLUX: ALICE+OH->BOB+Sec_Org PLEV
+help=Flux through ALICE+OH->BOB+Sec_Org reaction on pressure levels
+    =moles/s
+
+[stashmaster:code(52135)]
+description=DRY DEP FLUX: ALICE ON PRESS LEVELS
+help=Dry Deposition flux of ALICE (3D) on pressure levels
+    =moles/s
+
+[stashmaster:code(52136)]
+description=WET DEP FLUX: BOB ON PRESS LEVELS
+help=Wet Deposition flux of BOB (3D) on pressure levels
+    =moles/s
+
 [stashmaster:code(52140)]
 description=DMS+OH=>SO2+MeOO+HCHO ON PRESS LEVS
 help=Chemical reaction flux for DMS + OH => SO2 + MeOO + HCHO
Index: rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster_A
===================================================================
--- rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster_A (revision 46696)
+++ rose-meta/um-atmos/HEAD/etc/stash/STASHmaster/STASHmaster_A (revision 46718)
@@ -23558,6 +23558,24 @@
 4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
 5|    0 | 1871 |    0 |   65 |    0 |    0 |    0 |    0 |    0 |
 #
+1|    1 |   50 |  134 |RXN FLUX: ALICE+OH->BOB+Sec_Org     |
+2|    0 |    0 |   17 |    1 |    2 |   10 |   11 |    0 |    0 |    0 |    0 |
+3| 000000000000000000000000000000 | 00000000000000000001 |    3 |
+4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
+5|    0 | 1871 |    0 |   65 |    0 |    0 |    0 |    0 |    0 |
+#
+1|    1 |   50 |  135 |DRY DEP FLUX: ALICE (3D)            |
+2|    0 |    0 |   17 |    1 |    2 |   10 |   11 |    0 |    0 |    0 |    0 |
+3| 000000000000000000000000000000 | 00000000000000000001 |    3 |
+4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
+5|    0 | 1871 |    0 |   65 |    0 |    0 |    0 |    0 |    0 |
+#
+1|    1 |   50 |  136 |WET DEP FLUX: BOB (3D)              |
+2|    0 |    0 |   17 |    1 |    2 |   10 |   11 |    0 |    0 |    0 |    0 |
+3| 000000000000000000000000000000 | 00000000000000000001 |    3 |
+4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
+5|    0 | 1871 |    0 |   65 |    0 |    0 |    0 |    0 |    0 |
+#
 1|    1 |   50 |  140 |DMS + OH => SO2 + MeOO + HCHO       |
 2|    0 |    0 |   17 |    1 |    2 |   10 |   11 |    0 |    0 |    0 |    0 |
 3| 000000000000000000000000000000 | 00000000000000000001 |    3 |
@@ -26042,6 +26060,24 @@
 4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
 5|    0 | 1871 |    0 |    8 |    0 |    0 |    0 |    0 |    0 |
 #
+1|    1 |   52 |  134 |RXN FLUX: ALICE+OH->BOB+Sec_Org PLEV|
+2|    0 |    0 |   17 |    1 |    3 |    1 |    2 |    0 |    0 |    0 |    1 |
+3| 000000000000000000000000000000 | 00000000000000000001 |    3 |
+4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
+5|    0 | 1871 |    0 |    8 |    0 |    0 |    0 |    0 |    0 |
+#
+1|    1 |   52 |  135 |DRY DEP FLUX: ALICE ON PRESS LEVELS |
+2|    0 |    0 |   17 |    1 |    3 |    1 |    2 |    0 |    0 |    0 |    1 |
+3| 000000000000000000000000000000 | 00000000000000000001 |    3 |
+4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
+5|    0 | 1871 |    0 |    8 |    0 |    0 |    0 |    0 |    0 |
+#
+1|    1 |   52 |  136 |WET DEP FLUX: BOB ON PRESS LEVELS   |
+2|    0 |    0 |   17 |    1 |    3 |    1 |    2 |    0 |    0 |    0 |    1 |
+3| 000000000000000000000000000000 | 00000000000000000001 |    3 |
+4|    1 |     0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
+5|    0 | 1871 |     0 |     8 |     0 |     0 |     0 |     0 |     0 |
+#
 1|    1 |    52 |   140 |DMS+OH=>SO2+MeOO+HCHO ON PRESS LEVS |
 2|    0 |     0 |    17 |     1 |     3 |     1 |     2 |     0 |     0 |     0 |     1 |
 3| 000000000000000000000000000000 | 00000000000000000001 |     3 |
```

These differences can be found in the file **/home/ukca/Tutorial/vn10.9/worked_solutions/Task9.1/Task9.1_code.patch** on PUMA.

If you open the **.pa** file in Xconv, you should see the following fields:

```
20   : 96   72   38    1     molfluxd: Stash code = 50134
21   : 96   72   38    1     molfluxd: Stash code = 50135
22   : 96   72   38    1     molfluxd: Stash code = 50136
```

Sample output from this task can be found at **/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task9.1/atmosa.pa19810901_00** on ARCHER.

Checklist

☐ Make the required changes to branch's STASHmaster_A file, to add the new diagnostics. Make a note of the STASH items chosen, and also add the pressure-level diagnostics.

☐ Add help text for your diagnostics in your branch's STASHmaster-meta.conf file.

☐ Using a text editor, open the `app/um/rose-app.conf` file from your `roses/[SUITE-ID]` directory, and add the line `STASHMASTER=STASHmaster` in the `[env]` block, then save and close the file.

☐ Using a text editor, open the `rose-suite.conf` file from your `roses/[SUITE-ID]` directory, and add the following lines to the top of the file, before saving and closing it:

```
[file:app/um/file/STASHmaster]
source=fcm:um.xm_br/dev/[your MOSRS userid]/vnX.Y_your_branch_name/rose-meta/um-atmos/HEAD/etc/stash/STASHmaster
```

☐ Edit your suite using `rose edit -M /path/to/your/branch/working/copy`.

☐ Point the metadata in your suite to um-atmos/HEAD.

☐ Include your branch in your suite at: fcm_make → env → Sources.

☐ In `asad_flux_dat.F90`, make a new array of type `asad_flux_defn` and populate it with your new diagnostic specification(s), referencing the same item numbers as in your STASHmaster_A file.

☐ Append this new array at the end of `asad_chemical_fluxes`, and increment `n_chemical_fluxes` by the number of new diagnostics.

☐ If required, add new code for new diagnostics into `asad_chem_flux_diags.F90` and other UKCA routines as necessary.

☐ Output your diagnostics in STASH at: um → namelist → Model Input and Output → STASH Requests and Profiles → STASH Requests.

☐ Run the TidyStashTransform transform macro.

☐ Save your suite.

☐ In the `roses/[SUITE-ID]` directory, run `fcm commit` to commit your changes to the repository.

☐ Run your suite.

Tutorial 10

Written by Luke Abraham 2017

Retrieved from "http://www.ukca.ac.uk/wiki/index.php?title=UKCA_Chemistry_and_Aerosol_vn10.9_Tutorial_9&oldid=6717"

- This page was last modified on 3 January 2018, at 11:52.

UKCA Chemistry and Aerosol vn10.9 Tutorial 10

From UKCA

UKCA Chemistry and Aerosol Tutorials at vn10.9

Contents

What you will learn in this Tutorial

In this tutorial you will learn how to output and process aerosol diagnostics from UKCA.

Python

In this Tutorial you will make extensive use of Python to process the UM output. Instructions as to how to use this are in Tutorial 5.

Example python scripts are provided for each Task, and you should take the time to read through these and understand what they are doing. None of these scripts plot any of the output, so as an extension to these tasks you could try extending the scripts to plot the output from python directly, rather than visualising via Xconv. Further information (and examples) on how Iris plots can be found here:

- http://scitools.org.uk/iris/docs/latest/userguide/plotting_a_cube.html
- http://scitools.org.uk/iris/docs/latest/examples/General/global_map.html

Task 10.1: Output aerosol diagnostics

TASK 10.1: Output the following aerosol and radiation diagnostics to the **UPA** output stream. You should make a new *time profile* (called TRAD) to only output these on **radiation timesteps**. You will also need to make a new *domain profile* (called D3DAR) for s02i530 and s02i540, to **output these on both model levels and pseudo levels**.

Hint [hide]
Remember to output your aerosol diagnostics on the AOT pseudo levels using the DIAGAOT profile.

STASH Section	STASH Item	STASH Name
1	207	INCOMING SW RAD FLUX (TOA): ALL TSS
1	208	OUTGOING SW RAD FLUX (TOA)
2	205	OUTGOING LW RAD FLUX (TOA)
2	285	MINERAL DUST OPTICAL DEPTH IN RADN.
2	300	AITKEN MODE (SOLUBLE) OPTICAL DEPTH
2	301	ACCUM MODE (SOLUBLE) OPTICAL DEPTH
2	302	COARSE MODE (SOLUBLE) OPTICAL DEPTH
2	303	AITKEN MODE (INSOL) OPTICAL DEPTH
2	304	ACCUM MODE (INSOL) OPTICAL DEPTH
2	305	COARSE MODE (INSOL) OPTICAL DEPTH
2	585	MINERAL DUST ABS. OPICAL DEPTH
2	240	AITKEN (SOLUBLE) ABS OPTICAL DEPTH
2	241	ACCUM (SOLUBLE) ABS OPTICAL DEPTH
2	242	COARSE (SOLUBLE) ABS OPTICAL DEPTH
2	243	AITKEN (INSOL) ABS OPTICAL DEPTH
2	244	ACCUM (INSOL) ABS OPTICAL DEPTH
2	245	COARSE (INSOL) ABS OPTICAL DEPTH
2	530	UKCA 3D AEROSOL EXTINCTION
2	540	CLASSIC 3D AEROSOL EXTINCTION

Radiation Timesteps

The number of shortwave radiation timesteps per day is given by **`i_sw_radstep_perday_prog`**, found in um → namelist → UM Science Settings → Section 01 - 02 Radiation → Shortwave. The equivalent for longwave is **`i_lw_radstep_perday_prog`**, found in um → namelist → UM Science Settings → Section 01 - 02 Radiation → Longwave.

In the UKCA training suite, both of these are set to **16**, i.e. a call every 90 minutes. This value is very configuration dependent - sometimes radiation is called every hour (24), sometimes every 3 hours (8) etc. You should check the settings in your suite if you need to consider any of the radiation diagnostics.

Making new STASH profiles

While we have covered outputting and creating new diagnostics in Tutorial 3, Tutorial 4, Tutorial 5, and Tutorial 9, you were only making use of existing profiles. Here you will learn how to make new **time**, **domain**, and **usage** profiles.

Time profiles

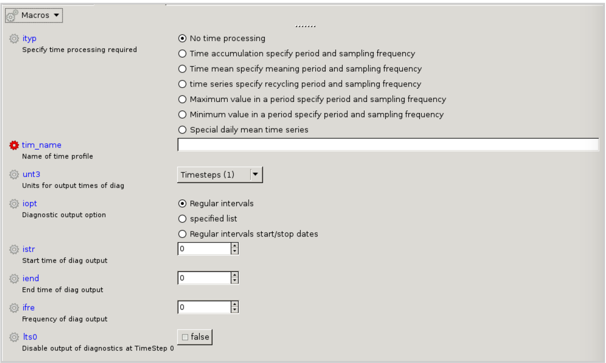
To make a new time profile, go to: **um** → **namelist** → **Model Input and Output** → **STASH Requests and Profiles** → **Time Profiles**. This contains a list of existing time profiles, with names like *`t3hmn_039ecafe`* etc.

To make a new time profile, you should:

1. Right-click anywhere on the list and click the blue plus symbol (**+**) named **new section**

    This will make a new blank line (usually labelled **1**, the next **2** etc.) with a red **X** next to it.

2. Right-click this new line and click **View namelist**.
3. Fill-in the values as required for what you want to do.
4. When you have finished, you will need to run the When you have finished, you will need to run the **stashindices.TidyStashTransform** macro by going to the **Metadata** → **um** drop-down menu. This will rename the new profile into the correct format.



ityp
Specify time processing required
● No time processing
○ Time accumulation specify period and sampling frequency
○ Time mean specify meaning period and sampling frequency
○ time series specify recycling period and sampling frequency
○ Maximum value in a period specify period and sampling frequency
○ Minimum value in a period specify period and sampling frequency
○ Special daily mean time series

tim_name
Name of time profile

unt3
Units for output times of diag
Timesteps (1)

iopt
Diagnostic output option
● Regular intervals
○ specified list
○ Regular intervals start/stop dates

istr
Start time of diag output
0

iend
End time of diag output
0

ifre
Frequency of diag output
0

lts0
Disable output of diagnostics at TimeStep 0
□ false

The new time panel in Rose.

Example: Output on Radiation Timesteps

You will have discovered that there are 16 radiation timesteps per day, but to work out how to output diagnostics you need to know many model timesteps per day there are. This can be found at: um → namelist → Top Level Model Control → Model Domain and Timestep. The number is given by the variable **`steps_per_periodim`**, which is defined as the number of steps per period given (in seconds) by **`secs_per_periodim`**. For one day, this value is 86400.

In the UKCA training suite, `secs_per_periodim`=86400, and `steps_per_periodim`=48. This means that there are 48 timesteps per day, i.e. each timestep is **30 minutes**.

We can combine this with the number of radiation timesteps per day (16), to calculate that **radiation is called every third model timestep** (48/16 = 3) in this configuration. We can now fill-in the values in the new time profile accordingly, e.g.

- **`ityp:`** No time processing
- **`tim_name:`** e.g. TRAD
- **`unt3:`** Timesteps (1)
- **`iopt:`** Regular intervals
- **`istr:`** 1 (i.e. start on first timestep)
- **`iend:`** -1 (i.e. never stop outputting)
- **`ifre:`** 3 (i.e. output every third timestep)

We need to start at timestep 1, as radiation is called on the first timestep, and then called every third model timestep after that. In contrast, the UKCA Newton-Raphson chemical solver is first called on the first hour, and then every hour after that. For the UKCA training suite this would mean that `istr`=2 and `ifre`=2.

Domain profiles

To make a new time profile, go to: **um** → **namelist** → **Model Input and Output** → **STASH Requests and Profiles** → **Domain Profiles**. This contains a list of existing domain profiles, with names like *dallth_1e2e730d* etc.

To make a new domain profile, you should:

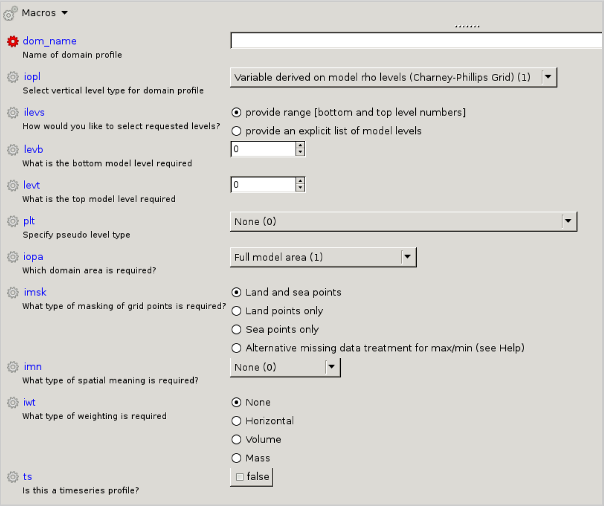
1. Right-click anywhere on the list and click the blue plus symbol (**+**) named **new section**

   This will make a new blank line (usually labelled **1**, the next **2** etc.) with a red **X** next to it.

2. Right-click this new line and click **View namelist**.
3. Fill-in the values as required for what you want to do.
4. When you have finished, you will need to run the When you have finished, you will need to run the **stashindices.TidyStashTransform** macro by going to the **Metadata** → **um** drop-down menu. This will rename the new profile into the correct format.

Example: Output AOT pseudo levels on all model theta levels

Aerosol diagnostics are defined on a series of 6 **pseudo levels**, corresponding to 0.38, 0.44, 0.55, 0.67, 0.87, and 1.02 microns. You will therefore need to output on your data on all of these. For some diagnostics, such as AOD, these are defined on a single level, but others, such as extinction, are a 3D field over the whole atmosphere. Therefore, to output the 3D aerosol extinction we need to make a new profile over all model (theta grid) levels, *and* all 6 pseudo levels.



dom_name
Name of domain profile

iopl
Select vertical level type for domain profile
Variable derived on model rho levels (Charney-Phillips Grid) (1)

ilevs
How would you like to select requested levels?
● provide range [bottom and top level numbers]
○ provide an explicit list of model levels

levb
What is the bottom model level required
0

levt
What is the top model level required
0

plt
Specify pseudo level type
None (0)

iopa
Which domain area is required?
Full model area (1)

imsk
What type of masking of grid points is required?
● Land and sea points
○ Land points only
○ Sea points only
○ Alternative missing data treatment for max/min (see Help)

imn
What type of spatial meaning is required?
None (0)

iwt
What type of weighting is required
● None
○ Horizontal
○ Volume
○ Mass

ts
Is this a timeseries profile?
□ false

The new domain panel in Rose.

We can now fill-in the values in the new domain profile accordingly, e.g.

- **`dom_name`:** e.g. D3DAR
- **`iopl`:** Variable derived on model theta levels (Charney-Phillips Grid) (2)
- **`ilevs`:** provide range [bottom and top level numbers]
- **`ilevb`:** 1 (i.e. 20m level)
- **`ilevt`:** 85 (i.e. highest level)
- **`plt`:** Radiation bands for calculating aerosol optical depth (4)

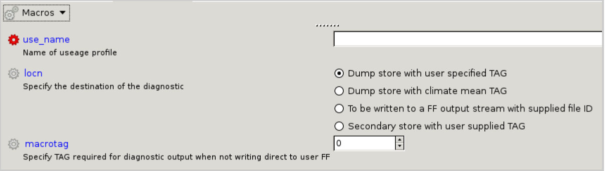
  This will then insert a new option below (`pslist`) to allow you to specify the bands you require.

- **`pslist`:** press the blue plus symbol (**+**) 5 times, and enter the numbers 1 to 6 in each of the boxes to output all AOT pseudo levels.
- **`iopa`:** Full model area (1)
- **`imsk`:** Land and sea points
- **`imn`:** None (0)
- **`iwt`:** None
- **`ts`:** false

Usage profiles

Making new usage profiles is not as straight-forward as for time and domain profiles, as these are used to output diagnostics to different output files (and possibly to climate mean files). To see what output streams are available, go to: **um** → **namelist** → **Model Input and Output** → **Model Output Streams**. In the UKCA training suite, only **pp0** is active.

To look at climate meaning, go to: **um** → **namelist** → **Model Input and Output** → **Dumping and Meaning**. This is controlled by the logical **`l_meaning_sequence`**, which in the UKCA training suite is set to **false** (i.e. no climate meaning). If this is set to *true* then various settings will need to be chosen, such as the reference time, the frequency of files, and which files to output.



use_name
Name of useage profile

locn
Specify the destination of the diagnostic
● Dump store with user specified TAG
○ Dump store with climate mean TAG
○ To be written to a FF output stream with supplied file ID
○ Secondary store with user supplied TAG

macrotag
Specify TAG required for diagnostic output when not writing direct to user FF
0

The new usage panel in Rose.

To view the usage profiles, go to: **um** → **namelist** → **Model Input and Output** → **STASH Requests and Profiles** → **Usage Profiles**. This contains a list of existing domain profiles, with names like *upa_ffb3f00b* etc. If you look at the **upa** namelist (by right-clicking on it and clicking **view namelist**) you can see that this corresponds with the *pp0* output stream.

It is also possible to write diagnostics to a **tag**, which can then be read by other parts of the code. This is how UKCA couples to the UM, via tag=98. This can be seen in the **upukca** usage profile.

When you take a copy of an existing configuration it is unlikely that you will need to make any changes to any (or make new) usage profiles. However, if you do want to make a new usage profile, you should:

1. Right-click anywhere on the list and click the blue plus symbol (**+**) named **new section**

   This will make a new blank line (usually labelled **1**, the next **2** etc.) with a red **X** next to it.

2. Right-click this new line and click **View namelist**.
3. Fill-in the values as required for what you want to do.
4. When you have finished, you will need to run the When you have finished, you will need to run the **stashindices.TidyStashTransform** macro by going to the **Metadata** → **um** drop-down menu. This will rename the new profile into the correct format.

Note that if you are making a new usage profile that is not associated with a *tag*, you may also need to make changes to the output streams. This could be quite involved.

Using your new profiles

Once you have made your new profiles you should then use them in the usual way when outputting diagnostics, as covered in Tutorial 3.

When all your new STASH is set-up, you can use the Validator Macros to check if the names are correct, or if there are any unused STASH items etc. To run this, go to **Metadata** ⟶ **Check all Validator Macros** from the drop-down menu. You can also check for *fail-if* or *warn-if* issues. However, these won't warn of incorrect domain requests (i.e. requesting pressure-levels for diagnostics only defined on a single level), so some care should still be taken.

Solution to Task 10.1

You were given the task

- *Output the following aerosol and radiation diagnostics to the **UPA** output stream. You should make a new* time profile *(called TRAD) to only output these on **radiation timesteps**. You will also need to make a new* domain profile *(called D3DAR) for s02i530 and s02i540, to **output these on both model levels and pseudo levels**.*

STASH Section	STASH Item	STASH Name
1	207	INCOMING SW RAD FLUX (TOA): ALL TSS
1	208	OUTGOING SW RAD FLUX (TOA)
2	205	OUTGOING LW RAD FLUX (TOA)
2	285	MINERAL DUST OPTICAL DEPTH IN RADN.
2	300	AITKEN MODE (SOLUBLE) OPTICAL DEPTH
2	301	ACCUM MODE (SOLUBLE) OPTICAL DEPTH
2	302	COARSE MODE (SOLUBLE) OPTICAL DEPTH
2	303	AITKEN MODE (INSOL) OPTICAL DEPTH
2	304	ACCUM MODE (INSOL) OPTICAL DEPTH
2	305	COARSE MODE (INSOL) OPTICAL DEPTH
2	585	MINERAL DUST ABS. OPICAL DEPTH
2	240	AITKEN (SOLUBLE) ABS OPTICAL DEPTH
2	241	ACCUM (SOLUBLE) ABS OPTICAL DEPTH
2	242	COARSE (SOLUBLE) ABS OPTICAL DEPTH
2	243	AITKEN (INSOL) ABS OPTICAL DEPTH
2	244	ACCUM (INSOL) ABS OPTICAL DEPTH
2	245	COARSE (INSOL) ABS OPTICAL DEPTH
2	530	UKCA 3D AEROSOL EXTINCTION
2	540	CLASSIC 3D AEROSOL EXTINCTION

and were given the hint

- *Remember to output your aerosol diagnostics on the AOT pseudo levels using the DIAGAOT profile.*

For a working Rose suite that has completed this task, please see

- **ARCHER:** `u-as292@62651`
- **vm:** `u-as297@62631`

The specific Rose changes made are:

- **ARCHER:** https://code.metoffice.gov.uk/trac/roses-u/changeset/62651/a/s/2/9/2/trunk
- **vm:** https://code.metoffice.gov.uk/trac/roses-u/changeset/62631/a/s/2/9/7/trunk

The specific Rose changes made are:

ARCHER:

```
Index: app/um/rose-app.conf
===================================================================
--- app/um/rose-app.conf       (revision 60289)
+++ app/um/rose-app.conf       (revision 62651)
@@ -2744,6 +2744,40 @@
 precip_segment_size=32
 ukca_mode_seg_size=4

+[namelist:umstash_domain(d3dar_72578706)]
+dom_name='D3DAR'
+!!iest=0
+ilevs=1
+imn=0
+imsk=1
+!!inth=0
+iopa=1
+iopl=2
+!!isth=0
+!!iwst=0
+iwt=0
+!!l_spml_ts=.false.
+levb=01
+!!levlst=0
+levt=85
+plt=4
+pslist=1,2,3,4,5,6
```

```
+!!rlevlst=0
+!!spml_bot=0
+!!spml_ew=0
+!!spml_ns=0
+!!spml_top=0
+!!tblim=0
+!!tblimr=0
+!!telim=0
+!!tnlim=0
+ts=.false.
+!!tslim=0
+!!tsnum=0
+!!ttlim=0
+!!ttlimr=0
+!!twlim=0
+
 [namelist:umstash_domain(dallrh_0496a967)]
 dom_name='DALLRH'
 !!iest=0
@@ -3025,6 +3059,22 @@
 tim_name='TALLTS'
 use_name='UPUKCA'
 
+[namelist:umstash_streq(01207_ed72c304)]
+dom_name='DIAG'
+isec=1
+item=207
+package=
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(01208_83140cd8)]
+dom_name='DIAG'
+isec=1
+item=208
+package=
+tim_name='TRAD'
+use_name='UPA'
+
 [namelist:umstash_streq(01235_3511dd9f)]
 dom_name='DIAG'
 isec=1
@@ -3033,14 +3083,142 @@
 tim_name='TALLTS'
 use_name='UPUKCA'
 
-[namelist:umstash_streq(02301_0f7c5f4a)]
+[namelist:umstash_streq(02205_357bf644)]
+dom_name='DIAG'
+isec=2
+item=205
+package=
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02240_d97aaab7)]
 dom_name='DIAGAOT'
 isec=2
+item=240
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02241_8cda3169)]
+dom_name='DIAGAOT'
+isec=2
+item=241
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02242_91e371db)]
+dom_name='DIAGAOT'
+isec=2
+item=242
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02243_21bc5f11)]
```

```
+dom_name='DIAGAOT'
+isec=2
+item=243
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02244_32fe0790)]
+dom_name='DIAGAOT'
+isec=2
+item=244
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02245_da00b6ef)]
+dom_name='DIAGAOT'
+isec=2
+item=245
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02285_1d9800f0)]
+dom_name='DIAGAOT'
+isec=2
+item=285
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02300_8b9907b5)]
+dom_name='DIAGAOT'
+isec=2
+item=300
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02301_b8711d23)]
+dom_name='DIAGAOT'
+isec=2
 item=301
-package='UKCA Testing'
-tim_name='T3HMN'
+package=' '
+tim_name='TRAD'
 use_name='UPA'
 
+[namelist:umstash_streq(02302_420d0ec7)]
+dom_name='DIAGAOT'
+isec=2
+item=302
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02303_ad5c3af4)]
+dom_name='DIAGAOT'
+isec=2
+item=303
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02304_8c1869b6)]
+dom_name='DIAGAOT'
+isec=2
+item=304
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02305_9ecd020a)]
+dom_name='DIAGAOT'
+isec=2
+item=305
+package=' '
+tim_name='TRAD'
+use_name='UPA'
```

```
+
+[namelist:umstash_streq(02530_7a218781)]
+dom_name='D3DAR'
+isec=2
+item=530
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02540_56484a28)]
+dom_name='D3DAR'
+isec=2
+item=540
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02585_38287367)]
+dom_name='DIAGAOT'
+isec=2
+item=585
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
 [namelist:umstash_streq(03025_c8768f77)]
 dom_name='DIAG'
 isec=3
@@ -3551,6 +3729,25 @@
 !!unt2=2
 unt3=1
 
+[namelist:umstash_time(trad_4c3a45a7)]
+!!iedt=0
+iend=-1
+ifre=3
+!!intv=0
+!!ioff=0
+iopt=1
+!!isam=0
+!!isdt=0
+!!iser=0
+istr=1
+!!itimes=0
+ityp=1
+!!lts0=.false.
+tim_name='TRAD'
+!!unt1=2
+!!unt2=2
+unt3=1
+
 [namelist:umstash_use(upa_ffb3f00b)]
 file_id='pp0'
 locn=3
```

These differences can be found in the file **/home/ukca/Tutorial/vn10.9/worked_solutions/Task10.1/Task10.1_rose.patch** on PUMA.

vm:

```

Index: app/um/rose-app.conf
===================================================================
--- app/um/rose-app.conf       (revision 60286)
+++ app/um/rose-app.conf       (revision 62631)
@@ -2849,6 +2849,40 @@
 !!ttlimr=0.0
 !!twlim=0
 
+[namelist:umstash_domain(d3dar_72578706)]
+dom_name='D3DAR'
+!!iest=0
+ilevs=1
+imn=0
+imsk=1
+!!inth=0
+iopa=1
+iopl=2
+!!isth=0
+!!iwst=0
+iwt=0
```

```
+!!l_spml_ts=.false.
+levb=01
+!!levlst=0
+levt=85
+plt=4
+pslist=1,2,3,4,5,6
+!!rlevlst=0
+!!spml_bot=0
+!!spml_ew=0
+!!spml_ns=0
+!!spml_top=0
+!!tblim=0
+!!tblimr=0
+!!telim=0
+!!tnlim=0
+ts=.false.
+!!tslim=0
+!!tsnum=0
+!!ttlim=0
+!!ttlimr=0
+!!twlim=0
+
 [namelist:umstash_domain(dallrh_0496a967)]
 dom_name='DALLRH'
 !!iest=0
@@ -3617,6 +3651,22 @@
 tim_name='TALLTS'
 use_name='UPUKCA'
 
+[namelist:umstash_streq(01207_ed72c304)]
+dom_name='DIAG'
+isec=1
+item=207
+package=
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(01208_83140cd8)]
+dom_name='DIAG'
+isec=1
+item=208
+package=
+tim_name='TRAD'
+use_name='UPA'
+
 [namelist:umstash_streq(01235_3511dd9f)]
 dom_name='DIAG'
 isec=1
@@ -3625,14 +3675,142 @@
 tim_name='TALLTS'
 use_name='UPUKCA'
 
-[namelist:umstash_streq(02301_0f7c5f4a)]
+[namelist:umstash_streq(02205_357bf644)]
+dom_name='DIAG'
+isec=2
+item=205
+package=
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02240_d97aaab7)]
 dom_name='DIAGAOT'
 isec=2
+item=240
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02241_8cda3169)]
+dom_name='DIAGAOT'
+isec=2
+item=241
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02242_91e371db)]
+dom_name='DIAGAOT'
+isec=2
```

```
+item=242
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02243_21bc5f11)]
+dom_name='DIAGAOT'
+isec=2
+item=243
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02244_32fe0790)]
+dom_name='DIAGAOT'
+isec=2
+item=244
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02245_da00b6ef)]
+dom_name='DIAGAOT'
+isec=2
+item=245
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02285_1d9800f0)]
+dom_name='DIAGAOT'
+isec=2
+item=285
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02300_8b9907b5)]
+dom_name='DIAGAOT'
+isec=2
+item=300
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02301_b8711d23)]
+dom_name='DIAGAOT'
+isec=2
 item=301
-package='UKCA Testing'
-tim_name='T3HMN'
+package=' '
+tim_name='TRAD'
 use_name='UPA'

+[namelist:umstash_streq(02302_420d0ec7)]
+dom_name='DIAGAOT'
+isec=2
+item=302
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02303_ad5c3af4)]
+dom_name='DIAGAOT'
+isec=2
+item=303
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02304_8c1869b6)]
+dom_name='DIAGAOT'
+isec=2
+item=304
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02305_9ecd020a)]
```

```
+dom_name='DIAGAOT'
+isec=2
+item=305
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02530_7a218781)]
+dom_name='D3DAR'
+isec=2
+item=530
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02540_56484a28)]
+dom_name='D3DAR'
+isec=2
+item=540
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
+[namelist:umstash_streq(02585_38287367)]
+dom_name='DIAGAOT'
+isec=2
+item=585
+package=' '
+tim_name='TRAD'
+use_name='UPA'
+
 [namelist:umstash_streq(03025_c8768f77)]
 dom_name='DIAG'
 isec=3
@@ -4504,6 +4682,25 @@
 unt2=2
 unt3=4
 
+[namelist:umstash_time(trad_4c3a45a7)]
+!!iedt=0
+iend=-1
+ifre=3
+!!intv=0
+!!ioff=0
+iopt=1
+!!isam=0
+!!isdt=0
+!!iser=0
+istr=1
+!!itimes=0
+ityp=1
+!!lts0=.false.
+tim_name='TRAD'
+!!unt1=2
+!!unt2=2
+unt3=1
+
 [namelist:umstash_time(traddm_fa7c24ce)]
 !!iedt=0
 iend=-1
```

If you open the **.pa** file in Xconv, you should see the following additional fields:

```

0    : 96    72    1     2     field200: INCOMING SW RAD FLUX (TOA): ALL TSS
1    : 96    72    1     2     field201: OUTGOING SW RAD FLUX (TOA)
2    : 96    72    1     2     olr: OUTGOING LW RAD FLUX (TOA)
3    : 96    72    6     2     unspecified: Stash code = 2240
4    : 96    72    6     2     unspecified: Stash code = 2241
5    : 96    72    6     2     unspecified: Stash code = 2242
6    : 96    72    6     2     unspecified: Stash code = 2243
7    : 96    72    6     2     unspecified: Stash code = 2244
8    : 96    72    6     2     unspecified: Stash code = 2245
9    : 96    72    6     2     unspecified: MINERAL DUST OPTICAL DEPTH IN RADN.
10   : 96    72    6     2     unspecified: AITKEN MODE (SOLUBLE) OPTICAL DEPTH
11   : 96    72    6     2     unspecified: ACCUM MODE (SOLUBLE) OPTICAL DEPTH
12   : 96    72    6     2     unspecified: COARSE MODE (SOLUBLE) OPTICAL DEPTH
13   : 96    72    6     2     unspecified: AITKEN MODE (INSOL) OPTICAL DEPTH
14   : 96    72    6     2     unspecified: ACCUM MODE (INSOL) OPTICAL DEPTH
15   : 96    72    6     2     unspecified: COARSE MODE (INSOL) OPTICAL DEPTH
```

```
16   : 96    72    38    2     unspecified: Stash code = 2530
17   : 96    72    38    2     unspecified: Stash code = 2530
18   : 96    72    38    2     unspecified: Stash code = 2530
19   : 96    72    38    2     unspecified: Stash code = 2530
20   : 96    72    38    2     unspecified: Stash code = 2530
21   : 96    72    38    2     unspecified: Stash code = 2530
22   : 96    72    38    2     unspecified: Stash code = 2540
23   : 96    72    38    2     unspecified: Stash code = 2540
24   : 96    72    38    2     unspecified: Stash code = 2540
25   : 96    72    38    2     unspecified: Stash code = 2540
26   : 96    72    38    2     unspecified: Stash code = 2540
27   : 96    72    38    2     unspecified: Stash code = 2540
28   : 96    72    6     2     unspecified: Stash code = 2585
```

Sample output from this task can be found at **/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.1/atmosa.pa19810901_00** on ARCHER.

Task 10.2: Calculate aerosol optical depth

TASK 10.2: Calculate the aerosol optical depth at 0.55 microns on the second radiation timestep.

Hint [hide]
The UM calculate optical depth diagnostics on 6 pseudo levels corresponding to 0.38, 0.44, 0.55, 0.67, 0.87, and 1.02 microns, therefore 0.55 microns is pseudo level 3.

Note: You will need to use the CLASSIC mineral dust optical depth diagnostic, as the configuration used in these tutorials does not use modal dust. Likewise, the insoluble accumulation and coarse mode diagnostics will be zero as these modes are not used in the configuration used here.

Python script

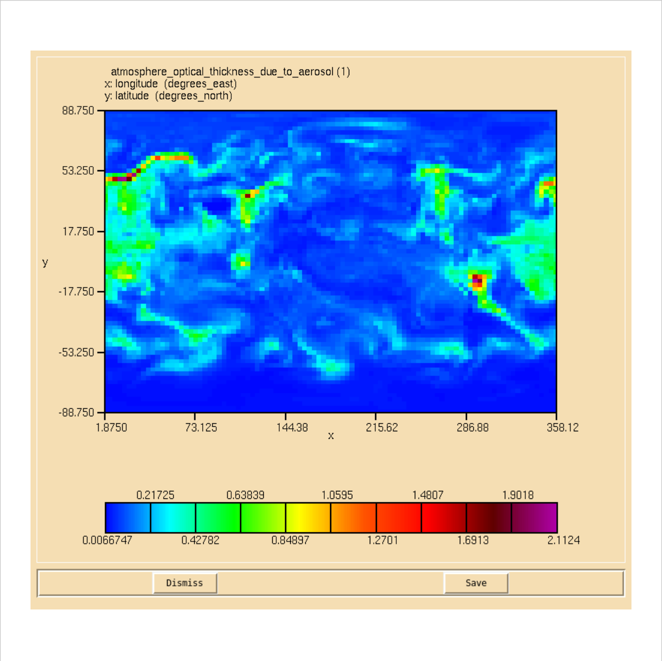
To calculate the total aerosol optical depth at 0.55 microns, you should sum up the contribution from the different aerosol components.

On ARCHER, an example python script to do this has been provided at **/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.2/write_AOD.py**:

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

# This file is part of the UKCA Tutorials:
#  http://www.ukca.ac.uk/wiki/index.php/UKCA_Chemistry_and_Aerosol_Tutorials_at_vn10.9

# Copyright (C) 2017  University of Cambridge

# This is free software: you can redistribute it and/or modify it under the
# terms of the GNU Lesser General Public License as published by the Free
# Software Foundation, either version 3 of the License, or (at your option)
# any later version.

# It is distributed in the hope that it will be useful, but WITHOUT ANY
# WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR
# A PARTICULAR PURPOSE.  See the GNU Lesser General Public License for more
# details.

# You find a copy of the GNU Lesser General Public License at <http://www.gnu.org/licenses/>.

# Written by N. Luke Abraham 2017-12-11 <nla27@cam.ac.uk>

# preamble
import iris
import iris.time

fname='/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.1/atmosa.pa19810901_00'

# constraint on time to get 2nd radiation timestep
tconstr=iris.Constraint(time=lambda cell: cell.point.hour == 2)

# load all AOD components at 0.55 micron
# must use this way of loading to account for constraint on time
with iris.FUTURE.context(cell_datetime_objects=True):
    aod=iris.load(fname,[
        iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i285') & tconstr,
       iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i300') & tconstr,
       iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i301') & tconstr,
       iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i302') & tconstr,
       iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i303') & tconstr,
       iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i304') & tconstr,
       iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i305') & tconstr])

# make cube to store total AOD
aodsum=aod[0].copy()
```

```
# add-up components
aodsum.data=aod[0].data+aod[1].data+aod[2].data+aod[3].data+aod[4].data+aod[5].data+aod[6].data

# rename
aodsum.rename('atmosphere_optical_thickness_due_to_aerosol')

# remove unlimited dimension when writing to netCDF
iris.FUTURE.netcdf_no_unlimited=True

# output to netCDF
iris.save(aodsum,'Task102_AOD.nc',netcdf_format='NETCDF3_CLASSIC')
```

Solution to Task 10.2



0.55 micron AOD calculated from the component aerosol AOD diagnostics.

You were asked to

- *Calculate the aerosol optical depth at 0.55 microns on the second radiation timestep.*

and were given the hint

- *The UM calculate optical depth diagnostics on 6 pseudo levels corresponding to 0.38, 0.44, 0.55, 0.67, 0.87, and 1.02 microns, therefore 0.55 microns is pseudo level 3.*

You should make use of the python script provided to do this.

Sample output from this task can be found in the **/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.2/** directory on ARCHER, containing the following:

```
Task102_AOD.nc
write_AOD.py
```

Task 10.3: Calculate the single-scattering albedo

TASK 10.3: Calculate the single-scattering albedo at 0.55 microns on the second radiation timestep, defined as:

$$1 - \left(\frac{\text{Absorption Aerosol Optical Depth}}{\text{Aerosol Optical Depth}}\right)$$

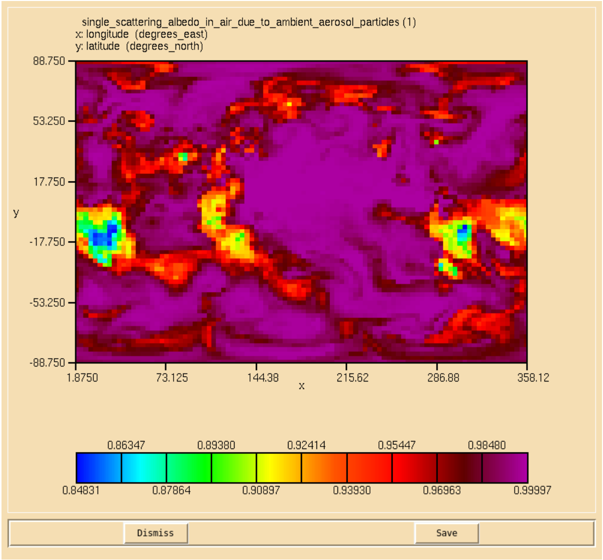
Python script

To calculate the single-scattering albedo at 0.55 microns, you should sum up the contribution to AAOD and AOD from the different aerosol components.

On ARCHER, an example python script to do this has been provided at **/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.3/write_SSA.py**:

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

# This file is part of the UKCA Tutorials:
#  http://www.ukca.ac.uk/wiki/index.php/UKCA_Chemistry_and_Aerosol_Tutorials_at_vn10.9

# Copyright (C) 2017  University of Cambridge

# This is free software: you can redistribute it and/or modify it under the
# terms of the GNU Lesser General Public License as published by the Free
# Software Foundation, either version 3 of the License, or (at your option)
# any later version.

# It is distributed in the hope that it will be useful, but WITHOUT ANY
# WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR
# A PARTICULAR PURPOSE.  See the GNU Lesser General Public License for more
# details.

# You find a copy of the GNU Lesser General Public License at <http://www.gnu.org/licenses/>.

# Written by N. Luke Abraham 2017-12-11 <nla27@cam.ac.uk>

# preamble
import iris
import iris.time

fname='/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.1/atmosa.pa19810901_00'

# constraint on time to get 2nd radiation timestep
tconstr=iris.Constraint(time=lambda cell: cell.point.hour == 2)

# load all AOD & AAOD components at 0.55 micron
# must use this way of loading to account for constraint on time
with iris.FUTURE.context(cell_datetime_objects=True):
    aod=iris.load(fname,[
        iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i285') & tconstr,
       iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i300') & tconstr,
       iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i301') & tconstr,
       iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i302') & tconstr,
       iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i303') & tconstr,
       iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i304') & tconstr,
       iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i305') & tconstr])
    aaod=iris.load(fname,[
         iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i585') & tconstr,
        iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i240') & tconstr,
        iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i241') & tconstr,
        iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i242') & tconstr,
        iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i243') & tconstr,
        iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i244') & tconstr,
        iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i245') & tconstr])

# make cube to store total AOD
aodsum=aod[0].copy()
# add-up components
aodsum.data=aod[0].data+aod[1].data+aod[2].data+aod[3].data+aod[4].data+aod[5].data+aod[6].data

# make cube to store total AAOD
aaodsum=aaod[0].copy()
# add-up components
aaodsum.data=aaod[0].data+aaod[1].data+aaod[2].data+aaod[3].data+aaod[4].data+aaod[5].data+aaod[6].data

# calculate single-scattering albedo
ssa=aodsum.copy()
ssa.data = 1.0 - (aaodsum.data/aodsum.data)

# rename
ssa.rename('single_scattering_albedo_in_air_due_to_ambient_aerosol_particles')

# remove unlimited dimension when writing to netCDF
iris.FUTURE.netcdf_no_unlimited=True

# output to netCDF
iris.save(ssa,'Task103_SSA.nc',netcdf_format='NETCDF3_CLASSIC')
```

Solution to Task 10.3



0.55 micron single-scattering albedo.

You were asked to

- *Calculate the single-scattering albedo at 0.55 microns on the second radiation timestep, defined as:*

$$1-\left(\frac{\text{Absorption Aerosol Optical Depth}}{\text{Aerosol Optical Depth}}\right)$$

You should use the python script provided to do this.

Sample output from this task can be found in the **`/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.3/`** directory on ARCHER, containing the following:

```
Task103_SSA.nc
write_SSA.py
```

Task 10.4: Calculate the top of the atmosphere net downward radiative flux

TASK 10.4: Calculate the net downward top of the atmosphere radiative flux on the second radiation timestep.

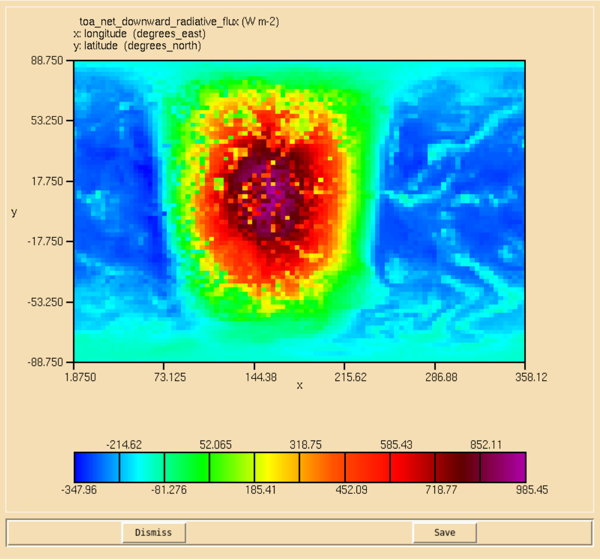
Python script

To calculate the net TOA downward radiative flux, you should sum up the outgoing contributions from shortwave and longwave radiation, and take this away from the incoming shortwave radiative flux.

On ARCHER, an example python script to do this has been provided at **`/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.4/write_TOA.py`**:

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

# This file is part of the UKCA Tutorials:
#  http://www.ukca.ac.uk/wiki/index.php/UKCA_Chemistry_and_Aerosol_Tutorials_at_vn10.9

# Copyright (C) 2017  University of Cambridge

# This is free software: you can redistribute it and/or modify it under the
# terms of the GNU Lesser General Public License as published by the Free
# Software Foundation, either version 3 of the License, or (at your option)
# any later version.

# It is distributed in the hope that it will be useful, but WITHOUT ANY
# WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR
# A PARTICULAR PURPOSE.  See the GNU Lesser General Public License for more
# details.

# You find a copy of the GNU Lesser General Public License at <http://www.gnu.org/licenses/>.

# Written by N. Luke Abraham 2017-12-11 <nla27@cam.ac.uk>

# preamble
import iris
import iris.time

fname='/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.1/atmosa.pa19810901_00'

# constraint on time to get 2nd radiation timestep
tconstr=iris.Constraint(time=lambda cell: cell.point.hour == 2)

# load all TOA components at 0.55 micron
# must use this way of loading to account for constraint on time
with iris.FUTURE.context(cell_datetime_objects=True):
    isw=iris.load_cube(fname,[iris.AttributeConstraint(STASH='m01s01i207') & tconstr])
    osw=iris.load_cube(fname,[iris.AttributeConstraint(STASH='m01s01i208') & tconstr])
    olw=iris.load_cube(fname,[iris.AttributeConstraint(STASH='m01s02i205') & tconstr])

# make cube to store net downward TOA flux
toa=isw.copy()
# add-up components
toa.data=isw.data - (osw.data + olw.data)

toa.rename('toa_net_downward_radiative_flux')

# remove unlimited dimension when writing to netCDF
iris.FUTURE.netcdf_no_unlimited=True

# output to netCDF
iris.save(toa,'Task104_TOA.nc',netcdf_format='NETCDF3_CLASSIC')
```

Solution to Task 10.4



Net downward TOA radiative flux.

You were asked to

- *Calculate the net downward top of the atmosphere radiative flux on the second radiation timestep.*

You should use the python script provided to do this.

Sample output from this task can be found in the **`/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.4/`** directory on ARCHER, containing the following:

```
Task104_TOA.nc
write_TOA.py
```

Task 10.5: Calculate aerosol optical depth from the 3D aerosol extinction

TASK 10.5: Using the 3D aerosol extinction, calculate the 0.55 micron aerosol optical depth on the second radiation timestep, and compare this to your AOD from Task 10.2.

Hint [hide]
You will need to include contributions from both UKCA and CLASSIC (due to the dust scheme).
Remember to correctly calculate the grid-cell heights.

Python script

To calculate the AOD from the aerosol extinction, you will need to integrate this in the column. To do this you should first multiply by the height of each grid-cell before summing-up.

On ARCHER, an example python script to do this has been provided at **/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.5/calc_AOD.py**:

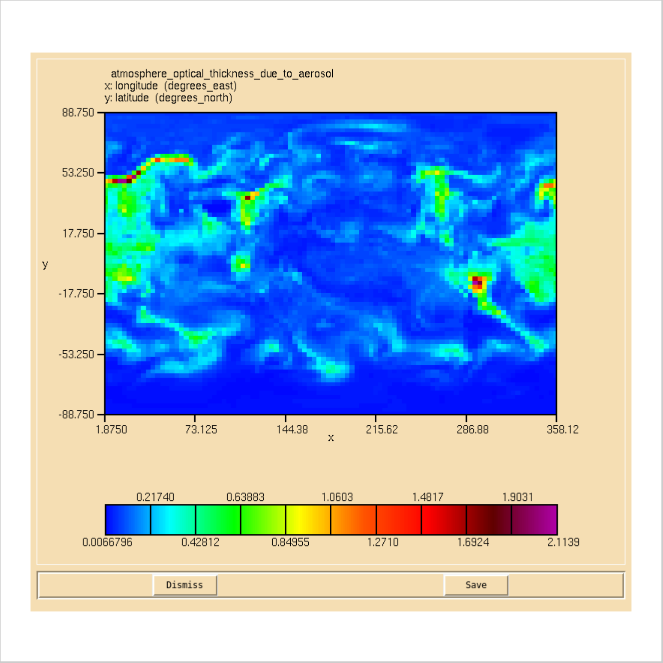
```
#!/usr/bin/env python

# This file is part of the UKCA Tutorials:
#  http://www.ukca.ac.uk/wiki/index.php/UKCA_Chemistry_and_Aerosol_Tutorials_at_vn10.9

# Copyright (C) 2017  University of Cambridge

# This is free software: you can redistribute it and/or modify it under the
# terms of the GNU Lesser General Public License as published by the Free
# Software Foundation, either version 3 of the License, or (at your option)
# any later version.

# It is distributed in the hope that it will be useful, but WITHOUT ANY
# WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR
# A PARTICULAR PURPOSE.  See the GNU Lesser General Public License for more
# details.

# You find a copy of the GNU Lesser General Public License at <http://www.gnu.org/licenses/>.

# Written by N. Luke Abraham 2017-12-11 <nla27@cam.ac.uk>

# preamble
import iris
import iris.time
import iris.analysis
import numpy as np

fname='/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.1/atmosa.pa19810901_00'

# constraint on time to get 2nd radiation timestep
tconstr=iris.Constraint(time=lambda cell: cell.point.hour == 2)

# load orography to enable correct calculation of level heights
orog=iris.load_cube(
     '/work/n02/n02/hum/ancil/atmos/n48e/orography/globe30/v1/qrparm.orog',
     iris.AttributeConstraint(STASH='m01s00i033'))

# load all extinction components at 0.55 micron
# must use this way of loading to account for constraint on time
with iris.FUTURE.context(cell_datetime_objects=True):
    ukca=iris.load_cube(fname,[
         iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i530') & tconstr])
    classic=iris.load_cube(fname,[
            iris.Constraint(pseudo_level=3) & iris.AttributeConstraint(STASH='m01s02i540') & tconstr])

# Calculate the correct height of each cell
# add the orography as an auxillary coordinate
auxcoord=iris.coords.AuxCoord(orog.data,standard_name=str(orog.standard_name),long_name="orography",var_name="or
# added in to lat/lon (ht=0,lat=1,lon=2)
ukca.add_aux_coord(auxcoord,(1,2,))
# now calculate the correct altitude above sea-level
factory=iris.aux_factory.HybridHeightFactory(delta=ukca.coord("level_height"),sigma=ukca.coord("sigma"),orograph
# now create the 'altitude' derrived coordinate
ukca.add_aux_factory(factory)
# now calculate the height from the bounds
bounds = ukca.coord('altitude').bounds[:,:,:,1] - ukca.coord('altitude').bounds[:,:,:,0]

# mutliply by the height of each cell
ukca.data = ukca.data * bounds
classic.data = classic.data * bounds

# now sum up the column
ukca_int=ukca.collapsed('model_level_number',iris.analysis.SUM)
classic_int=classic.collapsed('model_level_number',iris.analysis.SUM)

# add together
aod=ukca_int.copy()
```

```
aod.data = ukca_int.data + classic_int.data
# rename
aod.rename('atmosphere_optical_thickness_due_to_aerosol')

# remove unlimited dimension when writing to netCDF
iris.FUTURE.netcdf_no_unlimited=True

# output to netCDF
iris.save(aod,'Task105_AOD.nc',netcdf_format='NETCDF3_CLASSIC')
```

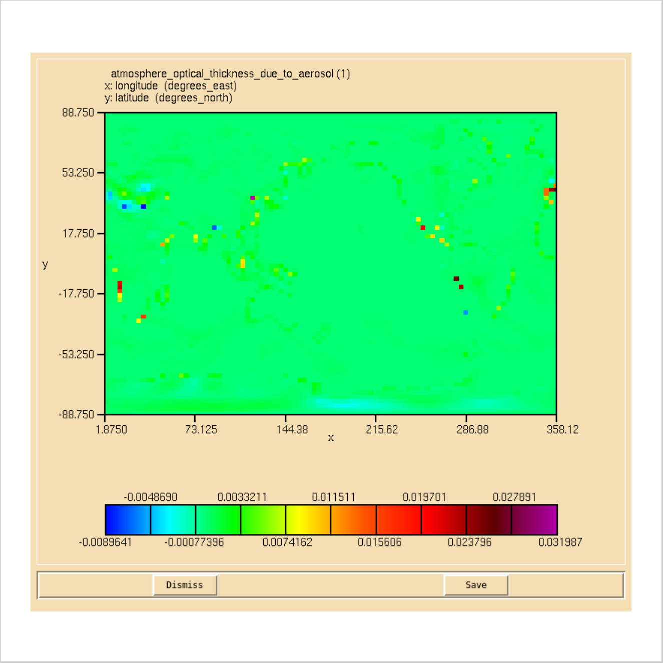
On ARCHER, an example python script to calculate the difference from this AOD and the AOD calculated in Task10.2 is also provided at **/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.5/diff_AOD_methods.py**:

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

# This file is part of the UKCA Tutorials:
#  http://www.ukca.ac.uk/wiki/index.php/UKCA_Chemistry_and_Aerosol_Tutorials_at_vn10.9

# Copyright (C) 2017  University of Cambridge

# This is free software: you can redistribute it and/or modify it under the
# terms of the GNU Lesser General Public License as published by the Free
# Software Foundation, either version 3 of the License, or (at your option)
# any later version.

# It is distributed in the hope that it will be useful, but WITHOUT ANY
# WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR
# A PARTICULAR PURPOSE.  See the GNU Lesser General Public License for more
# details.

# You find a copy of the GNU Lesser General Public License at <http://www.gnu.org/licenses/>.

# Written by N. Luke Abraham 2017-12-11 <nla27@cam.ac.uk>

# preamble
import iris

dname='./Task102_AOD.nc'
cname='./Task105_AOD.nc'

diag=iris.load_cube(dname)
calc=iris.load_cube(cname)

# difference the fields
calc.data=calc.data - diag.data

# remove unlimited dimension when writing to netCDF
iris.FUTURE.netcdf_no_unlimited=True

# output to netCDF
iris.save(calc,'Task105_AOD_diff.nc',netcdf_format='NETCDF3_CLASSIC')
```

Solution to Task 10.5



0.55 micron AOD calculated from the 3D extinction diagnostics.



Difference in AODs calculated by the two methods.

You were asked to

- *Using the 3D aerosol extinction, calculate the 0.55 micron aerosol optical depth on the second radiation timestep, and compare this to your AOD from Task 10.2.*

and were given the hints

- *You will need to include contributions from both UKCA and CLASSIC (due to the dust scheme).*
- *Remember to correctly calculate the grid-cell heights.*

You should use the python script provided to do this.

Note that there will be some small differences between these two fields, especially around coastlines.

Sample output from this task can be found in the **`/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.5/`** directory on ARCHER, containing the following:

```
Task105_AOD.nc
Task105_AOD_diff.nc
calc_AOD.py
diff_AOD_methods.py
```

Task 10.6: Calculate the difference in aerosol impacts when `Sec_Org` is no longer formed from ALICE

TASK 10.6: You should now remove the formation of **`Sec_Org`** from the chemical reaction added in Task 6.1, giving

$$\mathrm{ALICE} + \mathrm{OH} \longrightarrow \mathrm{BOB}$$

and assess the impact this has on the aerosol diagnostics calculated in the previous tasks, 10.2, 10.3, and 10.4.

Hint [hide]
Remember to remove `Sec_Org` from the diagnostics as well as the reaction.

Note: as this a very short run (that is still spinning-up) and you will only be considering a single timestep, the results will be incredibly noisy.

Code changes required to modify a reaction

Modifying a reaction is very similar to adding a new one, although simpler as you are only changing what is already there. The exact routines that will need to be altered will depend on the reaction, but likely include:

`ukca_chem_master.F90`

You must edit the master list of reactions, given by **`ratb_defs_master`** for bimolecular reactions, **`ratt_defs_master`** for termolecular reactions, **`rath_defs_master`** for heterogeneous reactions, or **`ratj_defs_master`** for photolysis reactions.

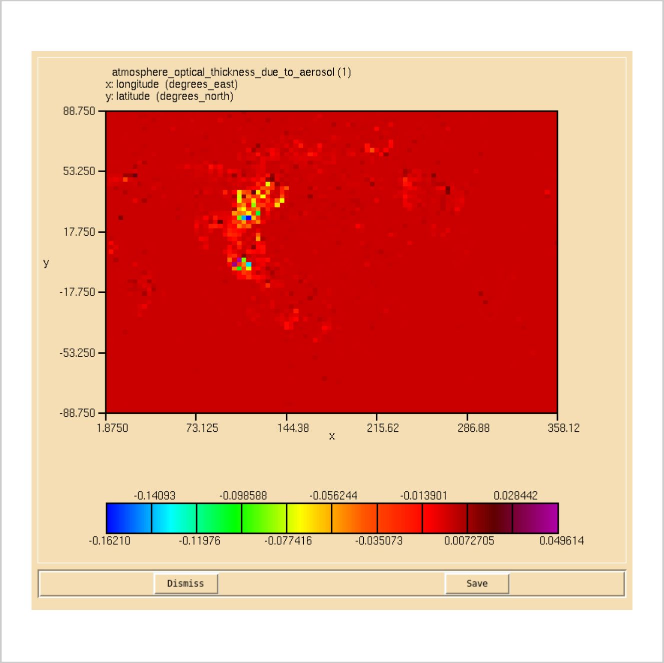
`asad_bimol.F90`, `asad_trimol.F90`, `ukca_hetero_mod.F90`, `asad_hetero.F90`, `ukca_phot2d.F90`, and `ukca_strat_update.F90`

If you added any special code for the reaction(s) to `asad_bimol.F90`, `asad_trimol.F90`, `ukca_hetero_mod.F90`, `asad_hetero.F90`, `ukca_phot2d.F90`, or `ukca_strat_update.F90` it will need to be altered accordingly. The location of the reaction in the complete list is usually found by searching on the reactants and products, and if these change then the lookup will fail unless it is altered here too.

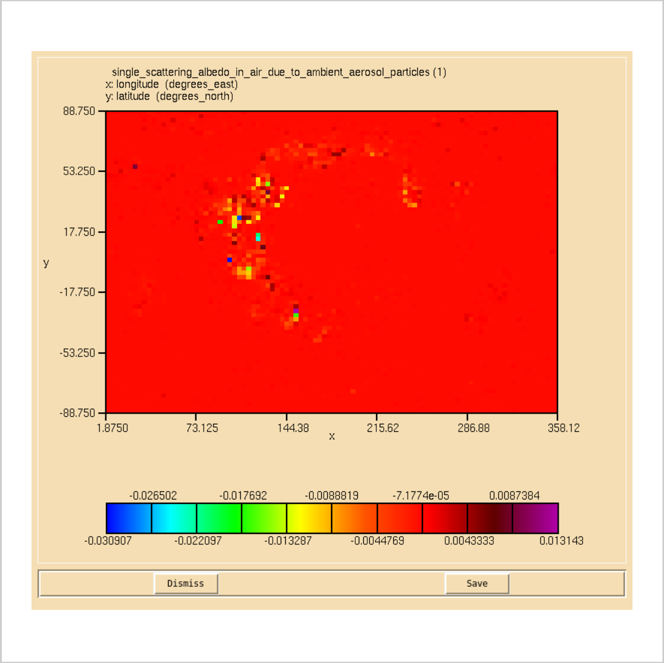
`asad_flux_dat.F90`

If there are any diagnostics to output the reactions you are modifying, you may need to alter this specification as well if you have changed the reactants and/or products.

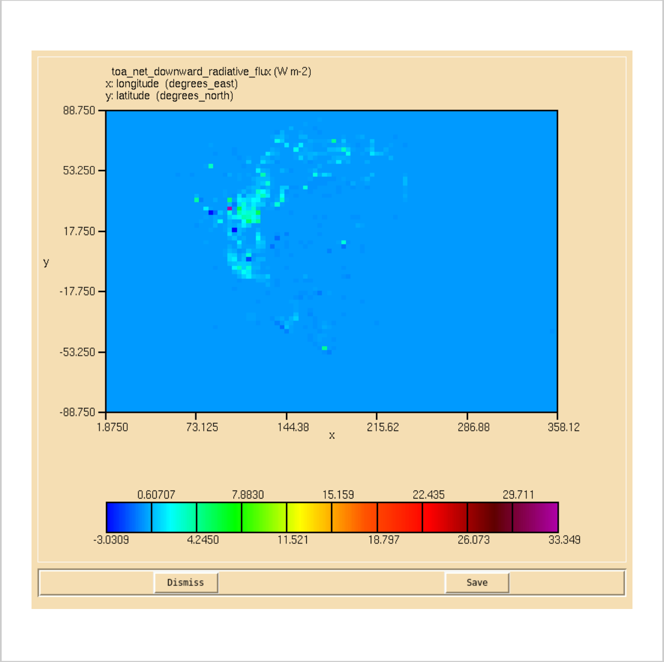
Solution to Task 10.6



Difference in AOD.



Difference in single-scattering albedo.



Difference in net downward TOA fluxes.

You were asked to

- *You should now remove the formation of* ***`Sec_Org`*** *from the chemical reaction added in Task 6.1, giving*

$$\mathrm{ALICE} + \mathrm{OH} \longrightarrow \mathrm{BOB}$$

and assess the impact this has on the aerosol diagnostics calculated in the previous tasks, 10.2, 10.3, and 10.4.

and were given the hint

- *Remember to remove `Sec_Org` from the diagnostics as well as the reaction.*

You should use the above python scripts provided to do this.

For a working Rose suite that has completed this task, please see

- **ARCHER:** `u-as292@62669`

- **vm:** `u-as297@62632`

The specific Rose changes made are:

- **ARCHER:** https://code.metoffice.gov.uk/trac/roses-u/changeset/62669/a/s/2/9/2/trunk
- **vm:** https://code.metoffice.gov.uk/trac/roses-u/changeset/62632/a/s/2/9/7/trunk

The specific Rose changes made are:

ARCHER:

```
Index: app/fcm_make/rose-app.conf
===================================================================
--- app/fcm_make/rose-app.conf (revision 62651)
+++ app/fcm_make/rose-app.conf (revision 62669)
@@ -42,4 +42,4 @@
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46718
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@47380
```

These differences can be found in the file **`/home/ukca/Tutorial/vn10.9/worked_solutions/Task10.6/Task10.6_rose.patch`** on PUMA.

vm:

```
Index: app/fcm_make/rose-app.conf
===================================================================
--- app/fcm_make/rose-app.conf (revision 62631)
+++ app/fcm_make/rose-app.conf (revision 62632)
@@ -42,4 +42,4 @@
 stash_version=1A
 timer_version=3A
 um_rev=vn10.9
-um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@46718
+um_sources=branches/dev/lukeabraham/vn10.9_UKCA_Tutorial_Solns@47380
```

The specific UM changes made are:

```
Index: src/atmosphere/UKCA/asad_flux_dat.F90
===================================================================
--- src/atmosphere/UKCA/asad_flux_dat.F90      (revision 46718)
+++ src/atmosphere/UKCA/asad_flux_dat.F90      (revision 47380)
@@ -1287,9 +1287,9 @@

 TYPE(asad_flux_defn), PARAMETER, PUBLIC ::                       &
                                  ukca_tutorial_fluxes(3) = (/ &
-asad_flux_defn('RXN',50134,'B',.FALSE.,0,4,                       &
+asad_flux_defn('RXN',50134,'B',.FALSE.,0,3,                       &
 (/'ALICE     ','OH        '/),                                    &
-(/'BOB       ','Sec_Org   ','          ','          '/)),        &
+(/'BOB       ','          ','          ','          '/)),        &
 asad_flux_defn('DEP',50135,'D',.FALSE.,0,1,                       &
 (/'ALICE     ','          '/),                                    &
 (/'          ','          ','          ','          '/)),        &
Index: src/atmosphere/UKCA/ukca_chem_master.F90
===================================================================
--- src/atmosphere/UKCA/ukca_chem_master.F90   (revision 46718)
+++ src/atmosphere/UKCA/ukca_chem_master.F90   (revision 47380)
@@ -2156,7 +2156,7 @@
 'HCHO      ',1.00e-12,  0.00,    0.00, 1.00, 0.75, 0.25, 2.75, TI,0,0,107),&
 ratb_t1(277,'MACRO2    ','MeOO      ','HO2       ','CO        ','          ',&
 '          ',1.00e-12,  0.00,    0.00, 1.17, 0.25, 0.00, 0.00, TI,0,0,107),&
-ratb_t1(278,'ALICE     ','OH        ','BOB       ','Sec_Org   ','          ',&
+ratb_t1(278,'ALICE     ','OH        ','BOB       ','          ','          ',&
 '          ',2.70E-11,  0.00, -390.00, 0.00, 0.00, 0.00, 0.00, ST,0,0,107) /)

 !---------------------------------------------------------------------
```

These differences can be found in the file **`/home/ukca/Tutorial/vn10.9/worked_solutions/Task10.6/Task10.6_code.patch`** on PUMA.

Sample output from this task can be found in the **`/work/n02/n02/ukca/Tutorial/vn10.9/sample_output/Task10.6/`** directory on ARCHER, containing the following:

```
Task106_AOD.nc
Task106_AOD_diff.nc
Task106_SSA.nc
```

```
Task106_SSA_diff.nc
Task106_TOA.nc
Task106_TOA_diff.nc
atmosa.pa19810901_00
diff_AOD_rxn.py
diff_SSA_rxn.py
diff_TOA_rxn.py
```

Checklist

- □ Make new time and domain profiles as needed.
- □ Run the TidyStashTransform transform macro.
- □ Output your diagnostics in STASH at: um ⟶ namelist ⟶ Model Input and Output ⟶ STASH Requests and Profiles ⟶ STASH Requests.
- □ When adding new aerosol diagnostics, remember to output on pseudo levels if appropriate.
- □ Run the TidyStashTransform transform macro again for the diagnostic requests.
- □ Save your suite.
- □ In the `roses/[SUITE-ID]` directory, run `fcm commit` to commit your changes to the repository.
- □ When modifying chemical reactions, remember to make changes for diagnostics and special cases, as well as altering `ukca_chem_master.F90`.
- □ The Iris python library can be used to process and visualise Unified Model output.

Tutorial 11

Written by Luke Abraham, Nicolas Bellouin, & Anja Schmidt 2017. With thanks to Ben Johnson.

Retrieved from "http://www.ukca.ac.uk/wiki/index.php?title=UKCA_Chemistry_and_Aerosol_vn10.9_Tutorial_10&oldid=6719"

- This page was last modified on 3 January 2018, at 11:59.

UKCA Chemistry and Aerosol vn10.9 Tutorial 11

From UKCA

UKCA Chemistry and Aerosol Tutorials at vn10.9

Contents

What you will learn in this Tutorial

In this tutorial you will go through a worked example showing you how to commit code to the UM trunk.

You should also open the working practices for the UM (https://code.metoffice.gov.uk/trac/um/wiki/working_practices) page on MOSRS and read through that as well.

This can only be completed on the Virtual Machine, as rose-stem does not currently work on ARCHER.

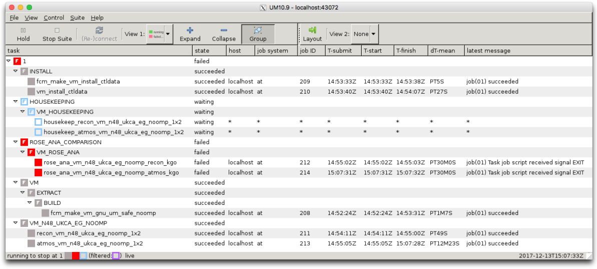
Before you start this Tutorial

Before you start this tutorial you will need to have a working Virtual Machine where **known good output** (or **KGO**) has been installed for the **`vm_n48_ukca_eg_noomp`**. You can do this by running the command

```
rose stem --group=vm_n48_ukca_eg_noomp -S GENERATE_KGO=true
```

within a vanilla copy of the vn10.9 trunk. When this runs for the first time the two KGO tasks will fail, but when this happens the KGO is installed. You just need to re-trigger these two tasks to run again, and they will succeed and the suite will stop.

For more information on the VM you should read-through umdp_X10 (https://code.metoffice.gov.uk/doc/um/latest/papers/umdp_X10.pdf) .



task	state	host	job system	job ID	T-submit	T-start	T-finish	dT-mean	latest message
1	failed								
INSTALL	succeeded								
fcm_make_vm_install_ctldata	succeeded	localhost	at	209	14:53:33Z	14:53:33Z	14:53:38Z	PT5S	job(01) succeeded
vm_install_ctldata	succeeded	localhost	at	210	14:53:40Z	14:53:40Z	14:54:07Z	PT27S	job(01) succeeded
HOUSEKEEPING	waiting								
VM_HOUSEKEEPING	waiting								
housekeep_recon_vm_n48_ukca_eg_noomp_1x2	waiting	*	*	*	*	*	*	*	*
housekeep_atmos_vm_n48_ukca_eg_noomp_1x2	waiting	*	*	*	*	*	*	*	*
ROSE_ANA_COMPARISON	failed								
VM_ROSE_ANA	failed								
rose_ana_vm_n48_ukca_eg_noomp_recon_kgo	failed	localhost	at	212	14:55:02Z	14:55:02Z	14:55:03Z	PT30M0S	job(01) Task job script received signal EXIT
rose_ana_vm_n48_ukca_eg_noomp_atmos_kgo	failed	localhost	at	214	15:07:31Z	15:07:31Z	15:07:32Z	PT30M0S	job(01) Task job script received signal EXIT
VM	succeeded								
EXTRACT	succeeded								
BUILD	succeeded								
fcm_make_vm_gnu_um_safe_noomp	succeeded	localhost	at	208	14:52:24Z	14:52:24Z	14:53:31Z	PT1M7S	job(01) succeeded
VM_N48_UKCA_EG_NOOMP	succeeded								
recon_vm_n48_ukca_eg_noomp_1x2	succeeded	localhost	at	211	14:54:11Z	14:54:11Z	14:55:00Z	PT49S	job(01) succeeded
atmos_vm_n48_ukca_eg_noomp_1x2	succeeded	localhost	at	213	14:55:05Z	14:55:05Z	15:07:28Z	PT12M23S	job(01) succeeded

running to stop at 1 (filtered:) live 2017-12-13T15:07:33Z

Installing KGO on the VM, showing the initial failed state.

Create a ticket

- https://code.metoffice.gov.uk/trac/um/newticket

You will need to fill-in several headings. These can all be changed later if needed.

- **Summary:** Give a short description of what the change is for.
- **Description:** Give a longer and more detailed description.
- **Type:** Putting **enhancement** is usually fine here.
- **Milestone:** This should be the UM (or Mule) version that you are targeting. If you don't know a particular version leave this as **Hopefully**, and if you aren't targeting a version then you can put this as **Not for Builds**. In this example please use *Not for Builds*.
- **Severity:** Usually changes will be **minor** or **significant**, depending on if results are changed. Usually **wholesale** and **trivial** aren't used very often. This doesn't matter too much, as the Code Reviewer will often change this during their review.
- **Keywords:** When working on UKCA, please include **UKCA** and **SC0138** (this is to make it easy to search for UKCA tickets)

Once you have created it, you should **Modify** it and **start work** and then click **submit**. This will change the ownership to you and change the ticket statue to *in_progress*. You will now be able to find your ticket on your view tickets (https://code.metoffice.gov.uk/trac/um/query) page.

You can find an example of this here: um:#3639 (https://code.metoffice.gov.uk/trac/um/ticket/3639) .

Tickets are important as they track the change through the MOSRS system. While initially *"owned"* by you, when you submit your code for review it will be passed to someone else. They will then mark on the ticket when it passes (or fails) their review and pass it on to the next stage prior to commit (or back to you to fix/clarify things). When the code is eventually committed it will be classed as *"closed"* and passed back to you.

It is best to keep the front page of the ticket relatively clear, as it will be used to pass information between yourself and the reviewers as the ticket progresses. You can always make sub-pages linked from the ticket to hold more information if needed.

Make a branch

Please make a branch as covered in Tutorial 4, e.g.

```
fcm branch-create --type dev -k ticket_number your_branch_name fcm:um.x_tr@vn10.9
```

and then check-out your branch by

```
fcm checkout fcm:um.x_br/dev/userid/vn10.9_your_branch_name
```

Merge in Initial Changes

You should merge-in branch `dev/lukeabraham/vn10.9_UKCA_Worked_Example` by changing into the top-level directory of your branch and doing

```
fcm merge fcm:um.x_br/dev/lukeabraham/vn10.9_UKCA_Worked_Example@47427
```

(**Note:** the revision number **47427** is important here) and then **`fcm commit`** your branch.

The code that you are checking in can be found here: https://code.metoffice.gov.uk/trac/um/changeset/47427

```
Index: src/atmosphere/UKCA/ukca_light.F90
===================================================================
--- src/atmosphere/UKCA/ukca_light.F90  (revision 47420)
+++ src/atmosphere/UKCA/ukca_light.F90  (revision 47427)
@@ -241,27 +241,60 @@
   !         3 from cloud base to 2 above top
   !         KLT is the level above cloud top

-  dpcg = ppress(1) - ppress(jniv)
-  dpcc = ppress(jniv) - ppress(klt)
+  IF (l_ukca_linox_logp) THEN
+    !  DO EVERYTHING LINEARLY IN LOG(PRESSURE)
+    ! sanity check to prevent dpcg==0
+    IF (jniv <= 1) jniv=2
+
+    dpcg = LOG(ppress(1)) - LOG(ppress(jniv))
+    dpcc = LOG(ppress(jniv)) - LOG(ppress(klt))
+    !         ...construct L-NOx profile in kg(N)/gridcell/s
+    !         ...first cloud-to-ground L-NOx profiles (kg(N)/gridcell/s)
+    IF ((jniv-1) == 1) THEN
+       anox(1) = acgnox
+    ELSE
+      DO k = 1,jniv-1
+         anox(k) = acgnox * ((LOG(ppress(k))-LOG(ppress(k+1)))/dpcg)
+      END DO
+    END IF
+
+    !         ...then cloud-to-cloud L-NOx profiles (kg(N)/gridcell/s)
+    IF (LOG(ppress(jniv)) <= LOG(ppress(klt))) THEN
+      ! jniv (level of the 500hPa level) is above the
+      ! cloud-top-height. In this case, put all C2C N
+      ! into the cloud-top level.
+      anox(klt-1) = anox(klt-1) + accnox
+    ELSE
+      ! jniv is greater than the cloud-top-height
+      ! Note: anox(k) is also on the RHS of this equation
+      DO k = jniv,klt-1
+         anox(k) = anox(k) + accnox * ( (LOG(ppress(k)) - LOG(ppress(k+1))) / dpcc )
+      END DO
+    END IF
+  ELSE ! .not. l_ukca_linox_logp
+    !  DO EVERYTHING LINEARLY IN PRESSURE
+    dpcg = ppress(1) - ppress(jniv)
+    dpcc = ppress(jniv) - ppress(klt)
+    !         ...construct L-NOx profile in kg(N)/gridcell/s
+    !         ...first cloud-to-ground L-NOx profiles (kg(N)/gridcell/s)
+    IF ((jniv-1) == 1) THEN
+      anox(1) = acgnox
+    ELSE
+      DO k = 1,jniv-1
+         anox(k) = acgnox * ((ppress(k)-ppress(k+1))/dpcg)
+      END DO
+    END IF
+
+    !         ...then cloud-to-cloud L-NOx profiles (kg(N)/gridcell/s)
+    IF (ppress(jniv) <= ppress(klt)) THEN
+      anox(klt-1) = anox(klt-1) + accnox
+    ELSE
+      DO k = jniv,klt-1
+         anox(k) = accnox * ((ppress(k)-ppress(k+1))/dpcc)
+      END DO
+    END IF
+  END IF ! l_ukca_linox_logp

-  !         ...construct L-NOx profile in kg(N)/gridcell/s
-  !         ...first cloud-to-ground L-NOx profiles (kg(N)/gridcell/s)
-  IF ((jniv-1) == 1) THEN
-    anox(1) = acgnox
-  ELSE
-    DO k = 1,jniv-1
-      anox(k) = acgnox * ((ppress(k)-ppress(k+1))/dpcg)
-    END DO
-  END IF
-
-  !         ...then cloud-to-cloud L-NOx profiles (kg(N)/gridcell/s)
-  IF (ppress(jniv) <= ppress(klt)) THEN
-    anox(klt-1) = anox(klt-1) + accnox
-  ELSE
-    DO k = jniv,klt-1
-      anox(k) = accnox * ((ppress(k)-ppress(k+1))/dpcc)
-    END DO
-  END IF
 END IF

 IF (lhook) CALL dr_hook(ModuleName//':'//RoutineName,zhook_out,zhook_handle)
```

The aim of this change is to allow the code to either redistribute lightning NOx emissions vertically either linearly in pressure (the current default) or linearly in LOG(pressure) (the new change).

Take a look through this code. Can you spot any issues that you think it might have?.

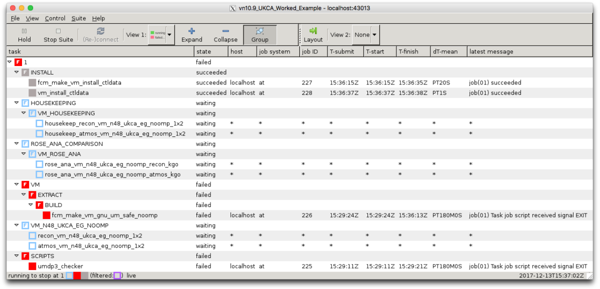
Run rose-stem

You should now run this through rose-stem by running the command

```
rose stem --group=vm_n48_ukca_eg_noomp,umdp3_check
```

The `vm_n48_ukca_eg_noomp` will run the code through one of the UKCA jobs, and the `umdp3_check` will test for coding standard compliance.

Rose-stem will fail. How does it fail? What are the error messages? What can you do to fix them?



task	state	host	job system	job ID	T-submit	T-start	T-finish	dT-mean	latest message
1	failed								
INSTALL	succeeded								
fcm_make_vm_install_ctldata	succeeded	localhost	at	227	15:36:15Z	15:36:15Z	15:36:35Z	PT20S	job(01) succeeded
vm_install_ctldata	succeeded	localhost	at	228	15:36:37Z	15:36:37Z	15:36:38Z	PT1S	job(01) succeeded
HOUSEKEEPING	waiting								
VM_HOUSEKEEPING	waiting								
housekeep_recon_vm_n48_ukca_eg_noomp_1x2	waiting	*	*	*	*	*	*	*	*
housekeep_atmos_vm_n48_ukca_eg_noomp_1x2	waiting	*	*	*	*	*	*	*	*
ROSE_ANA_COMPARISON	waiting								
VM_ROSE_ANA	waiting								
rose_ana_vm_n48_ukca_eg_noomp_recon_kgo	waiting	*	*	*	*	*	*	*	*
rose_ana_vm_n48_ukca_eg_noomp_atmos_kgo	waiting	*	*	*	*	*	*	*	*
VM	failed								
EXTRACT	failed								
BUILD	failed								
fcm_make_vm_gnu_um_safe_noomp	failed	localhost	at	226	15:29:24Z	15:29:24Z	15:36:13Z	PT180M0S	job(01) Task job script received signal EXIT
VM_N48_UKCA_EG_NOOMP	waiting								
recon_vm_n48_ukca_eg_noomp_1x2	waiting	*	*	*	*	*	*	*	*
atmos_vm_n48_ukca_eg_noomp_1x2	waiting	*	*	*	*	*	*	*	*
SCRIPTS	failed								
umdp3_checker	failed	localhost	at	225	15:29:11Z	15:29:11Z	15:29:21Z	PT180M0S	job(01) Task job script received signal EXIT

running to stop at 1 (filtered:) live 2017-12-13T15:37:02Z

Rose-stem failing.

Monsoon2

If you have access to Monsoon2, you should be able to run equivalent test to this by using the following rose-stem command

```
rose stem --group=meto_xc40_n48_ukca_eg_omp_noios_gnu,umdp3_check -S PROJECT=\'your-monsoon-project\' -S HOST_XC
```

This will run the Met Office (`meto`) equivalent to the virtual machine rose-stem job. The `xc40` indicates that it runs on the Cray XC40 architecture, rather than elsewhere (e.g. on a GNU/Linux system).

Note that as all the Met Office rose-stem tests are available, running `--group=developer`, `--group=ukca`, or any one of the larger groups will take a long time to complete due to the reduced size of the system. It is best to limit the number of tests to the minimum for this exercise.

`umdp3_check`

The error message (in the job.out file) is:

```
The following files have failed the UMDP3 compliance tests:
File src/atmosphere/UKCA/ukca_light.F90 :
  Line longer than 80 characters: '         anox(k) = anox(k) + accnox * ( (LOG(ppress(k)) - LOG(ppress(k+1))) /
```

This is relatively easy to solve by adding a continuation line.

```
Index: src/atmosphere/UKCA/ukca_light.F90
===================================================================
--- src/atmosphere/UKCA/ukca_light.F90  (revision 47427)
+++ src/atmosphere/UKCA/ukca_light.F90  (revision 47436)
@@ -268,7 +268,8 @@
       ! jniv is greater than the cloud-top-height
       ! Note: anox(k) is also on the RHS of this equation
       DO k = jniv,klt-1
-         anox(k) = anox(k) + accnox * ( (LOG(ppress(k)) - LOG(ppress(k+1))) / dpcc )
+         anox(k) = anox(k) + accnox *                                 &
+              ( (LOG(ppress(k)) - LOG(ppress(k+1))) / dpcc )
       END DO
     END IF
   ELSE ! .not. l_ukca_linox_logp
```

See https://code.metoffice.gov.uk/trac/um/changeset/47436

If you make this change and then `fcm commit` you can then re-run the `umdp3_check` task again (after stopping any still running suites).

```
rose stem --group=umdp3_check
```

You should find that it passes successfully.

`fcm_make_vm_gnu_um_safe_noomp`

The error is (in the job.err fail) is:

```
[FAIL] /home/vagrant/cylc-run/vn10.9_UKCA_Worked_Example/share/fcm_make_vm_gnu_um_safe_noomp/preprocess-atmos/sr
[FAIL]
[FAIL]    IF (l_ukca_linox_logp) THEN
[FAIL]                         1
[FAIL] Error: Symbol ‘l_ukca_linox_logp’ at (1) has no IMPLICIT type
[FAIL] compile    0.2 ! ukca_light_mod.o     <- um/src/atmosphere/UKCA/ukca_light.F90
[FAIL] ! ukca_light_mod.mod  : depends on failed target: ukca_light_mod.o
[FAIL] ! ukca_light_mod.o    : update task failed
```

This can be easily fixed by adding the following at the top of the `ukca_light.F90` routine:

```
Index: src/atmosphere/UKCA/ukca_light.F90
===================================================================
--- src/atmosphere/UKCA/ukca_light.F90  (revision 47436)
+++ src/atmosphere/UKCA/ukca_light.F90  (revision 47439)
@@ -138,6 +138,9 @@
 REAL :: ns_res_deg  ! gridbox height in degrees
 REAL :: fr_calib_fac ! model resolution calibration factor

+! logical to select redistribution in pressure (F) or LOG(pressure) (T)
+LOGICAL, PARAMETER :: l_ukca_linox_logp = .false.
+
 INTEGER(KIND=jpim), PARAMETER :: zhook_in  = 0
 INTEGER(KIND=jpim), PARAMETER :: zhook_out = 1
 REAL(KIND=jprb)               :: zhook_handle
```

See https://code.metoffice.gov.uk/trac/um/changeset/47439/

Note that the logical is set to **.false.** above, meaning that the code will continue to use the old method.

You can now run rose-stem again:

```
rose stem --group=vm_n48_ukca_eg_noomp
```

Rose-stem will now pass all tests and complete successfully.

What happens if you set `l_ukca_linox_logp = .true.`?

```
Index: src/atmosphere/UKCA/ukca_light.F90
===================================================================
--- src/atmosphere/UKCA/ukca_light.F90  (revision 47439)
+++ src/atmosphere/UKCA/ukca_light.F90  (revision 47446)
@@ -139,7 +139,7 @@
 REAL :: fr_calib_fac ! model resolution calibration factor

 ! logical to select redistribution in pressure (F) or LOG(pressure) (T)
-LOGICAL, PARAMETER :: l_ukca_linox_logp = .false.
+LOGICAL, PARAMETER :: l_ukca_linox_logp = .true.

 INTEGER(KIND=jpim), PARAMETER :: zhook_in  = 0
 INTEGER(KIND=jpim), PARAMETER :: zhook_out = 1
```

See https://code.metoffice.gov.uk/trac/um/changeset/47446

If you now make this change and re-run rose stem

```
rose stem --group=vm_n48_ukca_eg_noomp
```

what happens?

`rose_ana_vm_n48_ukca_eg_noomp_atmos_kgo`

Now, the **rose-ana** task fails with the following error (in the job.out file):

```
[FAIL] %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[FAIL] * CUMF-II Comparison Report *
[FAIL] %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[FAIL]
[FAIL] File 1: /home/vagrant/umdir/standard_jobs/kgo/vm_n48_ukca_eg_1x2/vn10.9/atmosa.pa19810901_00
[FAIL] File 2: /home/vagrant/cylc-run/vn10.9_UKCA_Worked_Example/work/1/atmos_vm_n48_ukca_eg_noomp_1x2/atmosa.pa
[FAIL] Files DO NOT compare
[FAIL]   * 0 differences in fixed_length_header (with 7 ignored indices)
[FAIL]   * 387 field differences, of which 387 are in data
[FAIL]
[FAIL] Compared 583/583 fields, with 196 matches
[FAIL] Maximum RMS diff as % of data in file 1: 129.036987416 (field 496)
[FAIL] Maximum RMS diff as % of data in file 2: 56.3431083041 (field 496)
```

i.e. ***the results have changed***. When finished, rose-stem will generate a file, called **`trac.log`**, that can be found in the `cylc-run` directory of your branch. For the test above, this will show that it has failed one test, but passed the others.

You can see an example of this output here: https://code.metoffice.gov.uk/trac/um/wiki/ticket/3639/stemfail

The `trac.log` file is formatted so that when it is pasted onto the MOSRS Twiki system it will appear as a formatted table. This is because rose-stem output (in the form of the `trac.log` file) is required to complete a ticket for submission to the trunk. More on this will be discussed later.

It is important not to paste the contents of the trac.log file to the front page of the ticket, as the formatting makes this hard to read. If you want to keep a record of multiple rose-stem tests, you should make a new wiki page linked from the ticket and copy the information there.

Hard-coding a logical is not a good idea in this case, as it can only be changed by re-compiling the code. It is easy to forget to do this, and also many UM runs use standard binaries.

Adding new namelist input

A better solution is to introduce the logical `l_ukca_linox_logp` into the input namelist and allow it to be set by a user in Rose. To do this we must make several code changes.

`ukca_option_mod.F90`

All variables within the *`run_ukca`* namelist, that holds all the UKCA inputs, are declared in **`ukca_option_mod.F90`**. This module also contains the namelist itself, as well as some `umPrint` statements and handling for parallel calls. To add a new namelist variable you need to:

1. Add a declaration for it, e.g.

   ```
   LOGICAL :: l_ukca_linox_logp =.FALSE.
   ```

   All logicals must be set to `.FALSE.`, character variables must be set to `'`*`VARIABLE NAME`*` is unset'` and integers and reals must be set to missing data values (defined as `imdi` and `rmdi` respectively).

2. Add the variable to the `run_ukca` namelist.
3. Print the value, e.g.

   ```
   WRITE(lineBuffer,'(A33,L1)')' l_ukca_linox_logp = ', l_ukca_linox_logp
   CALL umPrint(lineBuffer,src='ukca_option_mod')
   ```

4. Add the variable to the `my_namelist` derived type.
5. On processor 0, save the value of the new variable within the `my_nml` variable (which is of type `my_namelist`).
6. After `my_nml` has been broadcast, on all other processors copy the value of the variable out of the `my_nml` variable and into the variable itself. This then means that it can be used as expected in all routines.
7. If required, you may also need to add some checks on the variable to the `check_run_ukca` routine within this module, to ensure that the value (or value range) is correct.

`ukca_light.F90`

In `ukca_light.F90` we now need to stop declaring the variable at the top of the routine, and instead `USE` it from `ukca_option_mod.F90`, e.g.

```
USE ukca_option_mod, ONLY: l_ukca_linox_logp
```

rose-meta/um-atmos/HEAD/rose-meta.conf

We need to add a section here on the new variable to tell Rose how to treat it, e.g.

```
[namelist:run_ukca=l_ukca_linox_logp]
compulsory=true
description=When T, Lightning NOx emissions are distributed
           =vertically using LOG(pressure)
help=When T, this logical makes the UKCA Lightning NOx routine
    =redistribute the Lightning NOx emissions in the vertical
    =linearly using LOG(pressure)
sort-key=b16
type=logical
```

There needs to be help text (which will be shown by Rose when selecting *help* when clicking on the cog associated with the variable, as well as short description that is shown under the variable name to tell the user what it's for. The `sort-key` is used to order the variables in the Rose panel. For integer or real variables, it's possible to give an allowed range of values, or use radio buttons etc.

rose-meta/um-atmos/version109_110.py

As a new namelist input has been added (and similarly if one is removed or changed in some way), then an **upgrade macro** will need to be written to all the new variable to be automatically be inserted into Rose. When the UM goes from one version to another these macros are run sequentially to upgrade Rose to be able to use the new UM version, containing all the new (or changed) namelist items.

This macro takes the form of a short python function which inserts default values (formatted as strings), and looks like this:

```
class vn109_t3639(rose.upgrade.MacroUpgrade):

    """Upgrade macro for ticket #3639 by Luke Abraham."""

    BEFORE_TAG = "vn10.9"
    AFTER_TAG = "vn10.9_t3639"

    def upgrade(self, config, meta_config=None):
        """
        Introduce logical to interpolate linearly in LOG(p)
        for the vertical redistribution of Lightning NOx.
        """

        self.add_setting(config,["namelist:run_ukca",
                                 "l_ukca_linox_logp"],".false.")

        return config, self.reports
```

The name of the function is of the form **vnX.Y_tticket_number.**

All changes

```
Index: src/atmosphere/UKCA/ukca_light.F90
===================================================================
--- src/atmosphere/UKCA/ukca_light.F90  (revision 47446)
+++ src/atmosphere/UKCA/ukca_light.F90  (revision 47454)
@@ -57,6 +57,7 @@
 USE parkind1,        ONLY: jprb, jpim
 USE parcons_mod,     ONLY: rad, deg
 USE ukca_constants,  ONLY: avc
+USE ukca_option_mod, ONLY: l_ukca_linox_logp

 IMPLICIT NONE

@@ -138,9 +139,6 @@
 REAL :: ns_res_deg  ! gridbox height in degrees
 REAL :: fr_calib_fac ! model resolution calibration factor

-! logical to select redistribution in pressure (F) or LOG(pressure) (T)
-LOGICAL, PARAMETER :: l_ukca_linox_logp = .true.
-
 INTEGER(KIND=jpim), PARAMETER :: zhook_in  = 0
 INTEGER(KIND=jpim), PARAMETER :: zhook_out = 1
 REAL(KIND=jprb)               :: zhook_handle
Index: src/atmosphere/UKCA/ukca_option_mod.F90
===================================================================
--- src/atmosphere/UKCA/ukca_option_mod.F90     (revision 47446)
+++ src/atmosphere/UKCA/ukca_option_mod.F90     (revision 47454)
```

```
@@ -78,6 +78,9 @@
 ! T to pass columns to ASAD rather than theta_field
 LOGICAL :: l_ukca_asad_columns =.FALSE.

+! T to use LOG(p) to distribute lightning NOx in the vertical
+LOGICAL :: l_ukca_linox_logp =.FALSE.
+
 INTEGER :: chem_timestep = imdi         ! Chemical timestep in seconds for N-R
                                         ! and Offline oxidant schemes
 INTEGER :: dts0 = 300                   ! Default Backward Euler timestep
@@ -358,7 +361,7 @@
          i_ageair_reset_method, max_ageair_reset_level,          &
          max_ageair_reset_height,                                &
          i_ukca_sad_months, i_ukca_sad_start_year,               &
-         l_ukca_limit_nat
+         l_ukca_limit_nat, l_ukca_linox_logp

 ! -----------------------------------------------------------------------------
 ! These are set in ukca_setup_chem_mod after the namelist is read
@@ -756,6 +759,8 @@
       l_ukca_use_background_aerosol
 WRITE(lineBuffer,'(A33,L1)')' l_ukca_asad_columns = ', &
       l_ukca_asad_columns
+WRITE(lineBuffer,'(A33,L1)')' l_ukca_linox_logp = ', &
+      l_ukca_linox_logp
 CALL umPrint(lineBuffer,src='ukca_option_mod')
 WRITE(lineBuffer,'(A33,L1)')' l_ukca_primsu = ',l_ukca_primsu
 CALL umPrint(lineBuffer,src='ukca_option_mod')
@@ -1000,7 +1005,7 @@
 INTEGER, PARAMETER :: no_of_types = 4
 INTEGER, PARAMETER :: n_int = 21 + a_max_ukcavars
 INTEGER, PARAMETER :: n_real = 25
-INTEGER, PARAMETER :: n_log = 51
+INTEGER, PARAMETER :: n_log = 52
 INTEGER, PARAMETER :: n_chars = 10 * filenamelength                   &
                         + filenamelength * (1+ nr_cdf_files)          &
                         + filenamelength * (1+ max_offline_files)     &
@@ -1106,6 +1111,7 @@
   LOGICAL :: l_ukca_so2ems_expvolc
   LOGICAL :: l_ukca_quasinewton
   LOGICAL :: l_ukca_limit_nat
+  LOGICAL :: l_ukca_linox_logp
   CHARACTER (LEN=filenamelength) :: jvspec_dir
   CHARACTER (LEN=filenamelength) :: jvspec_file
   CHARACTER (LEN=filenamelength) :: jvscat_file
@@ -1243,6 +1249,7 @@
   my_nml % l_ukca_so2ems_expvolc = l_ukca_so2ems_expvolc
   my_nml % l_ukca_quasinewton = l_ukca_quasinewton
   my_nml % l_ukca_limit_nat = l_ukca_limit_nat
+  my_nml % l_ukca_linox_logp = l_ukca_linox_logp
   ! end of logicals
   my_nml % jvspec_dir     = jvspec_dir
   my_nml % jvspec_file    = jvspec_file
@@ -1371,6 +1378,7 @@
   l_ukca_so2ems_expvolc  = my_nml % l_ukca_so2ems_expvolc
   l_ukca_quasinewton = my_nml % l_ukca_quasinewton
   l_ukca_limit_nat = my_nml % l_ukca_limit_nat
+  l_ukca_linox_logp = my_nml % l_ukca_linox_logp
   ! end of logicals
   jvspec_dir     = my_nml % jvspec_dir
   jvspec_file    = my_nml % jvspec_file
Index: rose-meta/um-atmos/version109_110.py
===================================================================
--- rose-meta/um-atmos/version109_110.py        (revision 47446)
+++ rose-meta/um-atmos/version109_110.py        (revision 47454)
@@ -17,15 +17,23 @@




-class vn109_tXXXX(rose.upgrade.MacroUpgrade):
+class vn109_t3639(rose.upgrade.MacroUpgrade):

-    """Upgrade macro for ticket #XXXX by <author>."""
+    """Upgrade macro for ticket #3639 by Luke Abraham."""

     BEFORE_TAG = "vn10.9"
-    AFTER_TAG = "vn10.9_tXXXX"
+    AFTER_TAG = "vn10.9_t3639"

     def upgrade(self, config, meta_config=None):
```

```
-        """Upgrade a UM runtime app configuration."""
-        # Input your macro commands here
+        """
+        Introduce logical to interpolate linearly in LOG(p)
+        for the vertical redistribution of Lightning NOx.
+        """
+
+        self.add_setting(config,["namelist:run_ukca",
+                                 "l_ukca_linox_logp"],".false.")
+
         return config, self.reports

+
+
Index: rose-meta/um-atmos/HEAD/rose-meta.conf
===================================================================
--- rose-meta/um-atmos/HEAD/rose-meta.conf      (revision 47446)
+++ rose-meta/um-atmos/HEAD/rose-meta.conf      (revision 47454)
@@ -23534,6 +23534,16 @@
 sort-key=b15
 type=real

+[namelist:run_ukca=l_ukca_linox_logp]
+compulsory=true
+description=When T, Lightning NOx emissions are distributed
+           =vertically using LOG(pressure)
+help=When T, this logical makes the UKCA Lightning NOx routine
+    =redistribute the Lightning NOx emissions in the vertical
+    =linearly using LOG(pressure)
+sort-key=b16
+type=logical
+
 [namelist:run_ukca=max_ageair_reset_height]
 compulsory=true
 description=Maximum height for resetting Age-of-air tracer values
```

See https://code.metoffice.gov.uk/trac/um/changeset/47454/

Make a test branch and test it with rose-stem

```
fcm branch-create -k ticket_number --branch-of-branch -t test your_branch_name fcm:um.x_br/dev/your_MOSRS_userna
```

You should then checkout this branch, cd into its top-level directory and run the command

```
./admin/rose-stem/update_all.py /home/vagrant/path/to/rXXXXX_your_branch_name --um=vn10.9_tticket_number
```

where XXXXX is the last revision number of `vn10.9_your_branch_name` prior to you creating the test branch.

This python script will then go through and *upgrade* all the **apps**, which are different configurations of the UM used for testing, and insert the new namelist input and check the metadata.

A full list of the changes made in this case can be found here: https://code.metoffice.gov.uk/trac/um/changeset/47481

You should `fcm commit` your changes, and then run rose-stem again

```
rose stem --group=vm_n48_ukca_eg_noomp,scripts
```

where the `scripts` group runs the `umdp3_check` as well as some other checks. This will now pass, and you can see in the `trac.log` (https://code.metoffice.gov.uk/trac/um/wiki/ticket/3639/stempass) file that everything has completed successfully.

The Virtual Machine set of standard tests can be found here: https://code.metoffice.gov.uk/trac/um/wiki/VirtualMachine/StandardJobs10.9

The Met Office set of standard tests can be found here: https://code.metoffice.gov.uk/trac/um/wiki/StandardJobs10.9

The minimum group that is needed to be checked is **`developer`** group at the **Met Office**. If you do not have access to Monsoon2 you will need to ask someone who does to run these tests, or ask someone with access to the internal Met Office network.

Documentation

There are two levels of documentation on the **ticket** and in the relevant **documentation paper (UMDP)**. All changes must have a ticket description, and some may also require updates to one or model Unified Model documentation papers (UMDPs).

Ticket

For small changes, all that will be required is some text in the ticket **Description**, but for more involved changes a **ticket details** page will need to be added. To do this, modify the ticket to add the following at the end of the description:

```
[wiki:ticket/ticket_number/TicketDetails Ticket Details]
```

Save, and then click on the gray link. You will then be asked to create the page, and do so using the TicketDetails template from the drop-down menu. this will make a blank page containing the following:

```
= Ticket Details <#Ticket Nr> =

== Author: <name> ==

-----
== Branch ==

[log:main/branches/]

-----

== Documentation and Testing ==

Please add details here:


-----
```

You should add-in the information on the branch and the ticket etc., and then add details under the *Documentation and Testing* section. This should be as detailed as possible to describe what the ticket does. A blank example for this ticket/branch can be found here: https://code.metoffice.gov.uk/trac/um/wiki/ticket/3639/TicketDetails

UMDP

Please also see https://code.metoffice.gov.uk/trac/um/wiki/WorkingPractices/Documentation/UpdatingUMDPs

Note: Until UM vn11.0 it is not possible to generate UMDPs on the VM - this requires the change in #3612 (https://code.metoffice.gov.uk/trac/um/ticket/3612) , which has been committed at r47913 (https://code.metoffice.gov.uk/trac/um/changeset/47913) . This means it is possible to make a documentation branch from the head of the trunk (HoT) instead.

Small changes often do not require detailed documentation to be added - the ticket itself is usually sufficient. However, for changes to namelist variables (such as the change added here) the documentation must be updated. This does not need to be a big change, but the changes to the namelist need to be kept up-to-date. The documentation paper is a *living document* and so needs to reflect the state of the code that is being committed to the trunk.

A documentation branch is required for this, e.g.

```
fcm branch-create -k ticket_number your_documentation_branch_name fcm:um_doc.x_br@vn10.9
```

where `your_documentation_branch_name` will need to be different from `your_branch_name` to avoid confusion.

Virtual Machine Head of Trunk branch

To make a HoT branch on the VM the process is similar, e.g.

```
fcm branch-create -k ticket_number your_documentation_branch_name fcm:um_doc.x_br@HEAD
```

The name the branch created will now be something like rXXXXX_your_documentation_branch_name rather than vn10.9_your_documentation_branch_name. This is the only appreciable difference. If you wanted instead of `HEAD` you could also specify `r47913` instead.

For the VM to be able to build UMDPs correctly you need to have created a VM after 6th December 2017 that includes pull request #12 (https://github.com/metomi/metomi-vms/pull/12) as this adds support for LaTeX. You may also need to run `apt-get` to install a PDF viewer such as evince or xpdf.

You should then `checkout` your documenation branch and make the changes required. For this change they are:

`umdp84_app_control_vars.tex`

The file **`source/084/umdp84_app_control_vars.tex`** contains a table listing all the namelist variables used by UKCA. You will need to add information here:

```
Index: doc/branches/dev/lukeabraham/vn10.9_UKCA_Worked_Example_Doc/source/084/umdp84_app_control_vars.tex
===================================================================
--- a/doc/branches/dev/lukeabraham/vn10.9_UKCA_Worked_Example_Doc/source/084/umdp84_app_control_vars.tex
+++ b/doc/branches/dev/lukeabraham/vn10.9_UKCA_Worked_Example_Doc/source/084/umdp84_app_control_vars.tex
@@ -93,4 +93,6 @@
  L\_UKCA\_ASAD\_COLUMNS & True to pass columns to ASAD Newton-Raphson solver \\
                         & rather than latitude-longitude slices. \\
+ L\_UKCA\_LINOX\_LOGP  & True to redistribute Lightning NOx \\
+                       & emissions vertically using log(pressure) \\
  \hline
 \multicolumn{2}{|l|}{MODE Options} \\
```

umdp84_emissions.tex

A technical or scientific description of the change will also be required. Here this should be made to the **source/084/umdp84_emissions.tex** file, e.g.

```
Index: doc/branches/dev/lukeabraham/vn10.9_UKCA_Worked_Example_Doc/source/084/umdp84_emissions.tex
===================================================================
--- a/doc/branches/dev/lukeabraham/vn10.9_UKCA_Worked_Example_Doc/source/084/umdp84_emissions.tex
+++ b/doc/branches/dev/lukeabraham/vn10.9_UKCA_Worked_Example_Doc/source/084/umdp84_emissions.tex
@@ -265,4 +265,5 @@
 users to scale the derived Lightning NOx emissions by any factor between 0.0 and 5.0.

+If the control logical L\_UKCA\_LINOX\_LOGP is true, lightning NO$_x$ emissions will be vertically redistribute
linearly in pressure.

 \subsubsection{Sea salt primary emissions}
```

UM version

Unless you are working on a *head of trunk* branch, you will also need to update the UM version number in the UMDP to show that the change is in line with the next version. This is done in the **source/084/umdp84.tex** file:

```
Index: doc/branches/dev/lukeabraham/vn10.9_UKCA_Worked_Example_Doc/source/084/umdp84.tex
===================================================================
--- a/doc/branches/dev/lukeabraham/vn10.9_UKCA_Worked_Example_Doc/source/084/umdp84.tex
+++ b/doc/branches/dev/lukeabraham/vn10.9_UKCA_Worked_Example_Doc/source/084/umdp84.tex
@@ -3,5 +3,5 @@
 \title{United Kingdom Chemistry and Aerosol (UKCA) Technical Description}
 \paperno{084}
-\umversion{10.9}
+\umversion{11.0}
 \owner{Luke Abraham}
 \author{
```

This must be done for all changes to UMDPs

The documentation branch containing these changes can be seen here:
https://code.metoffice.gov.uk/trac/um/log/doc/branches/dev/lukeabraham/vn10.9_UKCA_Worked_Example_Doc

The UMDP containing all these changes can be found here: https://code.metoffice.gov.uk/trac/um/attachment/ticket/3639/Worked_Example-umdp_084.pdf

Next Steps

Ticket Summary

Once you've documented your ticket and made your UMDP, you will need to make a **ticket summary** prior to submitting the change to the trunk. A blank example can be found here: https://code.metoffice.gov.uk/trac/um/wiki/ticket/3639/TicketSummary

You will need to answer all the questions and append your final rose-stem output.

SciTech Review

When you have completed everything you will pass the ticket to someone else for **Sci/Tech Review**. This person may be the *code owner* of the (largest) section you are altering, or it may be someone else suitable who you have previously approached. They will then complete the Sci/Tech Review template and will either approve the change, or ask for further changes or clarification. If the latter, they will pass the ticket back to you, and when you're happy that you've answered all questions you should hand the ticket back to Sci/Tech Reviewer, and document the ticket, ticket details, or ticket summary with more information as appropriate.

When you pass the ticket to the Sci/Tech reviewer it is also a good idea to email the UM Systems Team at this point and ask for the name of the Code Reviewer, as the Sci/Tech Reviewer can pass it directly to them when the change is approved.

A blank Sci/Tech Review template can be found here: https://code.metoffice.gov.uk/trac/um/wiki/ticket/3639/SciTechReview

Code Review

Anyone with an SRS account can be a Sci/Tech Reviewer, but (generally) only members of the UM Systems Team can be Code Reviewers, as the Code Reviewer is the person who will commit the change to the trunk.

Once your change has passed Sci/Tech Review it will then be checked by the Code Reviewer for coding standards and consistency, whether the code change is linked to changes in other models, such as JULES, whether documentation has been made etc. A blank Code/System Review template can be seen here:
https://code.metoffice.gov.uk/trac/um/wiki/ticket/3639/CodeSystemReview

The Code Reviewer may have further changes or questions which will also need to be answered prior to the code being accepted for the UM trunk. They will pass the ticket back to you, and you will need to make changes and then pass the ticket back to them before they will respond. **It is important that the ownership of the ticket is changed to the correct person at each step**.

Once the code change has been approved the status of the ticket will be changed to *approved*, and when the code has been committed it will be *committed*. Then the code change will be automatically be checked overnight in the standard testing suites, and if there are no problems the ticket will be set to *closed* a few days later. At that point the Code Reviewer will change the ownership of the ticket to you again.

Congratuations! You have successfully had a change committed to the trunk!

Checklist

- □ Use `rose-stem` to test and develop your change.
- □ Remember to check for UMDP3 code compliance.
- □ Document your ticket sufficiently.
- □ You may need to make a test brach.
- □ You may need to update UMDPs. Remember to update the UM version number if you do.
- □ Remember to respond to all questions/corrections from your SciTech and Code reviewers.
- □ Remember to pass ownership of the ticket back to the appropriate reviewer when required.

UKCA Chemistry and Aerosol Tutorials at vn10.9

Written by Luke Abraham 2017

Retrieved from "http://www.ukca.ac.uk/wiki/index.php?title=UKCA_Chemistry_and_Aerosol_vn10.9_Tutorial_11&oldid=6756"

- This page was last modified on 26 January 2018, at 11:05.