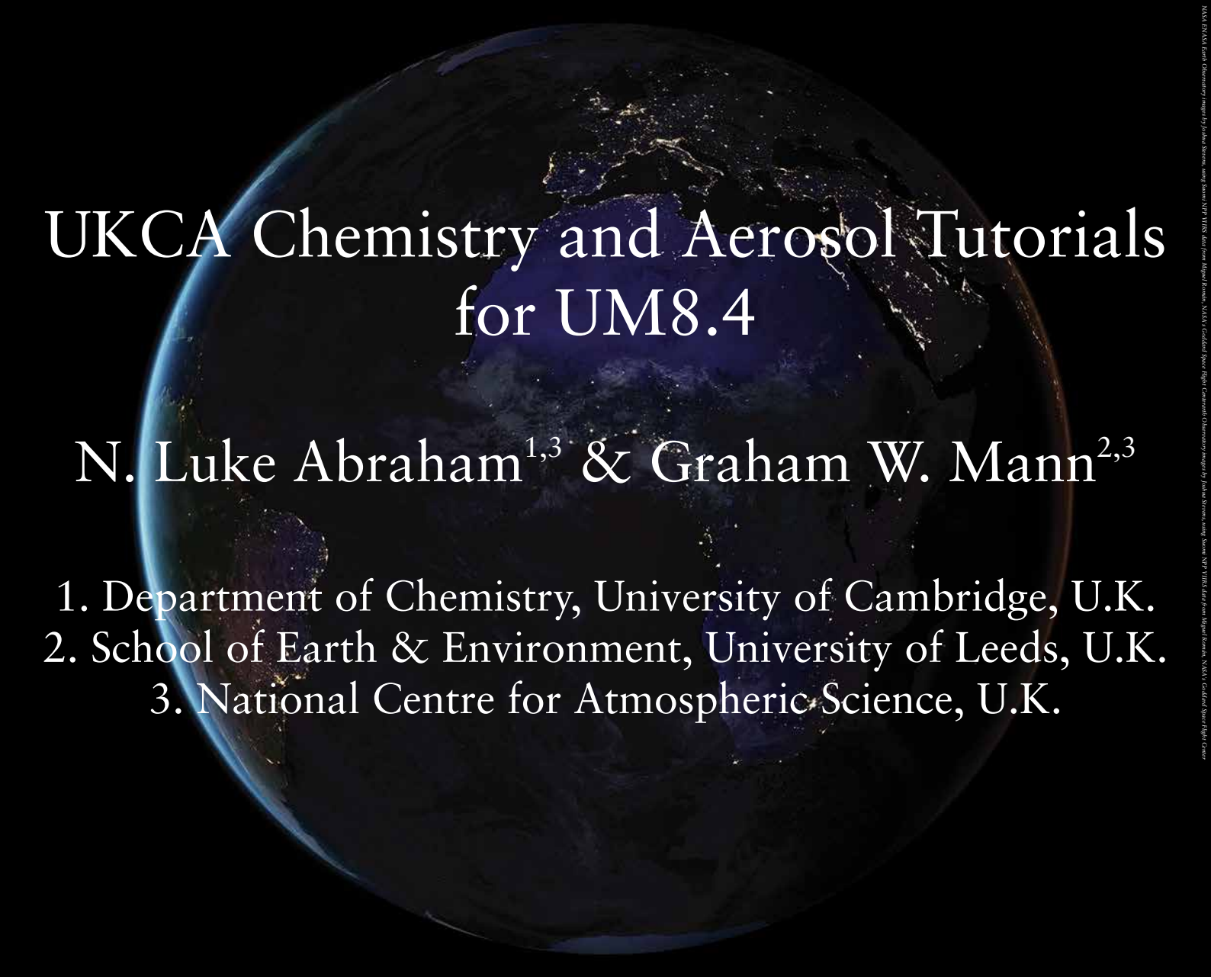


UKCA Chemistry and Aerosol Tutorials for UM8.4

N. Luke Abraham[1,3] & Graham W. Mann[2,3]

1. Department of Chemistry, University of Cambridge, U.K.
2. School of Earth & Environment, University of Leeds, U.K.
3. National Centre for Atmospheric Science, U.K.

https://doi.org/10.17863/CAM.22151

www.ukca.ac.uk/wiki/index.php/UKCA_Chemistry_and_Aerosol_Tutorials

February 2014

Acknowledgements

We would like to thank Alex Archibald, Ewa Bednarz, Antara Banerjee, Peer Nowack, Ines Heimann, David Wade, James Keeble, Paul Griffiths, Amanda Maycock, Mohit Dalvi, Colin 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 numbers NE/M006220/1 and NE/N000129/1.

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

From UKCA

UKCA Chemistry and Aerosol Tutorials for UM8.4

The following tutorials will teach you how to use various aspects of UKCA, and the parts of the UMUI (Unified Model User Interface) that are specific to UKCA.

- ### Things to know before you start

General use of the UMUI and UKCA:

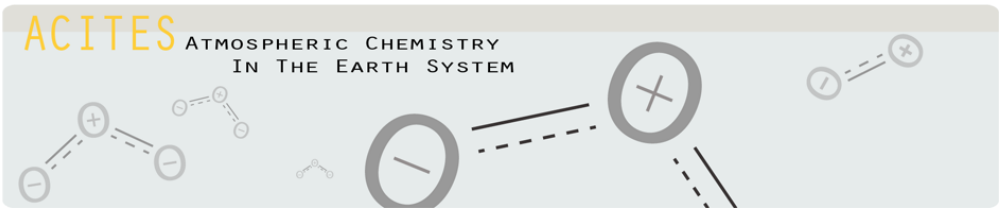
1. Running existing UKCA Job
2. Exploring the UMUI
3. What is STASH?
4. Adding new chemical tracers
5. Adding new emissions
6. Adding new chemical reactions
7. Adding dry deposition of chemical species
8. Adding wet deposition of chemical species
9. Adding new UKCA chemical diagnostics
10. Examining Aerosol Optical Depth changes
11. Changing the aerosol configuration
12. Diagnosing the radiative effect of aerosols
13. Example Rose Suite at vn10.2

The slides from the face-to-face workshop run in March 2014 can be found here: UKCA Tutorial Slides (//www.ukca.ac.uk/wiki/index.php/File:NLA_UKCA_Tutorial_Slides.pdf)

When you have completed these Tutorials please consider completing the **User Feedback Survey** (http://www.surveymonkey.com/s/G8VXHL2) .

These tutorials are an update and expansion of the UKCA & UMUI Tutorials which were at UM8.2. All the content is the same (although possibly in a slightly different order), but with additions for aerosol processes.

This tutorial has been funded by the ACITES Atmospheric Chemistry Modelling Network (http://www.ncas.ac.uk/index.php/en/what-is-acites) .



Written by Luke Abraham 2014

UKCA Chemistry and Aerosol Tutorials: Things to know before you start

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

You should not use the UKCA Tutoral jobs for scientific studies. They are designed to be used as training aids.

UKCA jobs at UM8.4 are currently being prepared and scientifically validated prior to release.

Contents

- 1 PUMA and ARCHER/MONsooN
- 2 UM Training and the FCM Tutorial
- 3 Location of output data files
- 4 Knowledge of scripting and programming languages
- 5 Structure
  - 5.1 UKCA Chemistry Tutorials
    - 5.1.1 Solutions
- 6 Model Configuration
- 7 Data Manipulation and Plotting
  - 7.1 Command-line tools
    - 7.1.1 convsh
    - 7.1.2 nco
    - 7.1.3 cdo
- 8 Further Information

PUMA and ARCHER/MONsooN

Before you begin this tutorial you should first get a PUMA and an ARCHER or MONSooN account as the UKCA Tutorial is based around jobs which run on these supercomputers, using PUMA to submit them.

Further information about how to apply for these account can be found on the **Getting Started with UKCA** page.

This tutorial can be run on both ARCHER and MONSooN. In the UMUI **Search → Filter...** for:

- The ARCHER experiment is **xjrn**
- ~~The MONSooN experiment is **xjrj**~~ **MONSooN Tutorials do not currently work on the XCM**

UM Training and the FCM Tutorial

There is some information about the UM here:

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

You must also have completed the NCAS-CMS UM FCM Tutorial (http://cms.ncas.ac.uk/wiki/Fcm) . This tutorial assumes that you have had some experience in using the UM, and know how to copy an existing job to your account and run it.

If you have never used the UM before you should also view the NCAS-CMS UM Training (http://cms.ncas.ac.uk/wiki/UmTraining) pages, which has a basic introduction to the UMUI (http://cms.ncas.ac.uk/wiki/UmTraining/IntroToUMUI) .

You will also find the NCAS-CMS pages very useful: cms.ncas.ac.uk (http://cms.ncas.ac.uk/) .

Location of output data files

By default the UKCA Tutorial jobs are not set to archive any data, and so the output data files can be found in the

```
/work/n02/n02/userid/jobid
```

directory on ARCHER, and in the

```
~~/projects/ukca/userid/jobid~~
```

directory on MONSooN (if you are not in the **ukca** project group you will need to set your project group here).

For consistency these will be known as your **job directory** throughout the rest of this tutorial.

Archiving (where data may be processed and moved to another disk or to tape) is controlled in the UMUI in the **Model Selection → Post Processing → Main Switch + General Questions** panel.

Knowledge of scripting and programming languages

UKCA (and the UM) is written in Fortran 90, and so you should have some knowledge of this language, as well as some knowledge of the Unix/Linux command-line. You will not be asked to do anything advanced, but you will need to add to arrays and other code blocks.

Structure

UKCA is not one particular chemistry scheme, but is in fact a **framework** for adding a chemistry scheme to the UM. Although in this tutorial you will adapt an exisiting scheme, you could extend this to replacing all tracers and reactions and add in a completely different scheme.

UKCA Chemistry Tutorials

These Tutorials are actually one large problem which someone who uses UKCA may be required to do: add some tracers, reactions, deposition, and diagnostics. To make this problem more manageable it has broken down into a series of small **tasks** which you will be asked to complete. Each Tutorial explains how to make the necessary changes (in general terms), and then you will be asked to make the required changes to your UKCA job.

To accompany each Task which requires UMUI or UKCA code changes, there are worked solutions (held in the same UMUI Experiment as the original Tutorial Base Job) along with sample output and a wiki page detailing the changes in full. If at any point you have difficulty completing one of the Tasks you can take a copy of the worked solution job and carry on from there. There is also sample output provided for each of the tasks.

Solutions

The solutions to these tasks can be found in the UMUI (as other jobs in the Tutorial experiments **xirb** and **xirc**) under the **ukca** user. The code-changes required can be viewed at the vn8.4_UKCA_Tutorial_Solns PUMA Trac page (https://puma.nerc.ac.uk/trac/UM/log/UM/branches/dev/luke/vn8.4_UKCA_Tutorial_Solns) (password required).

Model Configuration

The UKCA Tutorial job is at UM version 8.4 job using a Global Atmosphere 4.0 atmosphere-only configuration. It has a resolution of N96L85 (1.875 degrees by 1.25 degrees, with 85 vertical levels up to 85km). The current UKCA Chemistry and Aerosol Tutorial job uses the **CheST/StratTrop+GLOMAP-mode** chemistry and aerosol schemes.

While you may be interested in using UKCA at a different UM version, many aspects of this tutorial will be directly transferable. For instance, many of the UMUI panels and much of the UKCA source code are the same at UM7.3 and UM8.2 as at UM8.4.

Data Manipulation and Plotting

In these Tutorials you will only be asked to use Xconv (http://badc.nerc.ac.uk/help/software/xconv/) for basic data manipulation and plotting. Xconv can be found on ARCHER at

```
/work/n02/n02/hum/bin/xconv
```

and on the MONSooN ibm02 at

```
/projects/um1/bin/xconv
```

and the MONSooN postproc at

```
/usr/local/bin/xconv
```

and which can also be used to extract UM fields as netCDF. Xconv should already be in your `$PATH`.

Both IDL (using the Met Office library (http://cms.ncas.ac.uk/documents/IDL/idl_guide.html)) and Python (using either cf-python (http://cfpython.bitbucket.org/) or Iris (http://scitools.org.uk/iris/)) are able read the UM PP/FieldsFile format directly, although use of these tools will not be covered by these Tutorials.

Command-line tools

There are command-line tools available that can manipulate fieldsfiles/pp-files and netCDF files in various ways.

convsh

Convsh is the command-line backend of Xconv, and can be found in the same directories as Xconv on the supercomputer, i.e.

```
/work/n02/n02/hum/bin/convsh
```

on ARCHER,

```
/projects/um1/bin/convsh
```

on the MONSooN ibm02, and

```
/usr/local/bin/convsh
```

on the MONSooN postproc.

A handy list of convsh commands is available (http://www.met.reading.ac.uk/~jeff/xconv/appx1.html) .

nco

The netCDF operators (http://nco.sourceforge.net/) (nco) are a set of generic netCDF manipulation tools. On ARCHER, you can load nco by

```
module swap PrgEnv-cray PrgEnv-gnu
module load nco
```

You need to swap out the PrgEnv-cray environment (used with the UM) to the PrgEnv-gnu environment, as nco has been compiled with the gnu compiler.

This will now point your `$PATH` to see the location of ncdump in

```
/opt/cray/netCDF/4.3.0/bin
```

and the netCDF operators in

```
/work/y07/y07/cse/nco/4.4.2/install/bin
```

for nco version 4.4.2.

On the MONSooN postproc ncdump can be found at

```
/usr/bin/ncdump
```

and the netCDF operators can be found in

```
/project/ukmo/rhel6/nco/bin
```

cdo

The climate data operators (https://code.zmaw.de/projects/cdo) (cdo) are an expanded set of command line tools that are designed to work with netCDF files that contain climate data. They can be loaded on ARCHER by

```
module swap PrgEnv-cray PrgEnv-gnu
module load cdo
```

You need to swap out the PrgEnv-cray environment (used with the UM) to the PrgEnv-gnu environment, as cdo has been compiled with the gnu compiler.

This will now point your `$PATH` to see the location of cdo at

```
/work/y07/y07/cse/cdo/1.6.3-build1/bin/cdo
```

for cdo version 1.6.3.

On the MONSooN postproc cdo version 1.5.4 can be found at

```
/opt/ukmo/utils/bin/cdo
```

Further Information

More information on UKCA can be found by browsing this wiki. More detailed information on UKCA can also be found in the The UKCA documentation paper for vn8.4 of the MetUM (//www.ukca.ac.uk/wiki/index.php/File:Umdp_084-umdp84.pdf).

Written by Luke Abraham 2014

- This page was last modified on 15 December 2015, at 15:00.

UKCA Chemistry and Aerosol Tutorial 1

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Contents

- 1 What you will do in this tutorial
- 2 Getting hold of an existing UM-UKCA job
- 3 Running an existing UKCA job
  - 3.1 Task 1.1: Copy a UM-UKCA job and then run it
- 4 Checking the progress of a running job
- 5 Viewing and extracting output
  - 5.1 ncdump
- 6 .leave Files
  - 6.1 Compilation Output (.comp.leave)
  - 6.2 Reconfiguration Output (.rcf.leave)
  - 6.3 Model Output (.leave)

What you will do in this tutorial

In this tutorial you will now take a copy of a UKCA job and send it to compile and run on the supercomputer. We will then learn about how to check if the job is running, and also about some of the output files produced.

Getting hold of an existing UM-UKCA job

Before you can start using UKCA (and the UM generally), you first need to copy a UM or UKCA job from somewhere. You may be given a job from someone for a specific purpose, or you may just copy a standard job. In the UMUI there are two repositories of standard jobs, under the **umui** and **ukca** UMUI users. The **umui** user contains a series of standard jobs provided and maintained by NCAS-CMS. The **ukca** user contains jobs provided by the UKCA team.

Running an existing UKCA job

You will need to change a number of options within the UMUI to allow you to run this job successfully, such as your username, ARCHER TIC-code (if needed) etc. If you are using the MONSooN job you may also need to change the project group in

```
Model Selection
-> Post Processing
 -> Main Switch + General Questions
```

if you want to send output data to the /nerc data disk (this is advisable). The NCAS-CMS UMUI Training Video (http://cms.ncas.ac.uk/wiki/UmTraining/IntroToUMUI) will give you the minimum information that you need to be able to make these changes.

Task 1.1: Copy a UM-UKCA job and then run it

More details on how to copy UMUI jobs can be found in the NCAS-CMS Introduction to the UMUI (http://cms.ncas.ac.uk/wiki/UmTraining/IntroToUMUI) tutorial video.

TASK 1.1: Make a new experiment and take a copy of the UKCA Tutorial Base Job, filter for the **ukca** user and choose the correct experiment for the machine that you will be running on (ARCHER: **xjrn**, ~~MONSooN: **xjrj**~~). Select the **a** job, labeled ***Tutorial: Base UM-UKCA Chemistry and Aerosol Job*** and copy this one to your own experiment.

Now take your copy of the Tutorial Base Job and make the required changes to allow this job to run. Once you have made these changes you can submit your job. First click **Save**, then **Process**, and once this has completed, click **Submit**. This will then extract the code from the FCM repositories and submit them to the supercomputer. If you are running on MONSooN you will need to enter your passcode at this stage.

Note: To allow the jobs in this tutorial to run quickly this job is only set to run for 2 days. This means that there will be no **climate-mean** files produced (see the what is STASH? tutorial) produced, which require run lengths of a month or more.

Sample output from this job can be found in

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Base/
```

on ARCHER, and in

```
~~/projects/ukca/Tutorial/vn8.4/sample_ouput/Base/~~
```

on MONSooN.

Checking the progress of a running job

Log-in to the supercomputer, and check that your job is running. For ARCHER do

```
qstat -u $USER
```

and for MONSooN do

```
llq -u $USER
```

This should give a list of your running jobs. For example, on ARCHER you get output similar to

```
$ qstat -u $USER

sdb:
                                                            Req'd  Req'd   Elap
Job ID          Username Queue    Jobname    SessID NDS TSK Memory Time  S Time
--------------- -------- -------- ---------- ------ --- --- ------ ----- - -----
1515659.sdb     luke     par:8n_2 xjqka_run    7934   1   1    --  00:10 R 00:05
```

and on MONSooN you should get something like

```
$ llq -u $USER
Id                       Owner      Submitted   ST PRI Class        Running On
------------------------ ---------- ----------- -- --- ------------ -----------
mon001.64641.0           nlabra      6/5  12:36 R  50  parallel     c139

1 job step(s) in query, 0 waiting, 0 pending, 1 running, 0 held, 0 preempted
```

You can also check how far a job has gone while it is running. To do this you will need to `cd` into the job directory (this will be on your `/work` space on ARCHER or your `/projects` space on MONSooN). When you do this, you will see something like this

```
$ ls
baserepos/  history_archive/  umatmos/  umscripts/    xjlla.list      xjlla.stash    xjlla.xhist        xjllaa.p
bin/        pe_output/        umrecon/  xjlla.astart  xjlla.requests  xjlla.umui.nl  xjllaa.pa20051201
```

Now `cd` into the `pe_output/` directory and do

```
$ tail -f jobid.fort6.pe0 | grep Atm_Step
Atm_Step: Timestep       67   Model time:   2005-12-01 22:20:00
Atm_Step: Timestep       68   Model time:   2005-12-01 22:40:00
Atm_Step: Timestep       69   Model time:   2005-12-01 23:00:00
Atm_Step: Timestep       70   Model time:   2005-12-01 23:20:00
Atm_Step: Timestep       71   Model time:   2005-12-01 23:40:00
Atm_Step: Timestep       72   Model time:   2005-12-02 00:00:00
```

(changing jobid as appropriate for your job).

Viewing and extracting output

To take a look at the output, you will need to change into your **job directory**. Once in this directory `ls` to see the file listing. All the output data is contained in files with the naming convention of

```
jobida.pzYYYYMMDD
```

(e.g. `xjllaa.pa20051201`) or

```
jobida.pzYYYYmmm
```

(e.g. `xjcina.pb2006dec`). The what is STASH? tutorial will discuss the various output streams (denoted by the letter **z** above) will be discussed in more detail.

Restart files have a similar naming strategy:

```
jobida.daYYYYMMDD_HH
```

(e.g. `xjcina.da20070201_00`). These files are known as **dumps**.

To see what output files have been produced, do

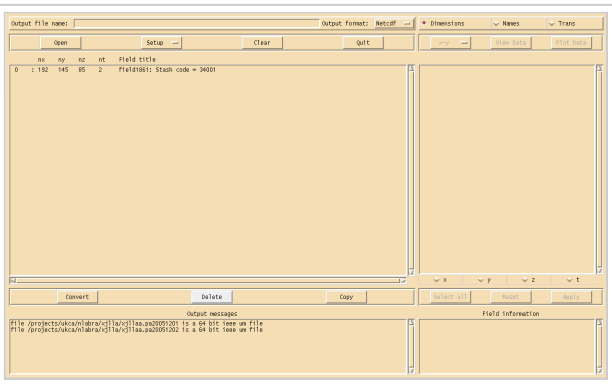


Figure 1: Xconv viewing the **pa** file.

```
$ ls *.p*
xjllaa.pa20051201
```

As you can see, there is one file present, the "**pa**" file. This file is a daily file that has come from the UPA PP stream (standard PP files will be covered in more detail in the What is STASH? tutorial. To quickly view output you can use Xconv (http://badc.nerc.ac.uk/help/software/xconv/) , which provides a simple data viewer. It can also be used to convert the UM format output files to netCDF.

You can open these files by

```
$ xconv -i *.pa*
```

which will show the Xconv window as can been seen in Figure 1. There is only one field present

```
0    : 192   145   85    1     Stash code = 34001
```

This is the UKCA chemical ozone tracer (although it is not labeled as such by default). A full listing of all UKCA fields can be found in the listing of UKCA fields at UM8.2 (which is the same in UM8.4). More information will be given on STASH in the What is STASH? tutorial.

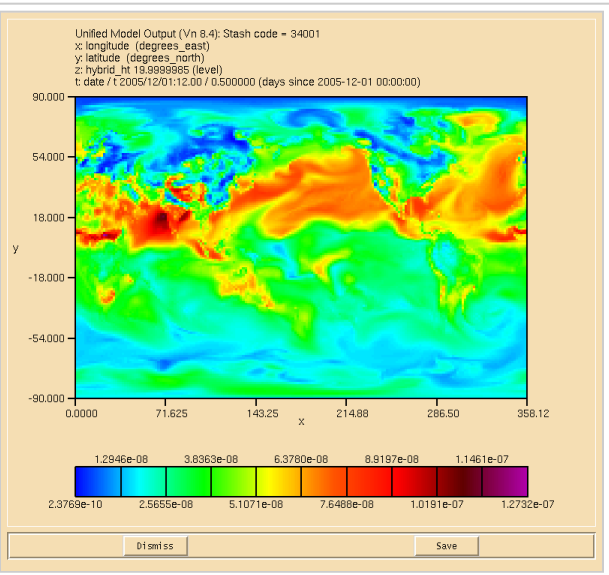


Figure 1: Xconv viewing surface O3 concentration.

You can use Xconv to view certain fields. For example, you could view the surface ozone concentration double-clicking on the *Stash code = 34001* field and clicking the **Plot data** button (see Figure 2). While this is good to quickly check data, the plotting functions are rather limited as it is not possible to change e.g. the colour-bar, the scale, add a map projection etc. It is advisable to either export fields as netCDF from within Xconv, or to use another program, such as IDL (using the Met Office library (http://cms.ncas.ac.uk/documents/IDL/idl_guide.html)) or Python (using either cf-python (http://cfpython.bitbucket.org/) or Iris (http://scitools.org.uk/iris/)) which is able to read the UM PP/FieldsFile format directly.

To export fields as netCDF select them using the mouse (they should then highlight blue), enter a name for the netCDF file in the **Output file name** box (making sure that the *Output format* is *Netcdf*) and click the **Convert** button. The window on the bottom right will show the progress of the conversion. For single fields this is usually quite quick, but it is possible to use Xconv to open multiple files containing a series of times. In this case Xconv will combine all the individual times into a single field, and outputting this can take some time.

One issue you may have is that Xconv uses a quantity called the *field code* to determine the variable name of each field (the netCDF name attribute). For UKCA tracer fields at UM8.2 this code is all the same, so all variables will be called *field1861*. It is possible to change the short field name in Xconv, prior to outputting a netCDF file. Select the variable you wish to output and select the **Names** button on the top-right of the Xconv window. Delete the contents of the **short field name** box and replace it with what you would like, e.g. for ozone (Stash code 34001) you may wish to use the CF standard name *mass_fraction_of_ozone_in_air* (as the units of UKCA tracers are kg(species)/kg(air)). The click **apply** and output the field as normal. When running `ncdump` on the resultant netCDF file you should see something like

```
      float mass_fraction_of_ozone_in_air(t, hybrid_ht, latitude, longitude) ;
              mass_fraction_of_ozone_in_air:source = "Unified Model Output (Vn 8.4):" ;
              mass_fraction_of_ozone_in_air:name = "mass_fraction_of_ozone_in_air" ;
              mass_fraction_of_ozone_in_air:title = "Stash code = 34001" ;
              mass_fraction_of_ozone_in_air:date = "01/12/05" ;
              mass_fraction_of_ozone_in_air:time = "00:00" ;
              mass_fraction_of_ozone_in_air:long_name = "Stash code = 34001" ;
              mass_fraction_of_ozone_in_air:units = " " ;
              mass_fraction_of_ozone_in_air:missing_value = 2.e+20f ;
              mass_fraction_of_ozone_in_air:_FillValue = 2.e+20f ;
              mass_fraction_of_ozone_in_air:valid_min = 2.29885e-10f ;
              mass_fraction_of_ozone_in_air:valid_max = 1.839324e-05f ;
```

Once you have your data as netCDF it is then possible to use any standard visualisation or processing package to view and manipulate the data.

ncdump

On MONSooN ncdump is located in

```
/home/accowa/MTOOLS/bin
```

and on ARCHER to use ncdump you will first need to

```
module swap PrgEnv-cray PrgEnv-gnu
module load nco
```

and it should then be available in your $PATH. You should not put these lines in your .profile or .bashrc file however, as this may affect your use of the UM.

.leave Files

The text output from any write statements within the code, or giving information about compilation, is outputted to several files with the extension **.leave**. These will be in your `$HOME/output` directory on ARCHER and MONSooN.

You will have three .leave files, one for the compilation, one for the reconfiguration step (if run), and one for the UM itself. By default for climate runs these will all have a common format, starting with 4 blocks of letters and numbers, like this:

```
xipfa000.xipfa.d13163.t120017
```

where this breaks down to

jobidXXX	e.g. xipfa000	The jobid of the job, followed by the job-step number. For compilation and reconfiguration jobs, this will be 000, but as the CRUN progresses this number will increment by 1 for each step, and then cycle round back through 000 (if you run more than 999 steps).
jobid	e.g. xipfa	The jobid of the job as listed in the UMUI.
dXXXXX	e.g. 13163	The year (the last two digits, i.e. 2013 is **13**) and the day of the year as 3 digits (i.e. 001-366, so this file was created on the 12th June (day 163)).
tXXXXXX	e.g. 120017	The time in **HHMMSS** format, as recorded by the system clock on the supercomputer.

Using this format this means that file was created on the 12th June 2013 at 12:00:17. Note that the timestamp on the file will be later than this, as this is the time the file was created, not the time that it was last written to.

There are then three file extensions: **.comp.leave** for compilation output, **.rcf.leave** for reconfiguration output, and **.leave** for the model output.

It is often easier to list your files in this directory by date, but using `ls -ltr`.

Compilation Output (.comp.leave)

This gives the output from either the XLF compiler on MONSooN or the Cray compiler on ARCHER. If the compilation step has an error and the code is not compiled you can find the source of the error by opening this file and searching for **failed** - this will highlight which routine(s) caused the problem. You may also get more detailed information such as the line number which had the error. In this case you can open the file on the supercomputer and view the line, as the line number given will not match with the line in your working directory on PUMA due to merging source code and the use of include files. Remember to make any required changes to your PUMA source code however!

Reconfiguration Output (.rcf.leave)

This gives output from the reconfiguration step, if run. At older UM versions, such as UM7.3 this output was found in the model output .leave file.

Model Output (.leave)

This gives output from the code which is generated as it is running, although this file is only updated and closed when the job finishes. To view the output generated as it is running you will need to see the output in the `pe_output/` directory mentioned above.

To run efficiently the UM is split into many domains, which communicate with each other with parallel calls, during runtime. The exact decomposition is defined in *Model Selection → User Information and Submit Method → Job submission method*, in the number of processes East-West and North South boxes. If you have a 12x8 decomposition there will be 96 processes, running on 96 cores of the supercomputer (3 nodes of MONSooN, 4 nodes of ARCHER). These processes will be numbered internally from 0 → 95, labelled as **PE0** to e.g. **PE95**. Only the output from PE0 will be sent to the .leave file, with output from the other PEs only held in the `pe_output/` directory. Whether or not these files are deleted at the end of a run is set in the UMUI in **Model Selection → Input/Output Control and Resources → Output management** panel. If you run fails then these files will not be deleted.

While there is a lot of information outputted to the .leave file, and you would usually only read it if the job fails, it is worth going through the messages, making a special note of any warnings.

If you see the line

```
jobid: Run terminated normally
```

around line 48 then your job has successfully run.

Written by Luke Abraham 2014

- This page was last modified on 15 December 2015, at 15:02.

UKCA Chemistry and Aerosol Tutorial 2

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Contents

- 1 What you will do in this tutorial
- 2 Exploring the UMUI
  - 2.1 What is the UMUI?
    - 2.1.1 Task 2.1: Explore your new UKCA job
  - 2.2 Closing UMUI panels
- 3 The UMUI: Quick Reference
  - 3.1 The Main Window and the Interface Bar
    - 3.1.1 Search
    - 3.1.2 Experiment
    - 3.1.3 Job
  - 3.2 Navigating a Job
    - 3.2.1 User information and Submit Method
      - 3.2.1.1 General details
      - 3.2.1.2 Job submission method
    - 3.2.2 Input/Output Control and Resources
      - 3.2.2.1 Output Choices
      - 3.2.2.2 Time Convention and SCRIPT Environment Variables
      - 3.2.2.3 Start Date and Run Length Options
      - 3.2.2.4 Re-submission pattern
    - 3.2.3 FCM Configuration
      - 3.2.3.1 FCM Extract directories and Output levels
      - 3.2.3.2 FCM Options for Atmosphere and Reconfiguration
    - 3.2.4 Compilation and Run Options
      - 3.2.4.1 Compile and run options for Atmosphere and Reconfiguration
      - 3.2.4.2 UM User Override Files
    - 3.2.5 Reconfiguration
      - 3.2.5.1 General Reconfiguration Options
    - 3.2.6 Post Processing
      - 3.2.6.1 Main Switch + General Questions
    - 3.2.7 Atmosphere
      - 3.2.7.1 Model Resolution and Domain → Vertical
      - 3.2.7.2 Model Configuration → UKCA Chemistry and Aerosols
      - 3.2.7.3 Scientific Parameters and Sections → Section by section choices
      - 3.2.7.4 Scientific Parameters and Sections → Spec of trace gases
      - 3.2.7.5 Ancillary and input data files → Start dump
      - 3.2.7.6 Ancillary and input data files → In file related options
      - 3.2.7.7 Ancillary and input data files → Climatologies & potential climatologies
      - 3.2.7.8 Ancillary and input data files → Other ancillary files and Lateral Boundary files
      - 3.2.7.9 STASH
    - 3.2.8 The UKCA Panel
      - 3.2.8.1 Main UKCA Panel
      - 3.2.8.2 Main UKCA Panel → NEXT follow-on window
      - 3.2.8.3 Main UKCA Panel → PHOTO follow-on window
      - 3.2.8.4 Main UKCA Panel → LOWBC follow-on window
      - 3.2.8.5 Main UKCA Panel → COUPL follow-on window
      - 3.2.8.6 Main UKCA Panel → UKCA_TRA follow-on window

What you will do in this tutorial

In this tutorial you will will explore various UMUI panels that you may find useful in your future work with UKCA. This can be done as fast or as slowly as you like, although future tutorials will assume that you have a general knowledge of the UMUI.

Exploring the UMUI

What is the UMUI?

The UMUI is the *Unified Model User Interface*. It is used to configure UM model options and parameters, and, for climate simulations, is also used to manage submission of the job to the supercomputer. As well as inserting values into the Fortran namelists used by the model, it also checks the logical of those choices. It is also possible to add variables into the namelists directly using **hand-edits**, which will be discussed in more detail in the adding new chemical tracers tutorial.

The UMUI itself is rather complicated, with many different panels, but no search function. Without experience it can be difficult to find which panels are useful. While this tutorial will go through many of the panels that are useful or relevant to using UKCA, it is recommended that you take some time to familiarise yourself generally with the UMUI.

This list is not exhaustive, but should be useful as a reference. Only the panels that are directly relevant to configuring and running UKCA jobs will be discussed in any detail.

Task 2.1: Explore your new UKCA job

TASK 2.1: Open your job and browse around it as you read through the list below.

If in later tutorials you find that you are having problems running you job, you may have accidentally made changes to it while you look through the UMUI now. You can see if this is the case by **differencing** your job with the original **a** job that you copied (*Tutorial: Base UM-UKCA Chemistry Job*). To do this you first need to **Search → Filter...** for both your experiment and the UKCA Tutorial experiment, then go to **Job → Difference** and select **Long**. The only differences between these jobs should be in the **Model Selection → User Information and Submit Method → General details** (called **personal_gen** in the difference window) where you have changed your user-name, email address, and TIC-code.

If you have had problems and have had to revert the job to the original **a** job, you may find that you need to clear all the directories that have been produced on the supercomputer. On ARCHER you may need to remove the

```
/home/n02/n02/userid/um/jobid
/work/n02/n02/userid/jobid
```

directories, and on MONSooN you may need to remove the

```
/projects/group/userid/jobid
```

directories.

Sample output from this job can be found in

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Base/
```

on ARCHER, and in

```
/projects/ukca/Tutorial/vn8.4/sample_ouput/Base/
```

on MONSooN.

Closing UMUI panels

You should be careful when closing panels in the UMUI. You should never close a panel by clicking the **x** in the corner of the window as this may result in the error

```
ERROR: addToWindowList: Table envirs1 already exists
```

If you want to save the changes that you have made, click **close** and if you don't want to save them, click **abandon changes**.

It is very easy, when exploring a UMUI job, to accidentally make a change. For this reason if you don't want to change anything on a panel it is best to exit by using the *abandon changes* button, rather than the *close* button as this will save any accidental and unwanted changes that you may have made.

The UMUI: Quick Reference

The Main Window and the Interface Bar

When you start-up the UMUI you will be given a list of all your experiments. Along the top of this window there is a bar which gives a number of useful options

Search

The **Filter** option is very useful. This allows you to filter UMUI jobs by user, experiment ID, version etc. You will mostly be copying a job from another user (as you have done for this tutorial) and so will need to use the search function for this.

Experiment

This menu is used primarily to make a **New** experiment. It is advisable to use different experiments for different studies, as keeping multiple studies in a single experiment can become confusing. Each experiment has a unique 4-letter identifier, and you can have up to 26 jobs under each experiment, labelled a-z.

Job

You will often use the Job menu to **Copy** jobs from other users experiments, as described in the NCAS-CMS Introduction to the UMUI (http://cms.ncas.ac.uk/wiki/UmTraining/IntroToUMUI) tutorial video. Another useful function is the **Difference** option, which allows you to compare, in detail, two different UMUI jobs (of the same UM version).

You can also use this menu to change the **job description** text or the **job identifier** (i.e. the a-z label) within the experiment.

If the UMUI is not closed down correctly, when you start it again and try to edit an existing job which you had open when the UMUI crashed you will get an error message telling you that the job is already open. In this case you can **Force Close** the job from this menu.

There is also a **Help** menu which may be useful.

Navigating a Job

When you open a job from the UMUI, either as read-write or as read-only, you are given a new window with several buttons along the bottom and a single tab labeled *Model Selection*. Clicking on this tab will give you a number of options, and we will go over the more relevant of these below.

User information and Submit Method

General details

In this panel you will set your user-id on the remote supercomputer (the **$USERID** variable), your email address (which may be different from your user-id), and the accounting (or *TIC*) code. These changes will also have been covered in the NCAS-CMS Introduction to the UMUI (http://cms.ncas.ac.uk/wiki/UmTraining/IntroToUMUI) tutorial video.

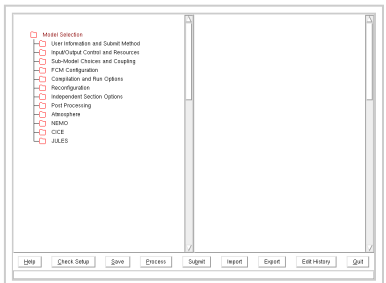


Figure 1: A UMUI Job showing the Input/Output Control and Resources options.

Job submission method

This panel determines what computer you will be running on, and how long the NRUN step (see information on the *Compile and Run Options* below) will request in the supercomputer queue. To access this information you will need to select either **LoadLev** (for MONSooN) or **Qsub** (for ARCHER).

It is unlikely you will need to change any information in this panel.

Input/Output Control and Resources

Output Choices

In this window you can change the level information given by the output messages that are placed in the mode output **.leave** file (for more information on this file, see the running existing UKCA Job tutorial.

Time Convention and SCRIPT Environment Variables

Often the directory containing an input ancillary file is not specified explicitly, but is instead specified using an environment variable (e.g. `$UKCA_EMISS`). These environment variables are set here.

This panel also defines the directories where the job will be run from, `$DATAM` and `$DATAW`.

Start Date and Run Length Options

A simple panel where the start date of the run and the number of years, months, days, etc of the run is set. It is best to use, e.g. 1 month rather than 30 days, or 1 year rather than 12 months.

Re-submission pattern

This panel sets whether or not re-submission is used (for most jobs over a few months in length, re-submission will always be used) and how long that job-step has requested in the queue on the supercomputer. This number may be different to that set in the *User information and Submit Method → Job submission method* window.

FCM Configuration

FCM Extract directories and Output levels

This panel contains the directory that your source code will be extracted to, prior to compilation (the **UM_ROUTDIR** field). You may need to change this field. For ARCHER users this should be something like

```
/home/n02/n02/$USERID/um
```

as you should to extract to `/home` and not `/work`. For MONSooN users it is often the same as the directory your job will run from, which will be similar to

```
/projects/group/$USERID/um
```

(setting the name of your MONSooN project group as appropriate). Note that you will have already set the `$USERID` value in the **User Information and Submit Method → General details** panel.

In this panel you can also force FCM to run a full extract and build of an existing directory, rather than just doing incremental extracts.

FCM Options for Atmosphere and Reconfiguration

This panel tells the UM which branches from the FCM repository to include in the build, or if you want to take source code from a working copy on PUMA. When you are adding new source code you will need to make changes here.

Compilation and Run Options

Compile and run options for Atmosphere and Reconfiguration

An example of this window can be seen in Figure 2. This window is used to tell the UMUI if you want to compile and/or run the model. If you are just running the model you can also select a **CRUN** (*continuation run*) step, which is done during the automatic re-submission.

This window gives the option to run the reconfiguration step or not. Reconfiguration will be discussed in more detail below.

UM User Override Files

In this panel you can define over-ride files which are used to change the compiler flags for some (or all) source files. It may be that you need to change the optimisation level of a routine, for example, and this is done here.

Reconfiguration

The reconfiguration step is used to create the initial conditions (*.astart*) file for the atmosphere model. Often this is done by supplying an existing restart (or *dump*) file produced by another model run. If no changes need to be made to this file then the reconfiguration step does not need to be run, but if anything does need to be done, such as changing the year of the dump file, or inserting a new field (as will be done in the adding new chemical tracers tutorial then this step needs to be performed.

General Reconfiguration Options

If reconfiguration has been requested in the *Compilation and Run Options → Compile and run options for Atmosphere and Reconfiguration* panel, then you can set more switches here, such as changing the year in the file. More reconfiguration options are also set in *Atmosphere → Ancillary and input data files → Start dump*.

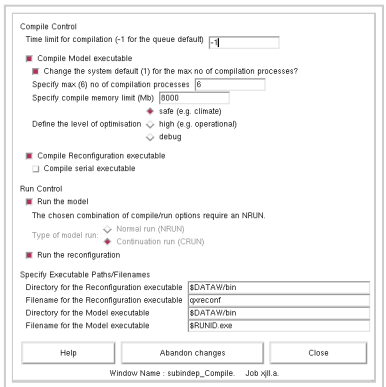


Figure 2: The compilation options window.

Post Processing

More information on the *Initialization and processing of mean & standard PP files* panel will be covered in the What is STASH? tutorial.

Main Switch + General Questions

This panel covers post-processing to be applied to a job as it runs, such as deleting superseded files, and the **archiving** of data to various places, depending on the supercomputer.

On ARCHER, you can choose to archive your data to the

```
/work/n02/n02/userid/jobid/archive
```

directory, which sits within your main **job directory**. On MONSooN this panel can be used to send data to the `/nerc` disk, or to the MOOSE (password required) (http://collab.metoffice.gov.uk/twiki/bin/view/Support/MASSMooseOverview) tape archive.

Atmosphere

The atmosphere tab is used to control the majority of the settings in the UM model, and so it is divided into a number of sub-levels. A few of the most relevant to UKCA are discussed below.

Model Resolution and Domain → Vertical

This panel gives the location (on the supercomputer) of the namelist giving the specification of the level structure. You will need this file to use with Xancil (http://cms.ncas.ac.uk/documents/xancil/) when producing 3D ancillary and initial condition files. Xancil will be discussed in more detail in the adding new emissions tutorial and in the Using Xancil page..

Model Configuration → UKCA Chemistry and Aerosols

This is a link to the UKCA panel (which can also be found in *Atmosphere → Scientific Parameters and Sections → Section by section choices → Section 34: UKCA Chemistry and Aerosols*). This will be dicussed in detail below.

Scientific Parameters and Sections → Section by section choices

The panels in this menu contain a large number of the UM sections (including UKCA) and various options for each of these sections can be selected, and viewed, by clicking on the required tab, e.g. *Section 1: SW Radiation* etc.

Scientific Parameters and Sections → Spec of trace gases

The panels in this menu allow the values of the trace gases of CH4, N2O, CFC-11, CFC-12, CFC-113, CFC-114, HCFC-22, HFC-125, HFC-134a, and CO2 to be set for use in the radiation scheme. Depending on settings in the UKCA panel, UKCA can use these values as the lower boundary conditions for the mixing ratios.

Ancillary and input data files → Start dump

In here you set the original restart (**dump**) file that is read-in by the model, and the type of interpolation that will be applied to it, if it doesn't match the existing model resolution (i.e. it is possible to initialise a N96L85 model with a N48L60 start dump).

Ancillary and input data files → In file related options

The **Ancillary version files** is used to define two files which give a set of standard names and locations of input ancillary files, which are then used in the **Ancillary and input data files → Climatologies & potential climatologies** sections.

Ancillary and input data files → Climatologies & potential climatologies

This menu lists all the possible ancillary files that are read-in by the model. Key ones are the **sea surface temperatures** and **sea ice fields**, where the SSTs and Sea-Ice ancillaries are specified. At UM8.4 you may find that the directory and file name are both environment variables (e.g. `$UM_ANCIL_SST_DIR` and `$UM_ANCIL_SST_FILE`). These are set in the *Ancil versions* and *Ancil filenames* files which are set in the **Ancillary and input data files → In file related options → Ancillary version files** panel.

UKCA emissions are held in the **user multi-level ancillary file & fields** (for 3D emissions, e.g. aircraft NOx) and **user single-level ancillary file & fields** (for surface emissions, e.g. CO etc.). It is standard to have an update time of 5 days for these emissions (for monthly mean data).

Ancillary and input data files → Other ancillary files and Lateral Boundary files

In this menu you will find the **orography**, **land-sea-mask**, and **land fraction** file specifications.

STASH

STASH will be discussed in more detail in the what is STASH? tutorial.

The UKCA Panel

The UKCA panel can be found in two ways, either by going to **Atmosphere → Model Configuration → UKCA Chemistry and Aerosols** or **Atmosphere → Scientific Parameters and Sections → Section by section choices → Section 34: UKCA Chemistry and Aerosols**. This contains a number of follow-on windows which are used to configure the options for UKCA.

This tutorial will focus on chemistry only, and will not cover any aspects of the GLOMAP-mode (http://www.see.leeds.ac.uk/research/icas/atmospheric-chemistry-and-aerosols/other-links/aerosol-modelling/the-glomap-model/) aerosol scheme.

Main UKCA Panel

In this panel you can turn UKCA on and off, choose the solver to use, and which scheme to use using that solver (changing the solver choice will change the available schemes). The **Help** button on this page is quite informative. This panel is shown in Figure 3.

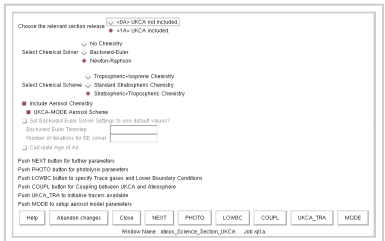


Figure 3: The main UKCA panel.

Main UKCA Panel → NEXT follow-on window

In this panel you can set some optional switches, which will be available depending on the scheme used. Again, the **Help** button is extensive. It is suggested for Stratospheric chemistry schemes (e.g. *CheS/Strat* and *CheST/StratTrop*) that the UM specific humidity field is always used as the UKCA H2O tracer. This panel is shown in Figure 4.

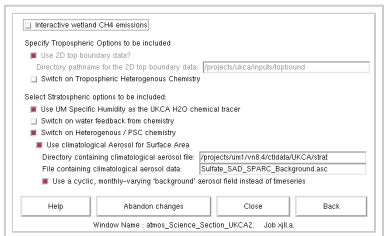


Figure 4: The first follow-on window from the main UKCA panel.

Main UKCA Panel → PHOTO follow-on window

This panel is used to define the photolysis scheme used, either using pure look-up tables, the (tropospheric) Fast-J scheme, or the (tropospheric and stratospheric) Fast-JX scheme. Within each scheme are possible other options, such as the location of required input files. The **Help** button has more information, and for *CheST/StratTrop* chemistry the options shown in Figure 5 are recommended.

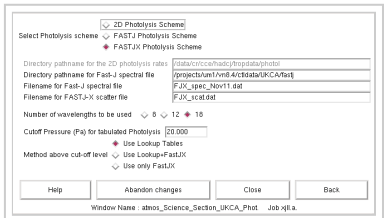


Figure 5: The UKCA photolysis follow-on panel.

Main UKCA Panel → LOWBC follow-on window

This panel is shown in Figure 6, and also has useful information under the **Help** button. For some UKCA schemes, such as the Stratospheric or Stratospheric + Tropospheric (*CheST/StratTrop*) chemistry, a lower-boundary condition is required for some of the long-lived species, principally the CFCs. The default in the code is to take these from the WMO A1b scenario specified for CMIP3 (and as was used in the CCMVal-2 model intercomparison). However, this scheme is tied in to the date during the run, changing as this changes. If instead a *timeslice* run is to be performed (i.e. although the date is changing, all the forcings remain the same) then these lower-boundary conditions need to be set, and this is done in this panel. Only those species that are not defined already are set here, as the others are taken from the **Scientific Parameters and Sections → Spec of trace gases** panels. Also, these values are also used by the chemical solver when a constant field is used, which is why H2 and N2 are also defined here if required.

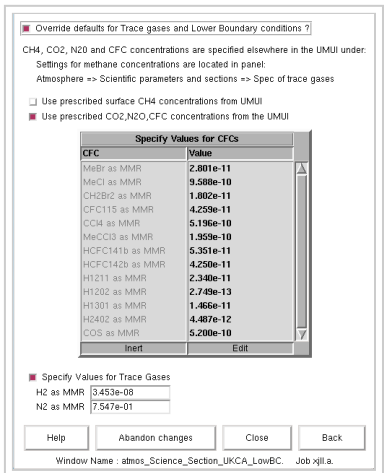


Figure 6: The UKCA lower-boundary condition panel.

If you need to get the values for these species, then the **scenario** program is available on PUMA at `/home/ukca/bin/scenario`. If you wanted the values for the 1st July 2000, you would run (and get the output) like this:

```
$ scenario 2000/07/01
-----------------------------------------------
| 2000/07/01  WMOA1 SCENARIO:                 |
-----------------------------------------------
| CFCl3       =   1.24350E-09     CFC11/F11   |
| CF2Cl2      =   2.25150E-09     CFC12/F12   |
| CF2ClCFCl2  =   5.31200E-10     CFC113/F113 |
| CF2ClCF2Cl  =   1.00400E-10     CFC114/F114 |
| CF2ClCF3    =   4.68200E-11     CFC115/F115 |
| CCl4        =   5.23050E-10                 |
| MeCCl3      =   2.10000E-10     CH3CCl3     |
| CHF2Cl      =   4.23850E-10     HCFC22      |
| MeCFCl2     =   5.12950E-11     HCFC141b    |
| MeCF2Cl     =   4.17500E-11     HCFC142b    |
| CF2ClBr     =   2.31150E-11     H1211       |
| CF2Br2      =   3.29800E-13     H1202       |
| CF3Br       =   1.40600E-11     H1301       |
| CF2BrCF2Br  =   3.68000E-12     H2402       |
| MeCl        =   9.58800E-10     CH3Cl       |
| MeBr        =   2.82700E-11     CH3Br       |
| CH2Br2      =   1.80186E-11                 |
| N2O         =   4.80605E-07                 |
| CH4         =   9.75050E-07                 |
| CF3CHF2     =   0.00000E+00     HFC125      |
| CH2FCF3     =   0.00000E+00     HFC134a     |
| H2          =   3.45280E-08                 |
| N2          =   7.54682E-01                 |
| CO2         =   5.62075E-04                 |
-----------------------------------------------
UM/UKCA LBC MMRs for: 2000/07/01, using the WMOA1 scenario
VALUES FOR USE IN THE UMUI (ZERO VALUES CAN BE TREATED AS "Excluded"):
 CH4         =  9.750E-07
 N2O         =  4.806E-07
 CFC11       =  1.244E-09
 CFC12       =  2.251E-09
 CFC113      =  5.312E-10
 HCFC22      =  4.238E-10
 HFC125      =   0.00
 HFC134a     =   0.00
 CO2         =  5.62075E-04
VALUES FOR USE IN THE UKCA HAND-EDIT:
 MeBrMMR=2.82700E-11,
 MeClMMR=9.58800E-10,
 CH2Br2MMR=1.80186E-11,
 H2MMR=3.45280E-08,
 N2MMR=0.75468    ,
```

```
CFC114MMR=1.00400E-10,
CFC115MMR=4.68200E-11,
CCl4MMR=5.23050E-10,
MeCCl3MMR=2.10000E-10,
HCFC141bMMR=5.12950E-11,
HCFC142bMMR=4.17500E-11,
H1211MMR=2.31150E-11,
H1202MMR=3.29800E-13,
H1301MMR=1.40600E-11,
H2402MMR=3.68000E-12,
```

The default is to use the WMO A1b, but you can also request any of the RCP scenarios for CMIP5, by running one of, e.g.

```
scenario 2000/07/01 RCP2.6
scenario 2000/07/01 RCP4.5
scenario 2000/07/01 RCP6.0
scenario 2000/07/01 RCP8.5
```

The WMO A1b scenario is only valid for the years 1950-2100 inclusive, whereas the RCP scenarios are valid from 1765-2500 inclusive. Note also that this script uses a 360-day calendar.

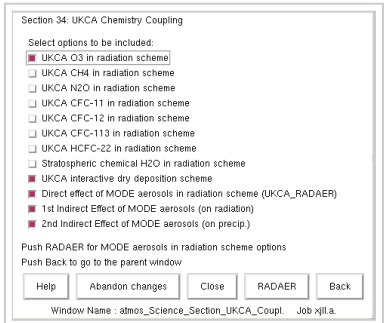


Figure 7: The UKCA coupling panel.

Main UKCA Panel → COUPL follow-on window

This panel is used to control which chemical species feed-back onto the Unified Model's radiation scheme, or if the dry deposition scheme is interactive with the land-surface, and is shown in Figure 7. If these trace gases feed-back, then, e.g. the N2O field from UKCA is used, rather than setting a constant value throughout the atmosphere. It is recommended that the interactive dry deposition scheme is always used.

Main UKCA Panel → UKCA_TRA follow-on window

This panel gives the listing of all the tracers which are turned on by default for the chemistry scheme chosen. If you are adding new tracers you are unable to change these values and must instead use a hand-edit. This will be covered in more detail in the adding new chemical tracers tutorial.

Written by Luke Abraham 2014

UKCA Chemistry and Aerosol Tutorial 3

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Contents

- 1 What you will learn in this tutorial
- 2 Task 3.1: add new output
- 3 What is STASH?
  - 3.1 Prognostic and Diagnostic Fields
  - 3.2 STASH Sections and Items
  - 3.3 User-STASHmaster files
- 4 Output Files
  - 4.1 Standard PP files
  - 4.2 Climate Mean files
- 5 The UMUI STASH Panel
  - 5.1 Time Profiles
  - 5.2 Domain Profiles
  - 5.3 Usage Profiles
  - 5.4 The Menu Bar
    - 5.4.1 STASH
    - 5.4.2 Profiles
    - 5.4.3 Diagnostics
      - 5.4.3.1 Load New Diagnostics (Control-l)
      - 5.4.3.2 Remove Diagnostics (Control-r)
      - 5.4.3.3 Output Table to File
      - 5.4.3.4 Set Package Switches (Control-t)
      - 5.4.3.5 Clear Table
      - 5.4.3.6 Verify Diagnostics (Control-v)
      - 5.4.3.7 Sort Diagnostics
      - 5.4.3.8 Change Sort Order
    - 5.4.4 Help
- 6 Solution to Task 3.1

What you will learn in this tutorial

In this tutorial you will learn how to edit your UKCA job to add some high-frequency output. You will also learn how to deal with any warnings that occur in the course adding new diagnostics.

Task 3.1: add new output

TASK 3.1: Output the instantaneous UKCA ozone field to the **pb/UPB** stream every 6 hours.

Hint [hide]
You will need to make the changes to the STASH table first, and remember to use **Verify Diagnostics** and correct any warnings generated for the **pb/UPB** stream (61).

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 Collaboration Twiki (password required) (http://collab.metoffice.gov.uk/twiki/pub/Support/Umdp) .

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 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.

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 512 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, which is located at

```
/work/n02/n02/hum/vn8.4/ctldata/STASHmaster/STASHmaster_A
```

on ARCHER, and at

```
/projects/um1/vn8.4/ctldata/STASHmaster/STASHmaster_A
```

on MONSooN. This is also a handy list of all the fields that can be outputted from the model, which is easier to search than by going through the UMUI panels. UKCA uses section 34 for chemical tracers and chemical diagnostics, and section 38 for aerosol diagnostics.

User-STASHmaster files

As well as the prognostic and diagnostic fields held in the *STASHmaster_A* file, it is possible to make your own **user STASHmaster file** which can either add new prognostics/diagnostics, or over-write existing prognostics/diagnostics. These are added to the UMUI in the **Model Selection → Atmosphere → STASH → User-STASHmaster files. Diags, Progs & Ancils** panel. These new fields may then need to be initialised in the **Model Selection → Atmosphere → STASH → Initialisation of User Prognostics** panel. More information on user STASHmaster files will be given in the adding new chemical tracers, adding new emissions, and adding new UKCA diagnostics tutorials.

Output Files

Before we cover outputting diagnostics from the UMUI STASH panel, we will first cover the different output fields files that are produced by the UM. Output from STASH is sent to different output streams, and there are two types of files used by these streams: **standard PP files** and **climate mean files**.

The basic building block of UM output files is a 2D latitude-longitude slice. Surface variables are made up of a single slice, whereas 3D variables are made up of many slices. If you output a 3D variable, such as ozone, this is in fact made up of 85 (as there are 85 model levels) slices.

Standard PP files

These files are controlled in **Model Selection → Post Processing → Initialization and processing of mean and standard PP files**. As well as choosing the packing profile (see the **Help** button for more information) this panel also describes how these different files are treated. As can be seen in Figure 1, there are 11 PP files defined, **PP0** to **PP10**, which correspond to designators **pa** to **pk** (with units 60-69, and 151 - these unit numbers may be important in warning and error messages).

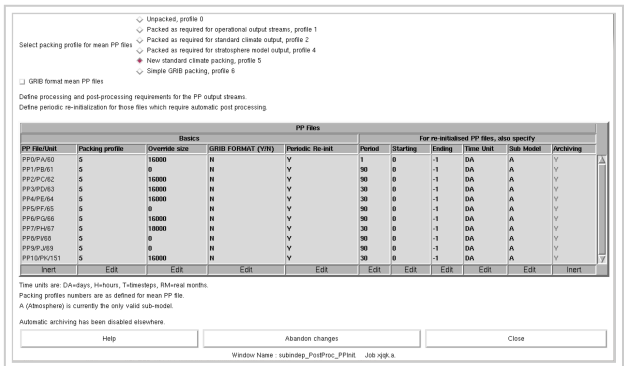


Figure 1: The initialization and processing of PP files UMUI panel.

The table in this panel can be used to set various properties of these files, such as the time period the file covers, and whether or not the file is **archived** (i.e. either to the `archive/` sub-directory or RDF facility on ARCHER, or two the `/nerc` disk or MOOSE on MONSooN). Being able to change the time period is very useful - while for many simulations monthly mean files are preferred (see the discussion about *climate meaning* below), often you may wish to output higher frequency data, e.g. 6-hourly. These PP files are usually limited to a finite size and so outputting a month of 6-hourly data may cause the model to crash at run-time. The solution to this is to change how frequently the files are generated, e.g. you could output daily files instead of monthly ones.

The last column in this table controls the archiving of these files. If you have set up your job to archive to a different directory (e.g. to the `archive/` sub-directory on ARCHER, or are sending data to the `/nerc` disk on MONSooN or to the RDF on ARCHER, then the files associated with each stream will only be moved if this column is set to **Y**. If it is **N** then the file will **not be archived but will be deleted** if you have requested the **Delete superseded PP files** in the **Model Selection → Post Processing → Main Switch + General Questions** panel.

If there is no data being sent to a stream (which is controlled in the STASH panel) then this may automatically be set to **N**. If this is the case then when you send data to a stream you will need to manually set it to **Y** after you have made the diagnostic request.

Climate Mean files

Climate meaning is controlled in the same panel where the restart dump frequency is set: **Model Selection → Atmosphere → Control → Post processing, Dumping & Meaning**. The main panel is shown in Figure 2. In this example, as is standard for all climate jobs, restart dumps are created every 10 days, and this example as these dumps archived every 9 files, starting with the 9th file after the job is started, i.e. every 3 months (to save space you may choose to change this to 1 year or every 36 files). All other dumps are deleted after they are no longer needed. This means that if the model crashes and needs to be restarted, you will at most have lost 10 days of run. **Note:** while the model will bit compare on restarting from an old dump, this will not be the case if the dump frequency is changed.

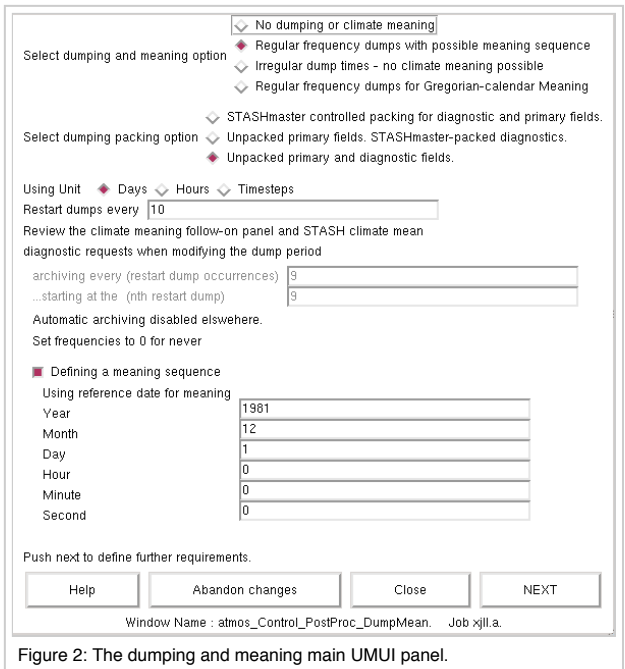


Figure 2: The dumping and meaning main UMUI panel.

This panel has selected **regular frequency dumps with possible meaning sequence** so that **climate meaning** is possible. To do this, you must specify a date for the meaning sequence. **Note:** you should be careful about this choice of date, especially if you are changing the start-date of the run. It is advised that this meaning sequence date is set to be on or before your start date. If it is not you will not generate some of the climate mean files.

The follow-on window from this panel (shown in Figure 3) gives more details about how climate meaning works. These files work on multiples of the dump period, so the first climate mean file created will be a multiple of 3 dump periods (i.e. 3x10 days, or 30 days. Since we are running a 360-day calendar, this is 1 model month) which is the **pm** monthly-mean file, which we considered when running existing UKCA Job. The second climate-mean file is a multiple of 3 of the monthly-mean files (i.e. 3x30 days, or 90 days, which is a season), the **ps** seasonal-mean file (the meaning date is important here to get the correct 3 months for each season). The third climate-mean file is a multiple of 4 of the seasonal-mean files (i.e. 3x90 days, or 360 days, a year), the **py** annual-mean file. The final file created is a multiple of 10 of the annual-mean files - a decadal-mean **px** file.

The fields outputted to each of these pm, ps, pa, or px file is the same, and in the same order. All that is different is the meaning done to each field. **Note:** if you change the dump frequency you will change the frequency of creation of these climate-mean files. In this instance you would need to change the multiple for the first file to get a monthly-mean file correctly.

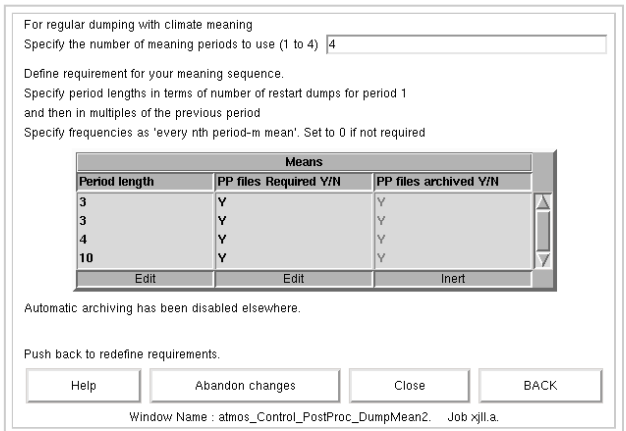


Figure 3: The dumping and meaning follow-on UMUI panel.

The UMUI STASH Panel

This panel can be found at **Model Selection → Atmosphere → STASH → STASH. Specification of Diagnostic requirements** and is shown in Figure 4. When you first open this panel you may get a window telling you that system diagnostics have been over-written by user diagnostics - just click **continue** here. We will cover user diagnostics in the adding new UKCA diagnostics tutorial.

This panel is organised as a table, listing the diagnostics that have been selected (currently - more can be added) and what the **time**, **domain**, and **usage** profile for each diagnostic is. In the table as well as giving the name of each diagnostic, the STASH section and item number is also given, with the table sorted by section, item.

Note: The UMUI STASH panel only reads the user STASHmaster files once, when it is initially started up. If you have opened this panel and then add in a new user STASHmaster file (which may over-write the name of an existing prognostic or diagnostic) then the changes it makes may not be shown in the STASH panel. For these changes to be seen you should **save** your job, **close** it, and then re-open it. This will then force the STASH panel to read (and reflect the changes in) the new user STASHmaster file.

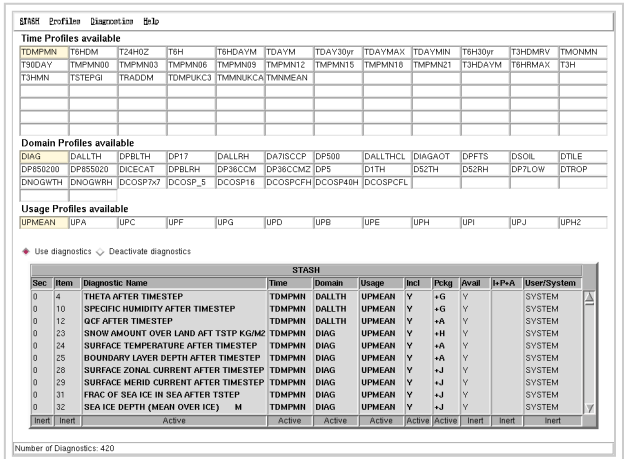


Figure 4: The UMUI STASH panel.

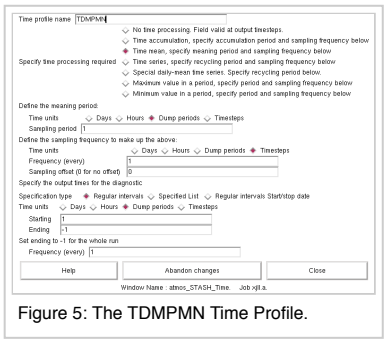


Figure 5: The TDMPMN Time Profile.

Time Profiles

There is a rather extensive list of time profiles already existing in this job, and it is possible to add more by making a new one in a blank slot. These time profiles define how the data is time-processed prior to being output. Some profiles, such as **T6H** have no meaning, and just output the field as it is every 6 hours, other profiles, such as **TDAYM** will result in a daily mean of a field. You can view how these profiles are defined by selecting a profile (it will highlight yellow) and going to **Profiles → Edit Profile → Edit time** in the STASH panel menu.

An example of the **TDMPMN** time profile is shown in Figure 5. This is the climate-meaning time profile, and samples fields every dump period, and should only be used with the **UPMEAN** usage profile (see below).

Domain Profiles

These profiles cover the horizontal and vertical domain covered by fields as they are output. Depending on the diagnostic which is outputted, some of these are on pressure levels, model theta levels, model rho levels, or on a single (e.g. a surface of top of the atmosphere) level etc.. The **DALLTH** outputs variables on all model theta levels, and it is this domain which UKCA fields (except for surface fields) are usually outputted on. If you edit this profile you can also see that it is possible to output zonal means of fields, or to limit the output to a particular area. This last option can be especially useful if a large amount (e.g. hourly) data is required, but only for a specific part of the world, over the length of a long run.

Usage Profiles

These profiles determine which output stream to use. The **UPMEAN** profile will send data to the climate-meaning stream discussed above, whereas the **UPA** profile will send data to the **pa** file etc.

Note: If you are sending data to the climate-meaning stream **UPMEAN** you **must** use the **TDMPMN** time profile (or a variation thereof), otherwise this output will become corrupted or incorrectly meaned.

The Menu Bar

The STASH panel is unusual for the UMUI in that it has a separate menu bar, rather than using a row of buttons along the bottom of the panel.

STASH

This menu contains the usual **Close** (i.e. save and close panel) and **Abandon** (i.e. close without saving) options.

Profiles

As discussed above, you can use the **Edit Profile** options to view (and change) new or existing profiles. You can also **Copy** or **Delete** profiles from this menu.

Diagnostics

This menu contains the items that you will probably use the most. It is here that you can add new diagnostics and change the order of the diagnostics in the table.

Load New Diagnostics (Control-l)

Clicking on this option will bring up a new window, organised by section, listing all the items that can be outputted. This table will list if the field is available in the current model configuration, and there may also be **Help** available for some diagnostics (if there is, double click where it says *"Help"*).

Remove Diagnostics (Control-r)

You can use this option to remove a diagnostic from the table, although it is easier to type **Control-r**.

Output Table to File

This outputs the current STASH table, in its current order, to a file (called jobid.A.diags) in your `$HOME/umui_jobs` directory on PUMA. If you wish to compare the STASH between two jobs it is best to output the STASH table from each and use xxdiff on PUMA, as comparing the two jobs using the UMUI diff is rather confusing when it comes to STASH.

Set Package Switches (Control-t)

You will notice that for some diagnostics in the 8th column of the STASH table (labelled *Pckg*) there is a letter, e.g. J or +G etc. This corresponds to a package, which is set in this table. This is a useful way of organising diagnostics so that they can be easily turned on or off. For instance, if package J is off then the letter just appears as **J** (and the *I+P+A* column would say *N*), however, this package can be turned on in the package switches table (set to **Y**), and if it is on this letter now appears as **+J**. You can only turn diagnostics which are organised through a package on by turning that package on (although you can remove the diagnostic from the package, or add it again but not in package).

Clear Table

Warning: clicking this option will remove all the diagnostics in your STASH table. If you do this in error go to **STASH → Abandon** to close the STASH window without saving.

Verify Diagnostics (Control-v)

This is a very useful option. When you add new diagnostics you may inadvertently have made some errors. For some diagnostics you may not be sure what levels (i.e. *domain profile*) it should be outputted on. For example, UKCA tracer fields should be outputted on model theta levels (*DALLTH*). If this field is requested on model rho levels (*DALLRH*) then the **Verify Diagnostics** window would give

```
Diag: "O3 MASS MIXING RATIO AFTER TIMESTEP " (34,1) (TDMPMN,DALLRH,UPMEAN)
 DOMAIN PROF ERROR: Use profile on model theta-levels.
```

Sort Diagnostics

Use this option to order the diagnostics by section, item. When adding new diagnostics they are usually added to the top of the list, rather than in order.

Change Sort Order

This will bring up a box where you can choose which columns are considered (and in which order) when it comes to sorting the diagnostics. The default ordering is equivalent to **1 2** i.e. sort by column 1 (section) then by column 2 (item).

Help

The **Help** menu has a more detailed description of the features and options in the STASH panel.

Solution to Task 3.1

Please see this page for a solution to Task 3.1.

Written by Luke Abraham 2014

Solution to UKCA Chemistry and Aerosol Tutorial 3 Task 3.1

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Back to the what is STASH? tutorial

Contents

- 1 Task
- 2 Solution
  - 2.1 Add the new diagnostic to the STASH table
  - 2.2 Verify the Diagnostics
  - 2.3 Fix
    - 2.3.1 Method 1: Increase the Override size
    - 2.3.2 Method 2: Change the output frequency
- 3 Output
- 4 Worked Solution

Task

You were asked to

> Output the instantaneous UKCA ozone field to the **pb/UPB** stream every 6 hours.

and were given the hint

> You will need to make the changes to the STASH table first, and remember to use **Verify Diagnostics** and correct any warnings generated for the **pb/UPB** stream (61).

Solution

Add the new diagnostic to the STASH table

Go to **Model Selection → Atmosphere → STASH → STASH. Specification of Diagnostic requirements**.

You will need to add the following to the STASH table

```
34   1 O3 MASS MIXING RATIO AFTER TIMESTEP T6H      DALLTH   UPB      Y +  Y
```

i.e.

- **STASH section** = 34
- **STASH item** = 1
- **Time Profile** = T6H
- **Domain Profile** = DALLTH
- **Usage Profile** = UPB

Verify the Diagnostics

Verifing the diagnostics (either *Control-v* or *Diagnostics → Verify Diagnostics*) gives

```
PP-Field Count Estimates


Warning!!! You may exceed the maximum number of PP fields per file

Estimated number of PP files to be written:

30600 fields in stream 61.  Maximum allowed is 4096.
22835 fields in Climate mean Period_1
22835 fields in Climate mean Period_2
22835 fields in Climate mean Period_3
22835 fields in Climate mean Period_4

Maximum allowed is 4096 fields per Climate Mean Period.
```

You don't need to worry about the Climate Mean Period here, and in fact this job will run perfectly happily with this many fields. The line that is of more concern is

```
30600 fields in stream 61.  Maximum allowed is 4096.
```

The value of 30600 comes from the fact that you are requesting 4x85 level variables for 90 days: 4x90x85=30600.

Fix

There are two ways to fix this. They both need you to go to **Model Selection → Post Processing → Initialization and processing of mean and standard PP files**.

Method 1: Increase the Override size

Increasing the value in the **Override size** column from 0 to 16000 for the **PP1/PB/61** stream will remove this error. However, the **Help** text explicitly states here

> Override size.
> Fieldsfiles usually can only contain 4096 fields. The recommended way to work around this restriction is to use periodic re-initialisation of the fieldsfiles. In the rare case that this is not a suitable option (e.g. for analysis time in a forecast) it is possible to extend the 4096 field restriction by overriding the default size. Be careful not to override the size by too much - large numbers of fieldsfile headers can be inefficient for both runtime and memory use and may cause problems with some small executables.

So the UMUI is explicitly suggesting that the frequency of output should be altered. This is Method 2.

Method 2: Change the output frequency

In this method we change the **Period** from 90 to 1, i.e. there will be a new file created every day.

We will use *Method 2*.

Output

Now save, process, and submit your job. When the job has run the output in your archive directory will now contain 4 output files (*.p*):

```
$ ls *.p*
xjllba.pa20051201  xjllba.pb20051201
```

Open the **pb** file. The variable list should look like

```
0    : 192   145   85    4     Stash code = 34001
```

Select this field and click the **t** radio button, which will list the times of each point in the field. This should give

```
2005/12/01:06.00 / 0.250000
2005/12/01:12.00 / 0.500000
2005/12/01:18.00 / 0.750000
2005/12/02:00.00 / 1.000000
```

Sample output from this job can be found in

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Task3.1/
```

on ARCHER, and in

```
~~/projects/ukca/Tutorial/vn8.4/sample_ouput/Task3.1/~~
```

on MONSooN.

Worked Solution

There is a worked solution to this problem in the UMUI Tutorial experiment. This is job **b**: *Tutorial: solution to Task 3.1 - outputting a diagnostic*.

Written by Luke Abraham 2014

UKCA Chemistry and Aerosol Tutorial 4

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Contents

- 1 What you will do in this tutorial
- 2 Task 4.1: Make slots for two new tracers
- 3 Adding new chemical tracers
- 4 Part 1. Adding a tracer into the UMUI
  - 4.1 Pick a free UKCA tracer slot
  - 4.2 Edit your user STASHmaster file
  - 4.3 Initialise your tracer(s)
  - 4.4 Tell the UMUI about your new tracer
- 5 Solution to Task 4.1: Make slots for two new tracers
- 6 Task 4.2: Add these two new tracers to UKCA
- 7 Part 2. Adding new a new chemical tracer to UKCA
  - 7.1 FCM branches and merging
  - 7.2 Reviewing your code changes and committing
  - 7.3 Editing the advected tracer list
  - 7.4 Increase the number of tracers to be used
  - 7.5 Editing the Chemistry Scheme Specification
  - 7.6 Define conversion factors for your new tracers
  - 7.7 Tell UKCA to use these conversion factors
  - 7.8 Increase the size of JPCTR and JPSPEC
- 8 Numerical Noise
- 9 Solution to Task 4.2: Add these two new tracers to UKCA

What you will do in this tutorial

In this tutorial you will learn how to make the required changes in the UMUI and in UKCA to include new chemical tracers (although the UMUI steps also count when adding new aerosol or passive tracers). As you learn how to do this you will add two new tracers to the UKCA tutorial job you have already been running.

You will first learn how to make new tracers in the UMUI, before learning how to then use these tracers in UKCA.

Task 4.1: Make slots for two new tracers

TASK4.1: Add in two new tracers in to slots 64 and 65. The tracer in slot 64 will be called **ALICE** and the tracer in slot 65 will be called **BOB**. You should also output these two tracers through the **pa/UPA** stream in STASH as daily means.

Note: If you were unable to successfully complete Task 3.1, then please take a copy of the **b** job from the Tutorial experiment (*Tutorial: solution to Task 3.1 - outputting a diagnostic*) and work from there, as this will allow you to only make the changes required for this task.

Adding new chemical tracers

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. While you may think that you should start this by editing the UKCA code, in fact you should first adapt the UM and the UMUI so that it knows that these new tracers exist. Once you have created them you can then edit the UKCA code to use them to transport chemical species around.

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_tr as the source of your branch.

For example, first you should make a ticket on the UM Trac pages (https://puma.nerc.ac.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_tr@vn8.4
```

before checking-out your branch by

```
fcm checkout fcm:um_br/dev/userid/vn8.4_your_branch_name
```

More information on FCM can be found at the NCAS-CMS FCM pages (http://cms.ncas.ac.uk/wiki/Fcm) .

During this tutorial you will be tasked to add in two new tracers into your UM branch.

Part 1. Adding a tracer into the UMUI

Pick a free UKCA tracer slot

The UM has **150** tracers available for use by UKCA in STASH section 34. The first 100 of these are chemical tracers, which may (or may not) be in use by the various schemes that are currently available. The last 101-150 are reserved for aerosol and diagnostic tracers. You should therefore ensure that the tracers that you add are in one of the 1-100 slots, although many of these slots will already be in use by the scheme that you are using.

To choose the correct slot(s), you should first examine the **nm_spec** array which is in the **ukca_setd1defs.F90** routine.

To do this, cd into

```
vn8.4_your_branch/src/atmosphere/UKCA
```

and view the ukca_setd1defs.F90 file. This will give the following for the **nm_spec** array which is found around line 370:

```
      IF (L_UKCA_RAQ) THEN
          ! This list of tracers is valid for the RAQ chemistry.
          ! If MODE aerosols are used with it but their positions
          ! change in the array then the list needs to be updated.
          nm_spec(1:n_all_tracers) = (/                                     &
         'O3        ','NO        ','NO3       ','NO2       ','N2O5      ',  &
         'HO2NO2    ','HONO2     ','H2O2      ','CH4       ','CO        ',  &
         'HCHO      ','MeOOH     ','HONO      ','C2H6      ','ETOOH     ',  &
         'MeCHO     ','PAN       ','C3H8      ','N-PrOOH   ','I-PrOOH   ',  &
         'EtCHO     ','Me2CO     ','MeCOCH2OOH','PPAN      ','MeONO2    ',  &
         'O3S       ','C5H8      ','ISOOH     ','ISON      ','MACR      ',  &
         'MACROOH   ','MPAN      ','HACET     ','MGLY      ','NALD      ',  &
         'HCOOH     ','MeCO3H    ','MeCO2H    ','MVK       ','MVKOOH    ',  &
         'Cl        ','ClO       ','Cl2O2     ','OClO      ','Br        ',  &
         'BrO       ','BrCl      ','BrONO2    ','N2O       ','HCl       ',  &
         'HOCl      ','HBr       ','HOBr      ','ClONO2    ','CFCl3     ',  &
         'CF2Cl2    ','MeBr      ','N         ','O(3P)     ','ORGNIT    ',  &
         'MeCl      ','CF2ClBr   ','CCl4      ','CF2ClCFCl2','CHF2Cl    ',  &
         'MeCCl3    ','CF3Br     ','H2OS      ','CH3OH     ','H2        ',  &
         'SO2       ','H2SO4     ','DMS       ','MSA       ','DMSO      ',  &
         'NH3       ','CS2       ','COS       ','H2S       ','H         ',  &
         'OH        ','HO2       ','MeOO      ','EtOO      ','MeCO3     ',  &
         'n-PrOO    ','i-PrOO    ','EtCO3     ','MeCOCH2OO ','RNC2H4    ',  &
         'RNC3H6    ','C2H4      ','C3H6      ','C4H10     ','C4H9OOH   ',  &
         'MEK       ','TOLUENE   ','MEMALD    ','GLYOXAL   ','oXYLENE   ',  &
         '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',  &
         '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',  &
         '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',  &
         '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       ',  &
         'Dust_Div_1','Dust_Div_2','Dust_Div_3','Dust_Div_4','Dust_Div_5',  &
         'Dust_Div_6','Rn-222    ','Pb-210    ','XXX       ','XXX       '   &
          /)
      ELSE
!  Tracers 98,99 & 100 are for lumped Nitrogen, Br and Cl for stratospheric chemistry,
!   but can only be renamed in STASHmaster file not in advt or nm_spec.
          nm_spec(1:n_all_tracers) = (/                                     &
         '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
          /)
      END IF
```

```
! Mode components: SU: sulphate, BC: black carbon, OC: organic carbon
!                  SS: sea-salt, Du: dust,         SO: organic carbon 2
!                  NH: ammonium, NT: nitrate,      ND: number density
```

The settings in the UMUI UKCA panel will tell you whether you are using the RAQ chemistry or not (in this tutorial you are not). If you look through the **ukca_chem_scheme.F90** file (e.g. *ukca_chem_strattrop.F90*) then you will see a list of the species that you are using.

We can relate this back to the **nm_spec** array above. The following section of that array gives the *CheST/StratTrop* chemical species in black. Red species should be avoided as (in general working) they may cause a clash some current settings. Species in green are not currently used by any scheme and so can be over-written (although they may be used in future).

```
        '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
```

You should pick one of the green slots (although if you are adding more than 13 new tracers and you know that you will not need any of the red ones, these can also be used. In the future the number of UKCA tracers will be extended.

In this example we will over-write tracer number **65** (CHF2Cl), although any of the green tracers would do just as well.

You should not make any code changes to ukca_setd1defs.F90 at this time. If you do so you will have problems performing the `fcm merge` step later in this tutorial.

Edit your user STASHmaster file

In your UMUI job go to **Model Selection → Atmosphere → STASH → User-STASHmaster files. Diags, Progs & Ancills.** This will contain a listing of the full path to a file that describes the UKCA tracers that you are using (in this tutorial it is called *UKCA_Tr_StratTropAeroMODE.prestash*). Take a copy (using `cp`) of this file to your local directory on PUMA. If you open this file you will see that each tracer has a listing which looks like

```
#
1|     1 |    34 |     1 |O3 MASS MIXING RATIO AFTER TIMESTEP |
2|     2 |     0 |     1 |     1 |     2 |    10 |    11 |     0 |     0 |     0 |     0 |
3| 00000000000000000000000000001 | 00000000000000000001 |     1 |
4|     1 |     0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
5|     0 | 1861 |     0 |    65 |     0 |     0 |     0 |     0 |     0 |
#
```

Each of the areas coloured in red above is unique to each tracer, i.e. the name of the tracer and the number of the tracer. Since we are over-writing tracer 65, you would make a new entry in the STASHmaster file which reads

```
#
1|     1 |    34 |    65 |MY TRACERS NAME                      |
2|     2 |     0 |     1 |     1 |     2 |    10 |    11 |     0 |     0 |     0 |     0 |
3| 00000000000000000000000000065 | 00000000000000000001 |     1 |
4|     1 |     0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
5|     0 | 1861 |     0 |    65 |     0 |     0 |     0 |     0 |     0 |
#
```

Note: the spacing of the pipes (|) in this file is important, so be careful not to change this.

Now, in the **Model Selection → Atmosphere → STASH → User-STASHmaster files. Diags, Progs & Ancills.** panel replace the original user STASHmaster file with your new one containing the additional tracer(s).

Initialise your tracer(s)

Now you need to initialise your tracer. By placing it in a STASHmaster file which the UM reads, as a tracer is a *prognostic* field, it will need to be initialised to a value in the model start dump (which is done by the reconfiguration step). To set this go to **Model Selection → Atmosphere → STASH → Initialisation of User Prognostics**. This will open a panel containing a table listing all the STASH items listed in all your STASHmaster files. Scroll down this list until you get to your new tracer. While the STASH number and name of the tracer (which you set in your STASHmaster file) is present, all other columns will be blank.

The simplest options are either to set the **Option** column to **3** (which sets the field to zero) or to **6** and then give a value to set the field to in the **If 6: CONSTANT** column. In either case this will set every single gridcell to be the same value. You can, if needed, make up an ancillary file containing an initial condition (option **7**) if you required there to be spatial variation in your field (e.g. the ukca ozone field must always be initialised to an approximate initial condition in this way). This can be done using Xancil (http://cms.ncas.ac.uk/documents/xancil/) . Xancil will be covered in more detail in the adding new emissions tutorial. If these fields existed in the start dump you are providing then you could set this option to **1**.

Tell the UMUI about your new tracer

Unfortunately you cannot edit the UMUI directly to add-in your new tracer(s), and instead you must use a **hand-edit**. This is a script which run when you click the **Process** button, and what it does is to directly make changes to one or more of the scripts that is held in your `$HOME/umui_jobs/jobid` directory (you can think of the UMUI as making *indirect* changes), and it is these scripts which are copied across to the supercomputer when you click **Submit**.

If you go to your `$HOME/umui_jobs/jobid` directory and view the SIZES script, you will see that it contains an entry similar to, e.g.

```
TC_UKCA=1,1,1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,0,1,1,1,0,1,1,1,1,1,1,1,1,1,1,0,0,0,
0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,0,0,0,1,1,1,1,1,
1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,0,1,1,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,1,1,
```

TC_UKCA is a list of which tracers are on or not (1 being on, 0 being off). If, in this example, you were to go through the TC_UKCA list and add-up all the 1s you would find that they add-up to 104, and that you would only have a 1 where you have a corresponding number in the user STASHmaster file. Of the 104 UKCA tracers in use, two of them (numbers 149 and 150) are diagnostic tracers and are not considered by the chemistry scheme itself. Also, the H2O field used by UKCA comes from the UM's specific humidity field and so is not set in this list. (**Note:** in older UM versions there is also the **TR_UKCA** variable, which is the number of UKCA tracers.)

You will need to make up a new list. However, rather than you having to put the 1 in the correct place by hand (not easy to do!), there is the **make_tracer_list** script on PUMA which is located in

```
/home/ukca/bin/make_tracer_list
```

This script is run on a STASHmaster file. If this script is run on the edited user STASHmaster file (which has tracer 65 included) the output is

```
$ make_tracer_list UKCA_Tr_StratTropMODE.prestash
TC_UKCA=1,1,1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,0,1,1,1,0,1,1,1,1,1,1,1,1,1,1,0,0,0,
0,1,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,0,0,0,1,1,1,1,1,
1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,0,1,1,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,1,1,
```

(with the change from before highlighted in blue). You will now need to add this data into the **SIZES** script using a hand-edit.

Hand-edits use the ed (http://linux.die.net/man/1/ed) text editor, which can be run in a batch mode. We will need to make a script (e.g. *add_UKCA_Tr_StratTropAeroMODE.ed*) which contains the following

```
ed SIZES<<\EOF
/TC_UKCA=/
d
d
d
d
d
i
TC_UKCA=1,1,1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,0,1,1,1,0,1,1,1,1,1,1,1,1,1,1,0,0,0,
0,1,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,0,0,0,1,1,1,1,1,
1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,0,1,1,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,1,1,
.
w
q
EOF
```

This tells *ed* to search for the line `TC_UKCA=` and the delete this line and the next 4 lines, before inserting the text (i.e. this replaces the block in SIZES with the text here).

You will need to make this file executable so that the UM can run it. To do this, in the directory where the file exists do

```
chmod a+rx add_UKCA_Tr_StratTropAeroMODE.ed
```

Now, in the UMUI we go to **Model Selection → Input/Output Control and Resource → User hand edit files**. Put the full path to this file at the end of the list in the table and put a **Y** in the **Include Y/N** column.

Solution to Task 4.1: Make slots for two new tracers

Please see this page for a solution to Task 4.1

Task 4.2: Add these two new tracers to UKCA

TASK4.2: Make the required code changes so that your **ALICE** and **BOB** tracers are now specified in the UKCA CheST/StratTrop scheme. You should set the conversion factor for each of these to 1.0.

Note: If you were unable to successfully complete Task 4.1 above, then please take a copy of the **c** job from the Tutorial experiment (*Tutorial: solution to Task 4.1 - adding new chemical tracers to the UMUI*) and work from there, as this will allow you to only make the changes required for this task.

Part 2. Adding new a new chemical tracer to UKCA

Now that you have created the new tracer(s) in the UM/UMUI, you can then tell UKCA to use this tracer slot to transport round a chemical species. To do this you will need to edit the UKCA source code.

FCM branches and merging

Before we can do this we need to check and see if there are any possible conflicts that may arise by making these changes. If you go to the FCM Atmosphere panel at **Model Selection → FCM Configuration → FCM Options for Atmosphere and Reconfiguration** you can view the current branches used by this job, e.g.:

```
fcm:um_br/dev/jwalton/vn8.4_fix_endgame_grib_cloud_UKMO/src  12122  Y
fcm:um_br/dev/jwalton/vn8.4_iceberg_calving_UKMO/src         12116  Y
fcm:um_br/dev/jwalton/vn8.4_reinstate_ISCCP_UKMO/src         12111  Y
fcm:um_br/dev/luke/vn8.4_UKCA_Tutorials/src                  14638  Y
fcm:um_br/pkg/Config/vn8.4_ncas/src                          12550  Y
```

Most of these branches will not clash with our required changes. However, the **fcm:um_br/dev/luke/vn8.4_UKCA_Tutorials/src** (at revision 14638) branch will, as it directly changes parts of the chemistry scheme routines that we want to use. Before we can make our code changes we must first make a copy of this branch. While ideally you should be able to make a new branch and edit the source-code as is, often you need to make changes to an existing branch. The recommended way to do this is to make a new branch and then merge in the clashing branch with your own, then stop using the clashing branch in the UMUI.

You should already have made a new branch that you used to find the free tracer slots. Now `cd` into the highest-level directory of the branch, which contains

```
CodeOwners.txt  COPYRIGHT.txt  src/
```

From here you will need to merge in the **vn8.4_UKCA_Tutorials** branch. Do

```
fcm merge fcm:um_br/dev/luke/vn8.4_UKCA_Tutorials
```

You will then be asked to confirm the revision number (the latest, 14638, is the one used in the UMUI). Confirm by pressing **return**. Now you will be asked if you wish to go ahead with the merge. Confirm this by typing **y** and pressing **return**. This will now merge in all the changes from the UM trunk that the *vn8.4_UKCA_Tutorials* contains into your branch.

In the terminal, this will have the following output:

```
$ fcm merge fcm:um_br/dev/luke/vn8.4_UKCA_Tutorials
Eligible merge(s) from /UM/branches/dev/luke/vn8.4_UKCA_Tutorials@14672: 14638 14260 14259 14258 14257 14256 142
Enter a revision (or just press <return> for "14638"):
------------------------------------------------------------------------
Merge: /UM/branches/dev/luke/vn8.4_UKCA_Tutorials@14638
 c.f.: /UM/trunk@11592
------------------------------------------------------------------------dry-run
--- Merging r11593 through r14638 into '.':
U    src/control/top_level/atmos_physics1.F90
U    src/control/top_level/atm_step_4A.F90
U    src/control/top_level/atm_step.F90
U    src/control/top_level/um_input_control_mod.F90
U    src/include/common/version.h
U    src/include/common/clfhist.h
U    src/include/constant/chsunits.h
U    src/include/data/cenvirdt.h
U    src/script/control/qsatmos
U    src/atmosphere/UKCA/ukca_setd1defs.F90
U    src/atmosphere/UKCA/ukca_calcnucrate.F90
U    src/atmosphere/UKCA/asad_flux_dat.F90
U    src/atmosphere/UKCA/asad_bimol.F90
U    src/atmosphere/UKCA/asad_trimol.F90
U    src/atmosphere/UKCA/ukca_fastjx.F90
U    src/atmosphere/UKCA/ukca_radaer_struct_mod.F90
U    src/atmosphere/UKCA/ukca_main1-ukca_main1.F90
U    src/atmosphere/UKCA/ukca_radaer_compute_aod.F90
U    src/atmosphere/UKCA/ukca_aero_step.F90
U    src/atmosphere/UKCA/fastjx_specs.F90
U    src/atmosphere/UKCA/ukca_mode_setup.F90
U    src/atmosphere/UKCA/ukca_abdulrazzak_ghan.F90
U    src/atmosphere/UKCA/ukca_surfddr.F90
A    src/atmosphere/UKCA/ukca_scavenging_mod.F90
U    src/atmosphere/UKCA/ukca_coagwithnucl.F90
```

```
U    src/atmosphere/UKCA/ukca_d1_defs.F90
U    src/atmosphere/UKCA/asad_chem_flux_diags.F90
U    src/atmosphere/UKCA/ukca_rainout.F90
U    src/atmosphere/UKCA/ukca_mode_ems_um_mod.F90
U    src/atmosphere/UKCA/ukca_emission_ctl.F90
U    src/atmosphere/UKCA/ukca_activate.F90
U    src/atmosphere/UKCA/ukca_calc_drydiam.F90
U    src/atmosphere/UKCA/ukca_volume_mode.F90
U    src/atmosphere/UKCA/ukca_radaer_band_average.F90
U    src/atmosphere/UKCA/ukca_radaer_init-ukca1.F90
U    src/atmosphere/UKCA/asad_spimpmjp.F90
U    src/atmosphere/UKCA/ukca_aero_ctl.F90
U    src/atmosphere/UKCA/ukca_chem_strattrop.F90
U    src/atmosphere/UKCA/ukca_light.F90
U     src/atmosphere/UKCA/ukca_chem_tropisop.F90
U    src/atmosphere/UKCA/ukca_radaer_lut_mod.F90
U    src/atmosphere/UKCA/ukca_radaer_read_precalc.F90
U    src/atmosphere/UKCA/ukca_radaer_precalc_mod.F90
U    src/atmosphere/UKCA/ukca_setup_indices.F90
U    src/atmosphere/UKCA/ukca_radaer_read_luts.F90
U    src/atmosphere/UKCA/ukca_radaer_set_aerosol_field.F90
U    src/atmosphere/UKCA/ukca_init.F90
U    src/atmosphere/UKCA/ukca_option_mod.F90
A    src/atmosphere/UKCA/ukca_mode_check_artefacts_mod.F90
U    src/atmosphere/convection/convec2_mod-6a.F90
U    src/atmosphere/convection/glue_conv-gconv5a.F90
U    src/atmosphere/convection/ni_conv_ctl.F90
U    src/atmosphere/convection/convec2-conv24a.F90
U    src/atmosphere/radiation_control/rad_input_mod.F90
U    src/atmosphere/radiation_control/r2_lwrad3z.F90
U    src/atmosphere/radiation_control/r2_re_mrf_umist.F90
U    src/atmosphere/radiation_control/swrdiag_mod.F90
U    src/atmosphere/radiation_control/set_swdiag_logic.F90
U    src/atmosphere/radiation_control/diagnostics_lw.F90
U    src/atmosphere/radiation_control/init_swdiag_logic.F90
U    src/atmosphere/radiation_control/allocate_sw_diag.F90
U    src/atmosphere/radiation_control/r2_cloud_level_diag.F90
U    src/atmosphere/radiation_control/deallocate_swdiag.F90
U    src/atmosphere/radiation_control/deallocate_lwdiag.F90
U    src/atmosphere/radiation_control/lwrdiag_mod.F90
U    src/atmosphere/radiation_control/r2_swrad3z.F90
U    src/atmosphere/radiation_control/allocate_lw_diag.F90
U    src/atmosphere/radiation_control/fill_missing_data_sw.F90
U    src/atmosphere/radiation_control/r2_set_cloud_field.F90
U    src/atmosphere/radiation_control/set_lwdiag_logic.F90
U    src/atmosphere/radiation_control/fill_missing_data_lw.F90
U    src/atmosphere/radiation_control/diagnostics_sw.F90
U    src/atmosphere/radiance_core/radiance_calc.F90
------------------------------------------------------------------------dry-run
Would you like to go ahead with the merge?
Enter "y" or "n" (or just press <return> for "n"): y
Merge succeeded.
```

Now if you type **fcm status** you can see which routines have been changed:

```
$ fcm status
M       src/control/top_level/atmos_physics1.F90
M       src/control/top_level/atm_step_4A.F90
M       src/control/top_level/atm_step.F90
M       src/control/top_level/um_input_control_mod.F90
M       src/include/common/version.h
M       src/include/common/clfhist.h
M       src/include/constant/chsunits.h
M       src/include/data/cenvirdt.h
M       src/script/control/qsatmos
M       src/atmosphere/UKCA/ukca_setd1defs.F90
M       src/atmosphere/UKCA/ukca_calcnucrate.F90
M       src/atmosphere/UKCA/asad_flux_dat.F90
M       src/atmosphere/UKCA/asad_bimol.F90
M       src/atmosphere/UKCA/asad_trimol.F90
M       src/atmosphere/UKCA/ukca_fastjx.F90
M       src/atmosphere/UKCA/ukca_radaer_struct_mod.F90
M       src/atmosphere/UKCA/ukca_main1-ukca_main1.F90
M       src/atmosphere/UKCA/ukca_radaer_compute_aod.F90
M       src/atmosphere/UKCA/ukca_aero_step.F90
M       src/atmosphere/UKCA/fastjx_specs.F90
M       src/atmosphere/UKCA/ukca_mode_setup.F90
M       src/atmosphere/UKCA/ukca_abdulrazzak_ghan.F90
```

```
M       src/atmosphere/UKCA/ukca_surfddr.F90
A  +    src/atmosphere/UKCA/ukca_scavenging_mod.F90
M       src/atmosphere/UKCA/ukca_coagwithnucl.F90
M       src/atmosphere/UKCA/ukca_d1_defs.F90
M       src/atmosphere/UKCA/asad_chem_flux_diags.F90
M       src/atmosphere/UKCA/ukca_rainout.F90
M       src/atmosphere/UKCA/ukca_emission_ctl.F90
M       src/atmosphere/UKCA/ukca_mode_ems_um_mod.F90
M       src/atmosphere/UKCA/ukca_activate.F90
M       src/atmosphere/UKCA/ukca_calc_drydiam.F90
M       src/atmosphere/UKCA/ukca_volume_mode.F90
M       src/atmosphere/UKCA/ukca_radaer_band_average.F90
M       src/atmosphere/UKCA/ukca_radaer_init-ukca1.F90
M       src/atmosphere/UKCA/asad_spimpmjp.F90
M       src/atmosphere/UKCA/ukca_aero_ctl.F90
M       src/atmosphere/UKCA/ukca_chem_strattrop.F90
M       src/atmosphere/UKCA/ukca_light.F90
M       src/atmosphere/UKCA/ukca_chem_tropisop.F90
M       src/atmosphere/UKCA/ukca_radaer_lut_mod.F90
M       src/atmosphere/UKCA/ukca_radaer_read_precalc.F90
M       src/atmosphere/UKCA/ukca_radaer_precalc_mod.F90
M       src/atmosphere/UKCA/ukca_setup_indices.F90
M       src/atmosphere/UKCA/ukca_radaer_read_luts.F90
M       src/atmosphere/UKCA/ukca_radaer_set_aerosol_field.F90
M       src/atmosphere/UKCA/ukca_init.F90
M       src/atmosphere/UKCA/ukca_option_mod.F90
A  +    src/atmosphere/UKCA/ukca_mode_check_artefacts_mod.F90
M       src/atmosphere/convection/convec2_mod-6a.F90
M       src/atmosphere/convection/glue_conv-gconv5a.F90
M       src/atmosphere/convection/ni_conv_ctl.F90
M       src/atmosphere/convection/convec2-conv24a.F90
M       src/atmosphere/radiation_control/rad_input_mod.F90
M       src/atmosphere/radiation_control/r2_lwrad3z.F90
M       src/atmosphere/radiation_control/r2_re_mrf_umist.F90
M       src/atmosphere/radiation_control/swrdiag_mod.F90
M       src/atmosphere/radiation_control/set_swdiag_logic.F90
M       src/atmosphere/radiation_control/diagnostics_lw.F90
M       src/atmosphere/radiation_control/init_swdiag_logic.F90
M       src/atmosphere/radiation_control/allocate_sw_diag.F90
M       src/atmosphere/radiation_control/r2_cloud_level_diag.F90
M       src/atmosphere/radiation_control/deallocate_swdiag.F90
M       src/atmosphere/radiation_control/deallocate_lwdiag.F90
M       src/atmosphere/radiation_control/lwrdiag_mod.F90
M       src/atmosphere/radiation_control/r2_swrad3z.F90
M       src/atmosphere/radiation_control/allocate_lw_diag.F90
M       src/atmosphere/radiation_control/fill_missing_data_sw.F90
M       src/atmosphere/radiation_control/r2_set_cloud_field.F90
M       src/atmosphere/radiation_control/set_lwdiag_logic.F90
M       src/atmosphere/radiation_control/fill_missing_data_lw.F90
M       src/atmosphere/radiation_control/diagnostics_sw.F90
M       src/atmosphere/radiance_core/radiance_calc.F90
```

Before you make any further changes you should **commit** these changes. Do this now by typing the command

```
fcm commit
```

press **return** and put in an appropriate comment when asked if you want to commit this branch and type **y** and then press **return** again.

Now you need to add this branch into the UMUI, and turn off the original `vn8.4_UKCA_Tutorial_Solns` branch. We will initially use a **working copy**.

- Go to the *Model Selection → FCM Configuration → FCM Options for Atmosphere and Reconfiguration'* panel, scroll down the table and change the *Y* next to the *fcm:um_br/dev/luke/vn8.4_UKCA_Tutorials/src* branch to **N**.
- Add your own branch to this table. The revision URL will be off the form
  fcm:um_br/dev/your PUMA username/vn8.4_your_branch_name/src
  Leave the **revision** number column empty and put a **N** in the **Use Y/N** column.
- Turn on the button which says **Include modifications from a user working copy** and put the path to your source directory on PUMA in this box (e.g. something like **/home/your PUMA username/.../vn8.4_your_branch_name/src** etc.). Once we are happy that this code is working we can turn off the option to run from this copy, and turn on the option to run from the respository.

Using a **working copy** means that FCM will pick up any changes that you have made to your branch without you needed to `fcm commit` it.

Note: In general, you may, or may not, need to do this step. It is possible that you can make your edits on a fresh branch which contains only the UM trunk. However, it is likely that you will have conflicts of a sort with an existing branch, especially if the job you are using contains a large number (or a number of large) branches.

Reviewing your code changes and committing

After you have made code changes you will want to commit these to the repository using the

```
fcm commit
```

command. At this stage you may find it helpful to review those code changes using

```
fcm diff -g
```

which will bring up a **xxdiff** window comparing your working copy with the last version you checked in. It will highlight what changes have been made so you should be able to see more easily if there is a mistake. The

```
fcm status
```

command will also give a listing of the changed files. If you have added a new file you will also need to

```
fcm add name_of_file
```

prior to `fcm commit`-ing it.

Frequent committing of code to the FCM respository is encouraged.

You don't even need to commit working code, as committing is a useful way of backing-up code changes.

Editing the advected tracer list

To edit the UKCA source code, go to your branch and **cd** into

```
src/atmosphere/UKCA/
```

This directory contains the majority of the UKCA specfic code, and contains over 200 files.

Note: When we found which tracer slot to use, we did so consulting the **nm_spec** array in the **ukca_setd1defs.F90** routine. Due to the changes you have just merged in, **nm_spec** can now be found around line 130 in the **ukca_init.F90** routine. After version 8.4 **nm_spec** will now be found in **ukca_init.F90**.

You should open **ukca_init.F90** and edit the **nm_spec** array so that the slot you have chosen now contains the name of your tracer. The *nm_spec* array tells UKCA what slot in the 150 reserved tracers is used for which transported chemical species.

Note: remember that this name is **case sensitive**, and be wary of **tabs** as these **should not be used** in Fortran programs. Also remember that the string length for a UKCA species is **10 characters**.

Increase the number of tracers to be used

Also 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 *CheST/StratTrop* chemistry, which is located in the *Stratospheric Chemistry* section. The following IF block tests against **L_ukca_strattrop** (i.e. *CheST/StratTrop*, the chemistry scheme used) and **L_ukca_achem** (which determines whether or not you require the additional chemistry used to drive the GLOMAP-mode aerosol scheme). The settings for this particular job have both of these logicals as *.TRUE.*.

Inside this block you will find the following lines

```
           n_aero_tracers = 12
           n_chem_tracers = 71         ! No chem tracers
```

whether or not you are using the aerosol chemistry additions, as the base *CheST/StratTrop* chemistry is the same. You should increase the value of `n_chem_tracers` number by the number of chemical tracers that you are adding in the correct section of the IF block. If you are adding to a different chemistry scheme then you will need to make those changes accordingly.

Note: If you are using an older UM version which also requires the value of the `TR_UKCA` variable to be set (the total number of section 34 tracers which are switched on), you will find that this number may be different than that specified in the value of `TR_UKCA` which is set in the **SIZES** script. This is due to two reasons. The first is that there can be diagnostic tracers in section 34 (e.g. *Age of Air*, or *Passive O3*) which will appear in SIZES but not in the value of `n_chem_tracers` in ukca_setd1defs.F90. The second is that the UKCA H2O field is advected in section 0 and not section 34, and so will need to be included in ukca_setd1defs.F90, but is not needed in SIZES.

Editing the Chemistry Scheme Specification

For this example we are using the *CheST/StratTrop* chemistry scheme with the aerosol chemistry additions (as **GLOMAP-mode** is also being used), so we will need to edit the specification of this scheme so that it knows that it is using new tracer(s). The chemistry schemes are specified in the files called **ukca_chem_scheme.F90**. So we should open the file called **ukca_chem_strattrop.F90**.

The UKCA chemistry schemes are held within Fortran modules and are defined by different *derived types* which hold the specification of the different components required to describe the scheme. Looking through this file you will find two blocks which hold the different species used, called *chch_defs_strattrop_chem* (which is used when you are running chemistry-only) and *chch_defs_strattrop_aer* (which is used when you are running with the GLOMAP-mode aerosol scheme coupled to the chemistry). As we are currently using chemistry and aerosol, we will only need to edit the **chch_defs_strattrop_aer** specification. This currently looks like

```
TYPE(CHCH_T), PUBLIC :: chch_defs_strattrop_aer(1:87)=(/             &
!   1
chch_t(  1,'O(3P)     ',  1,'TR        ','Ox        ',  0,  0,  0),  &
!   2
```

```
chch_t(  2,'O(1D)     ',  1,'SS        ','Ox        ',  0,  0,  0),  &
!   3 DD: 1,WD: 1,
chch_t(  3,'O3        ',  1,'TR        ','Ox        ',  1,  1,  0),  &
!   4
chch_t(  4,'N         ',  1,'TR        ','NOx       ',  0,  0,  0),  &
!   5 DD: 2,
chch_t(  5,'NO        ',  1,'TR        ','NOx       ',  1,  0,  0),  &
!   6 DD: 3,WD: 2,
chch_t(  6,'NO3       ',  1,'TR        ','NOx       ',  1,  1,  0),  &
!   7 DD: 4,      EM: 1
chch_t(  7,'NO2       ',  1,'TR        ','NOx       ',  1,  0,  1),  &
!   8 DD: 5,WD: 3,
chch_t(  8,'N2O5      ',  1,'TR        ','          ',  1,  1,  0),  &
!   9 DD: 6,WD: 4,
chch_t(  9,'HO2NO2    ',  1,'TR        ','          ',  1,  1,  0),  &
!  10 DD: 7,WD: 5,
chch_t( 10,'HONO2     ',  1,'TR        ','          ',  1,  1,  0),  &
!  11 DD: 8,WD: 6,
chch_t( 11,'H2O2      ',  1,'TR        ','          ',  1,  1,  0),  &
!  12 EM: 2
chch_t( 12,'CH4       ',  1,'TR        ','          ',  0,  0,  2),  &
!  13 DD: 9,      EM:  3
chch_t( 13,'CO        ',  1,'TR        ','          ',  1,  0,  3),  &
!  14 DD:10,WD: 7,EM:  4
chch_t( 14,'HCHO      ',  1,'TR        ','          ',  1,  1,  4),  &
!  15       WD: 8,
chch_t( 15,'MeOO      ',  1,'TR        ','          ',  0,  1,  0),  &
!  16 DD:11,WD: 9,
chch_t( 16,'MeOOH     ',  1,'TR        ','          ',  1,  1,  0),  &
!  17
chch_t( 17,'H         ',  1,'TR        ','HOx       ',  0,  0,  0),  &
!  18
chch_t( 18,'H2O       ',  1,'TR        ','          ',  0,  0,  0),  &
!  19
chch_t( 19,'OH        ',  1,'TR        ','HOx       ',  0,  0,  0),  &
!  20       WD:10,
chch_t( 20,'HO2       ',  1,'TR        ','HOx       ',  0,  1,  0),  &
!  21
chch_t( 21,'Cl        ',  1,'TR        ','Clx       ',  0,  0,  0),  &
!  22
chch_t( 22,'Cl2O2     ',  1,'TR        ','Clx       ',  0,  0,  0),  &
!  23
chch_t( 23,'ClO       ',  1,'TR        ','Clx       ',  0,  0,  0),  &
!  24
chch_t( 24,'OClO      ',  1,'TR        ','          ',  0,  0,  0),  &
!  25
chch_t( 25,'Br        ',  1,'TR        ','Brx       ',  0,  0,  0),  &
!  26
chch_t( 26,'BrO       ',  1,'TR        ','Brx       ',  0,  0,  0),  &
!  27
chch_t( 27,'BrCl      ',  1,'TR        ','          ',  0,  0,  0),  &
!  28       WD:11,
chch_t( 28,'BrONO2    ',  1,'TR        ','          ',  0,  1,  0),  &
!  29
chch_t( 29,'N2O       ',  1,'TR        ','          ',  0,  0,  0),  &
!  30 DD:12,WD:12,
chch_t( 30,'HCl       ',  1,'TR        ','          ',  1,  1,  0),  &
!  31 DD:13,WD:13,
chch_t( 31,'HOCl      ',  1,'TR        ','          ',  1,  1,  0),  &
!  32 DD:14,WD:14,
chch_t( 32,'HBr       ',  1,'TR        ','          ',  1,  1,  0),  &
!  33 DD:15,WD:15,
chch_t( 33,'HOBr      ',  1,'TR        ','          ',  1,  1,  0),  &
!  34       WD:16,
chch_t( 34,'ClONO2    ',  1,'TR        ','          ',  0,  1,  0),  &
!  35
chch_t( 35,'CFCl3     ',  1,'TR        ','          ',  0,  0,  0),  &
!  36
chch_t( 36,'CF2Cl2    ',  1,'TR        ','          ',  0,  0,  0),  &
!  37
chch_t( 37,'MeBr      ',  1,'TR        ','          ',  0,  0,  0),  &
!  38 DD:16,WD:17,
chch_t( 38,'HONO      ',  1,'TR        ','          ',  1,  1,  0),  &
!  39             EM: 5
chch_t( 39,'C2H6      ',  1,'TR        ','          ',  0,  0,  1),  &
!  40
chch_t( 40,'EtOO      ',  1,'TR        ','          ',  0,  0,  0),  &
!  41 DD:17,WD:18,
chch_t( 41,'EtOOH     ',  1,'TR        ','          ',  1,  1,  0),  &
!  42 DD:18,      EM: 6
```

```
chch_t( 42,'MeCHO     ',  1,'TR        ','          ',  1,  0,  1),  &
!  43
chch_t( 43,'MeCO3     ',  1,'TR        ','          ',  0,  0,  0),  &
!  44 DD:19,
chch_t( 44,'PAN       ',  1,'TR        ','          ',  1,  0,  0),  &
!  45             EM: 7
chch_t( 45,'C3H8      ',  1,'TR        ','          ',  0,  0,  1),  &
!  46
chch_t( 46,'n-PrOO    ',  1,'TR        ','          ',  0,  0,  0),  &
!  47
chch_t( 47,'i-PrOO    ',  1,'TR        ','          ',  0,  0,  0),  &
!  48 DD:20,WD:19,
chch_t( 48,'n-PrOOH   ',  1,'TR        ','          ',  1,  1,  0),  &
!  49 DD:21,WD:20,
chch_t( 49,'i-PrOOH   ',  1,'TR        ','          ',  1,  1,  0),  &
!  50 DD:22,
chch_t( 50,'EtCHO     ',  1,'TR        ','          ',  1,  0,  0),  &
!  51
chch_t( 51,'EtCO3     ',  1,'TR        ','          ',  0,  0,  0),  &
!  52             EM: 8
chch_t( 52,'Me2CO     ',  1,'TR        ','          ',  0,  0,  1),  &
!  53
chch_t( 53,'MeCOCH2OO ',  1,'TR        ','          ',  0,  0,  0),  &
!  54 DD:23,WD:21,
chch_t( 54,'MeCOCH2OOH',  1,'TR        ','          ',  1,  1,  0),  &
!  55 DD:24,
chch_t( 55,'PPAN      ',  1,'TR        ','          ',  1,  0,  0),  &
!  56
chch_t( 56,'MeONO2    ',  1,'TR        ','          ',  0,  0,  0),  &
!  57             EM: 9
chch_t( 57,'C5H8      ',  1,'TR        ','          ',  0,  0,  1),  &
!  58
chch_t( 58,'ISO2      ',  1,'TR        ','          ',  0,  0,  0),  &
!  59 DD:25,WD:22,
chch_t( 59,'ISOOH     ',  1,'TR        ','          ',  1,  1,  0),  &
!  60 DD:26,WD:23,
chch_t( 60,'ISON      ',  1,'TR        ','          ',  1,  1,  0),  &
!  61 DD:27,
chch_t( 61,'MACR      ',  1,'TR        ','          ',  1,  0,  0),  &
!  62
chch_t( 62,'MACRO2    ',  1,'TR        ','          ',  0,  0,  0),  &
!  63 DD:28,WD:24,
chch_t( 63,'MACROOH   ',  1,'TR        ','          ',  1,  1,  0),  &
!  64 DD:29,
chch_t( 64,'MPAN      ',  1,'TR        ','          ',  1,  0,  0),  &
!  65 DD:30,WD:25,
chch_t( 65,'HACET     ',  1,'TR        ','          ',  1,  1,  0),  &
!  66 DD:31,WD:26,
chch_t( 66,'MGLY      ',  1,'TR        ','          ',  1,  1,  0),  &
!  67 DD:32,
chch_t( 67,'NALD      ',  1,'TR        ','          ',  1,  0,  0),  &
!  68 DD:33,WD:27,
chch_t( 68,'HCOOH     ',  1,'TR        ','          ',  1,  1,  0),  &
!  69 DD:34,WD:28,
chch_t( 69,'MeCO3H    ',  1,'TR        ','          ',  1,  1,  0),  &
!  70 DD:35,WD:29,
chch_t( 70,'MeCO2H    ',  1,'TR        ','          ',  1,  1,  0),  &
!  71
chch_t( 71,'H2        ',  1,'TR        ','          ',  0,  0,  0),  &
!  72 DD:36,WD:30,
chch_t( 72,'MeOH      ',  1,'TR        ','          ',  1,  1,  0),  &
!  73
chch_t( 73,'CO2       ',  1,'CT        ','          ',  0,  0,  0),  &
!  74
chch_t( 74,'O2        ',  1,'CT        ','          ',  0,  0,  0),  &
!  75
chch_t( 75,'N2        ',  1,'CT        ','          ',  0,  0,  0),  &
!  76
chch_t( 76,'DMS       ',  1,'TR        ','          ',  0,  0,  0),  &
!  77 DD:37,WD:31,EM:10
chch_t( 77,'SO2       ',  1,'TR        ','          ',  1,  1,  1),  &
!  78
chch_t( 78,'H2SO4     ',  1,'TR        ','          ',  0,  0,  0),  &
!  79
chch_t( 79,'MSA       ',  1,'TR        ','          ',  0,  0,  0),  &
!  80 DD:38,WD:32
chch_t( 80,'DMSO      ',  1,'TR        ','          ',  1,  1,  0),  &
!  81 DD:39,WD:33,EM:11
chch_t( 81,'NH3       ',  1,'TR        ','          ',  1,  1,  1),  &
!  82
```

```
chch_t( 82,'CS2       ',  1,'TR        ','          ',  0,  0,  0),  &
!  83
chch_t( 83,'COS       ',  1,'TR        ','          ',  0,  0,  0),  &
!  84
chch_t( 84,'H2S       ',  1,'TR        ','          ',  0,  0,  0),  &
!  85
chch_t( 85,'SO3       ',  1,'TR        ','          ',  0,  0,  0),  &
!  86 DD:40,      EM: 12
chch_t( 86,'Monoterp  ',  1,'TR        ','          ',  1,  0,  1),  &
!  87 DD:41,WD:34
chch_t( 87,'Sec_Org   ',  1,'TR        ','          ',  1,  1,  0)   &
  /)
```

You should note that even though there are 83 chemical tracers defined, there are actually 87 species considered in this scheme, as three (CO2, O2, and N2) are constants and O(1D) is set to be in steady-state and is not transported. This is set in the 4th column of the list:

- TR = tracer
- CT = constant
- SS = steady state

The 6th and 7th columns of the list control dry and wet deposition respectively. We will cover these in more detail in the adding dry deposition of chemical species and adding wet deposition of chemical species tutorials.

You should edit this list, adding the name your tracer(s) (exactly as it is called in the **nm_spec** array) on at the end (remember to correctly add/remove the comma on the last line!). Make sure that the last 3 columns are all set to **0**. Also remember to increase the size of the **chch_defs_strattrop_chem_aer** array by the number of species that you are adding.

Define conversion factors for your new tracers

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 (**Note:** in UM7.3 this is done in the **c_v_m.h** include file).

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.

Tell UKCA to use these conversion factors

After you have added these in, 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 contains entries like these:

```
    ...
    WHERE (advt == 'MEMALD    ') c_species = c_memald
    WHERE (advt == 'MVK       ') c_species = c_mvk
    WHERE (advt == 'MVKOOH    ') c_species = c_mvkooh
    WHERE (advt == 'TOLUENE   ') c_species = c_toluene
    ...
```

The **advt** array is automatically generated by UKCA at run-time from the **CHCH** 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 block.

Increase the size of JPCTR and JPSPEC

As you have changed the number of chemical species, you will need to increase the size of two integer parameters that tell UKCA how many chemical species and how many chemical tracers it should consider.

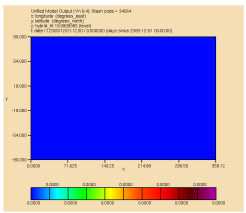
- **JPSPEC** is the number of chemical species used in UKCA (effectively the size of *chch_defs_strattrop_chem* or *chch_defs_strattrop_chem*, whichever is being used by the UKCA configuration you are editing)
- **JPCTR** is the number of chemical tracers used in UKCA (always less-than or equal-to JPSPEC, effectively the number **TR** in *chch_defs_strattrop_chem*/*chch_defs_strattrop_aer*, or the new value of **n_chem_tracers (+ n_aero_tracers)** that you set in *ukca_setd1defs.F90*)

These are currently set automatically by the UMUI (depending on which chemistry scheme you choose). As this cannot be edited from within the UMUI you will need to make a hand-edit to change it. The values of *JPSPEC* and *JPCTR* are set in **CNTLATM**. Your hand-edit should contain a block similar to the following:

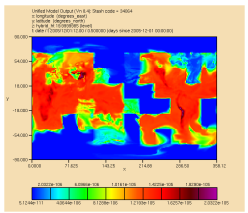
```
ed CNTLATM<<\EOF
/JPCTR/
d
i
 JPCTR = value of (n_chem_tracers + n_aero_tracers),
.
/JPSPEC/
d
i
 JPSPEC = new size of chch_defs_strattrop_chem or chch_defs_strattrop_aer array (whichever is being used by the
.
w
q
EOF
```

Save this hand-edit and make it executable, and place it in the table in **Model Selection → Input/Output Control and Resource → User hand edit files** with a **Y** in the **Include Y/N** column.

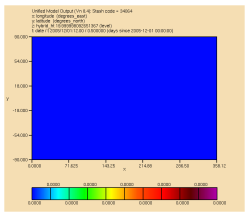
Numerical Noise



Tracer field from the UM fieldsfile prior to inclusion in UKCA



Tracer field from the UM fieldsfile after inclusion in UKCA showing the numerical noise



Tracer field from the UM pp-file after inclusion in UKCA showing that the numerical noise is no longer evident

When you first add your tracer to the Unified Model and then view it in an output file, you will see that the field contains zeros. However, after then adapting the UKCA code to include it (but not yet including any reactions or emissions etc) you may now find that the tracer field appears to be non-zero. If you look closer at the output you will find that the values in this field are exceptionally small, usually in the range of $10^{-\mathcal{O}(\geq 80)}$. This is because the UKCA chemical solver will be adding a small amount of numerical noise to this field. The output files from the UM are 64-bit fieldsfiles, and if you convert these to 32-bit pp-files by e.g. using `ff2pp` which can be found at

```
/work/n02/n02/hum/bin/ff2pp
```

on ARCHER, and

```
/projects/um1/vn8.4/ibm/utils/ff2pp
```

on MONSooN, you will see that these artifacts are removed when converted to 32-bit. The meaningful values of the tracers (usually in the $10^{-\mathcal{O}(6)}$ to $10^{-\mathcal{O}(12)}$ range) will be unaffected.

Solution to Task 4.2: Add these two new tracers to UKCA

Please see this page for a solution to Task 4.2

Written by Luke Abraham 2014

Solution to UKCA Chemistry and Aerosol Tutorial 4 Task 4.1

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Back to the adding new chemical tracers tutorial

Contents

- 1 Task
- 2 Solution
  - 2.1 1. The user STASHmaster file
  - 2.2 2. Initialise the new tracers
  - 2.3 3. Add the hand-edit
  - 2.4 Changes to STASH
- 3 Output
- 4 Worked Solution

Task

You were asked to

> Add in two new tracers in to slots 64 and 65. The tracer in slot 64 will be called **ALICE** and the tracer in slot 65 will be called **BOB**. You should also output these two tracers through the **pa/UPA** stream in STASH as daily means.

Solution

1. The user STASHmaster file

Go to **Model Selection → Atmosphere → STASH → User-STASHmaster files. Diags, Progs & Ancills.**. In this panel you will see that a number of user STASHmaster files listed. The one containing the tracers is *UKCA_Tr_StratTrop.prestash*.

Take a copy of the **UKCA_Tr_StratTropAeroMODE.prestash** user STASHmaster file, which can be found at

```
/home/ukca/userprestash/VN8.4/UKCA_Tr_StratTropAeroMODE.prestash
```

on PUMA, and copy it to your own directory. Edit this file to add the lines

```
#
1|    1 |   34 |   64 |UKCA TUTORIAL TRACER: ALICE          |
2|    2 |    0 |    1 |    1 |    2 |   10 |   11 |    0 |    0 |    0 |    0 |
3| 00000000000000000000000000064 | 00000000000000000001 |    1 |
4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
5|    0 | 1861 |    0 |   65 |    0 |    0 |    0 |    0 |    0 |
#
1|    1 |   34 |   65 |UKCA TUTORIAL TRACER: BOB            |
2|    2 |    0 |    1 |    1 |    2 |   10 |   11 |    0 |    0 |    0 |    0 |
3| 00000000000000000000000000065 | 00000000000000000001 |    1 |
4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
5|    0 | 1861 |    0 |   65 |    0 |    0 |    0 |    0 |    0 |
#
```

An example file can be found at

```
/home/ukca/userprestash/VN8.4/Tutorial/Task4.1_UKCA_Tr_StratTropAeroMODE.prestash
```

2. Initialise the new tracers

Go to **Model Selection → Atmosphere → STASH → Initialisation of User Prognostics**, scroll down the list of variables until you find your new tracers and set these to option **3** (i.e. zero).

3. Add the hand-edit

Run the **/home/ukca/bin/make_tracer_list** script on your user STASHmaster file to produce the following output

```
TC_UKCA=1,1,1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,0,1,1,1,0,1,1,1,1,1,1,1,1,1,1,0,0,0,
1,1,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,0,0,0,1,1,1,1,1,
1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,0,1,1,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,1,1,
TR_UKCA=106,
```

You can make a hand-edit file like this

```
# Hand edit to add tracers for UKCA tutorial
# vn8.4  64:ALICE, 65:BOB

ed SIZES<<\EOF
/TC_UKCA=/
d
d
d
d
d
i
 TC_UKCA=1,1,1,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
 1,0,1,1,1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,0,1,1,1,0,1,1,1,1,1,1,1,1,1,1,0,0,0,
 1,1,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,0,0,0,1,1,1,1,1,
 1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,0,1,1,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
 0,0,0,0,0,0,0,1,1,
.
w
q
EOF
```

and make this script executable (`chmod a+rx script.ed`).

An example of this hand-edit can be found at

```
/home/ukca/hand_edits/VN8.4/Tutorial/Task4.1_add_UKCA_Tr_StratTropAeroMODE.ed
```

Add your hand-edit to the UMUI at **Model Selection → Input/Output Control and Resource → User hand edit files** by placing it in the table and putting a **Y** in the *Include Y/N* column.

Changes to STASH

Go to the STASH panel at **Model Selection → Atmosphere → STASH → STASH. Specification of Diagnostic requirements** and go to **Diagnostics → Load New Diagnostics** and scroll down the list until you come to section **34** and double-click where it says **UKCA Chemistry**. Scroll down the list until you get to the ALICE and BOB tracers, and add them to STASH like so

```
34  64 UKCA TUTORIAL TRACER: ALICE         TDAYM    DALLTH   UPA      Y +  N X
34  65 UKCA TUTORIAL TRACER: BOB           TDAYM    DALLTH   UPA      Y +  N X
```

You will see that the final two columns say *N* (*Avail*) and *X* (*I+P+A*). This is because STASH thinks that these tracers are not available as they are not turned on until the *Process* stage when the hand-edit defined above is applied. Also, if you verify diagnostics you get the following warning

```
Diag: "UKCA TUTORIAL TRACER: ALICE         " (34,64) (TDAYM,DALLTH,UPA)
  DIAGNOSTIC    ERROR: Diagnostic is not available for this model configuration.
Diag: "UKCA TUTORIAL TRACER: BOB           " (34,65) (TDAYM,DALLTH,UPA)
  DIAGNOSTIC    ERROR: Diagnostic is not available for this model configuration.
```

However, you should not worry about this, as the tracers are available and will be output. However, they will contain zeros as nothing has been placed into them (yet).

Note: If you find that you get this error message, but with different names (although the section and item numbers are the same) then you should **save** your job and **close** it. Now when you re-open it and open the STASH panel again you will find that the names have been updated. This occurs because STASH only reads the user STASHmaster files the first time the panel is opened (in that job instance) so you must re-load the job for the changes to properly take effect.

Output

If you open a **pa** file in your job directory you will find that it now contains the fields

```
0    : 192   145   85    1     Stash code = 34001
1    : 192   145   85    1     Stash code = 34064
2    : 192   145   85    1     Stash code = 34065
```

If you open 34064 and 34065 you will find that they only contain zeros.

Sample output from this job can be found in

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Task4.1/
```

on ARCHER, and in

```
/projects/ukca/Tutorial/vn8.4/sample_ouput/Task4.1/
```

on MONSooN.

Worked Solution

There is a worked solution to this problem in the UMUI Tutorial experiment. This is job **c**: *Tutorial: solution to Task 4.1 - adding new chemical tracers to the UMUI*.

Written by Luke Abraham 2014

- This page was last modified on 15 December 2015, at 15:06.

Solution to UKCA Chemistry and Aerosol Tutorial 4 Task 4.2

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Back to the adding new chemical tracers tutorial

Contents

- 1 Task
- 2 Solution
  - 2.1 Changes to ukca_init.F90
  - 2.2 Changes to ukca_setd1defs.F90
  - 2.3 Changes to ukca_chem_strattrop.F90
  - 2.4 Changes to ukca_constants.F90
  - 2.5 Changes to ukca_cspecies.F90
  - 2.6 Hand-edit to increase values of JPCTR and JPSPEC
- 3 Output
- 4 Worked Solution

Task

You were asked to

> Make the required code changes so that your **ALICE** and **BOB** tracers are now specified in the UKCA CheST/StratTrop scheme. You should set the conversion factor for each of these to 1.0.

Solution

The solution to this task involves making changes to 4 files in the UKCA sub-directory of your source code. These files are

```
ukca_init.F90
ukca_setd1defs.F90
ukca_chem_strattrop.F90
ukca_constants.F90
ukca_cspecies.F90
```

The following changes need to be made

Changes to ukca_init.F90

The **nm_spec** array has been edited to add in the ALICE and BOB tracers (changes in red):

```
       nm_spec(1:n_all_tracers) = (/                                    &
      '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      ','ALICE     ','BOB       ',  &
      '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
        /)
```

Changes to ukca_setd1defs.F90

The value of **n_chem_tracers** need to be increased from 71 to **73**, e.g.

```
        ELSE IF (L_ukca_strattrop .AND. L_ukca_achem) THEN
           n_chem_emissions = 21      ! em_chem_spec below
           n_3d_emissions   = 2       ! volc SO2 & aircraft NOX
           ALLOCATE(em_chem_spec(n_chem_emissions+n_3d_emissions))
           em_chem_spec =                                          &
               (/'NO        ','CH4       ','CO        ','HCHO      ', &
                 'C2H6      ','C3H8      ','Me2CO     ','MeCHO     ', &
                 'C5H8      ','BC_fossil ','BC_biofuel','OC_fossil ', &
                 'OC_biofuel','Monoterp  ','NVOC      ','SO2_low   ', &
                 'SO2_high  ','DMS       ','NH3       ','SO2_nat   ', &
                 'BC_biomass','OC_biomass','NO_aircrft'/)
           n_aero_tracers = 12
           n_chem_tracers = 73         ! No. chem tracers
           IF (L_ukca_trophet) THEN
             nr_therm     = 241        ! thermal reactions
           ELSE
             nr_therm     = 239        ! thermal reactions
           ENDIF
           nr_phot        = 59         ! photolytic (ATA)
```

Changes to ukca_chem_strattrop.F90

The following lines have been added to the end of the **chch_defs_strattrop_chem** specification:

```
!  88
chch_t( 88,'ALICE     ',  1,'TR        ','          ',  0,  0,  0),  &
!  89
chch_t( 90,'BOB       ',  1,'TR        ','          ',  0,  0,  0)   &
```

(also, a comma has been added after the final ")" on array line 87, specifying *Sec_Org*).

The size of the `chch_defs_strattrop_aer` array has been increased from 87 to **89**.

Changes to ukca_constants.F90

The following lines have been added to the UKCA_CONSTANTS module:

```
!     UKCA Tutorial tracers
      REAL, PARAMETER :: C_ALICE      = 1.0000
      REAL, PARAMETER :: C_BOB        = 1.0000
```

Changes to ukca_cspecies.F90

The following lines have been added to the UKCA_CALC_CSPECIES subroutine which is held within the UKCA_CSPECIES module

```
!     UKCA Tutorial Tracers
      WHERE (advt == 'ALICE     ') c_species = C_ALICE
      WHERE (advt == 'BOB       ') c_species = C_BOB
```

Hand-edit to increase values of JPCTR and JPSPEC

A hand-edit is required to increase the values of **JPCTR** and **JPSPEC**. This should contain the following:

```
ed CNTLATM<<\EOF
/JPCTR/
d
i
 JPCTR = 85,
.
/JPSPEC/
d
i
 JPSPEC = 89,
.
w
```

```
q
EOF
```

An example can be found at

```
/home/ukca/hand_edits/VN8.4/Tutorial/Task4.2_incr_JPvals.ed
```

Output

If you open the **pa** file in your job directory you will find that it still contains the fields

```
0    : 192   145   85    1     Stash code = 34001
1    : 192   145   85    1     Stash code = 34064
2    : 192   145   85    1     Stash code = 34065
```

If you open 34064 and 34065 you will find that they still only contain zeros. This will change in later tasks as we add-in emissions and reactions.

Sample output from this job can be found in

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Task4.2/
```

on ARCHER, and in

```
~~/projects/ukca/Tutorial/vn8.4/sample_ouput/Task4.2/~~
```

on MONSooN.

Worked Solution

There is a worked solution to this problem in the UMUI Tutorial experiment. This is job **d**: *Tutorial: solution to Task 4.2 - adding new chemical tracers to UKCA*.

The code changes can be viewed by using the following FCM command (add `-g` for to view in xxdiff)

```
fcm diff fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns@14676 fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns@14692
```

This gives the following output:

```
Index: src/atmosphere/UKCA/ukca_setd1defs.F90
===================================================================
--- src/atmosphere/UKCA/ukca_setd1defs.F90      (revision 14676)
+++ src/atmosphere/UKCA/ukca_setd1defs.F90      (revision 14692)
@@ -284,7 +284,7 @@
                     'SO2_high  ','DMS       ','NH3       ','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_constants.F90
===================================================================
--- src/atmosphere/UKCA/ukca_constants.F90      (revision 14676)
+++ src/atmosphere/UKCA/ukca_constants.F90      (revision 14692)
@@ -267,6 +267,10 @@
       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

Index: src/atmosphere/UKCA/ukca_chem_strattrop.F90
===================================================================
--- src/atmosphere/UKCA/ukca_chem_strattrop.F90 (revision 14676)
+++ src/atmosphere/UKCA/ukca_chem_strattrop.F90 (revision 14692)
@@ -218,7 +218,7 @@
 chch_t( 75,'N2        ',  1,'CT        ','          ',  0,  0,  0)   &
   /)
```

```
-TYPE(CHCH_T), PUBLIC :: chch_defs_strattrop_aer(1:87)=(/             &
+TYPE(CHCH_T), PUBLIC :: chch_defs_strattrop_aer(1:89)=(/             &
 !   1
 chch_t(  1,'O(3P)     ',  1,'TR        ','Ox        ',  0,  0,  0),  &
 !   2
@@ -392,7 +392,11 @@
 !  86 DD:40,      EM: 12
 chch_t( 86,'Monoterp  ',  1,'TR        ','          ',  1,  0,  1),  &
 !  87 DD:41,WD:34
-chch_t( 87,'Sec_Org   ',  1,'TR        ','          ',  1,  1,  0)   &
+chch_t( 87,'Sec_Org   ',  1,'TR        ','          ',  1,  1,  0),  &
+!  88
+chch_t( 88,'ALICE     ',  1,'TR        ','          ',  0,  0,  0),  &
+!  89
+chch_t( 90,'BOB       ',  1,'TR        ','          ',  0,  0,  0)   &
    /)

 TYPE(RATB_T) :: ratb_defs_strattrop_chem(200)
Index: src/atmosphere/UKCA/ukca_cspecies.F90
===================================================================
--- src/atmosphere/UKCA/ukca_cspecies.F90       (revision 14676)
+++ src/atmosphere/UKCA/ukca_cspecies.F90       (revision 14692)
@@ -272,6 +272,9 @@
       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: src/atmosphere/UKCA/ukca_init.F90
===================================================================
--- src/atmosphere/UKCA/ukca_init.F90   (revision 14676)
+++ src/atmosphere/UKCA/ukca_init.F90   (revision 14692)
@@ -183,7 +183,7 @@
          '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
```

Written by Luke Abraham 2014

- This page was last modified on 15 December 2015, at 15:05.

UKCA Chemistry and Aerosol Tutorial 5

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Contents

- 1 What you will learn in this Tutorial
- 2 Task 5.1: Create a new emissions file and use it in your job
- 3 Part 1. Making a new Emissions Ancillary File
  - 3.1 Regridding data with Xconv
    - 3.1.1 Horizontal Regridding
    - 3.1.2 Vertical Regridding
  - 3.2 Choosing a STASH slot for your new emission(s)
    - 3.2.1 Listing of emissions from STASH
    - 3.2.2 Code in ukca_setd1defs.F90
    - 3.2.3 Emissions STASHmaster File
  - 3.3 Using Xancil
    - 3.3.1 Extract your current emissions
    - 3.3.2 Make a new emissions ancillary file
  - 3.4 Use your new emissions file in the UMUI
- 4 Solution to Task 5.1: Create a new emissions file and use it in your job
- 5 Task 5.2: make the required code changes to add your emission into UKCA
- 6 Part 2. UKCA Code Changes
  - 6.1 Changes to ukca_setd1defs.F90
    - 6.1.1 Add the species to em_chem_spec
    - 6.1.2 Tell UKCA the STASH code associated with your new emission
  - 6.2 Changes to ukca_constants.F90
  - 6.3 Changes to ukca_emission_ctl.F90
- 7 Solution to Task 5.2: make the required code changes to add your emission into UKCA

What you will learn in this Tutorial

During this tutorial you will learn how to make new UM ancillary 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 ancillary file that the UM 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 ancillary file will be the same for a species which is currently emitted into, although in this simplier case you would not need to make any code changes.

During this tutorial you will be tasked with making a new emissions ancillary file, and add the emissions contained within it into one of your new tracers.

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

Task 5.1: In the

```
/work/n02/n02/ukca/Tutorial/vn8.4/Task5.1
```

directory on ARCHER, or the

```
~~/projects/ukca/Tutorial/vn8.4/Task5.1~~
```

directory on MONSooN, 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 **N96**, and then make a new surface emissions ancillary file with this as slot **316**. You should then use this new file, and the corresponding user STASHmaster file, in your UMUI job.

Note: If you were unable to successfully complete Task 4.2, then please take a copy of the **d** job from the Tutorial experiment (*Tutorial: solution to Task 4.2 - adding new chemical tracers to UKCA*) and work from there, as this will allow you to make only the changes required for this task. Please also make a new branch and merge-in branch **fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns** at revision number **14692** to allow you to proceed.

Part 1. Making a new Emissions Ancillary File

The UM uses its own format to read-in initial and emissions data, the *ancillary file*. UKCA makes use of these files for the surface and aircraft emissions, and these files can easily be made up from netCDF data using the Xancil (http://cms.ncas.ac.uk/documents/xancil/) program. However, before we can use Xancil to create our emissions ancillary, we may first need to use Xconv (http://badc.nerc.ac.uk/help/software/xconv/) to regrid the raw data to the correct resolution of the UM configuration that you are using.

Regridding data with Xconv

We are using a model at N96L85 resolution. In the horizontal this is 1.875 degrees by 1.25 degrees. There are 85 vertical levels.

The **N96** UM grid

1. Has 192 points in longitude (x) and 145 points in latitude (y)
2. Starts at 0.0 longitude with a spacing of 1.875 degrees
3. Starts at -90.0 latitude (i.e. the South Pole) with a spacing of 1.25 degrees

Horizontal Regridding

Horizontal regridding in Xconv is straight-forward. First, open your dataset in Xconv by

```
xconv -i file.nc
```

and double-click on the field that you want to regrid and then click the **Trans** button at the far top right. This will show a window similar to Figure 1 (which in fact shows the default settings for a N96 field).

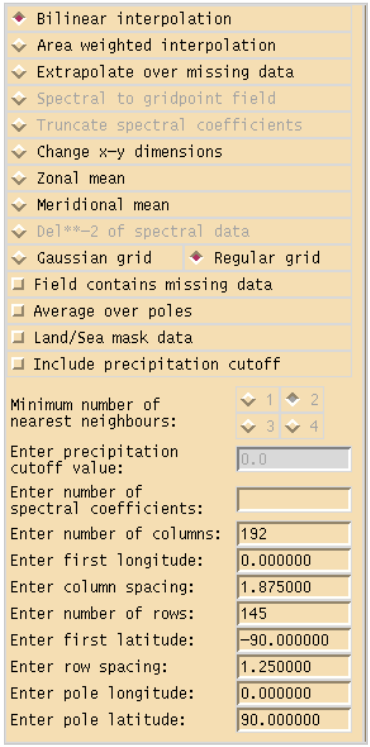


Figure 1: The Xconv *Trans* window.

As we are regridding an emission we need to select **area weighted interpolation** as we need to conserve the total amount of quantity emitted. Then scroll-down to the boxes at the end and enter the following values:

- Number of columns = 192
- First longitude = 0.000000
- Column spacing = 1.875000
- Number of rows = 145
- First latitude = -90.000000
- Row spacing = 1.250000

which match up with the grid definition above, and ensure that

- Pole longitude = 0.000000
- Pole latitude = 90.000000

Now click **Apply**. You should be given a message similar to

```
Area weighted interpolation from 720x360 Regular grid to 192x145 Regular grid
```

in the dialogue window. You can now extract this data as a new netCDF file (you cannot re-save a file in Xconv) by putting a name in the **output file name** box and clicking **convert**.

Vertical Regridding

It is not possible to vertically regrid data using Xconv. You will need to do this in another way. If you need to vertically regrid data, and am unsure of the best way, please contact Luke Abraham for advice.

Remember that

- Emissions data must be re-gridded in a mass conserving way, so you will probably need to integrate the field on one grid and then decompose it again on the new grid.
- Tracer data can be fitted to the profile of the field on the old grid.

Choosing a STASH slot for your new emission(s)

To make a new ancillary file for your new emission(s), you should first decide on a STASH item for it/them. Currently UKCA makes use of the **user single-level ancillary file** and **user multi-level ancillary file** which uses STASH section 0 items 301-320 (single-level) and 321-340 (multi-level). What these numbers correspond to is set in the file **ukca_setd1defs.F90**, as well as in the user STASHmaster file associated with the job you are using (which can be found in **Model Selection → Atmosphere → STASH → User-STASHmaster files. Diags, Progs & Ancills**).

Listing of emissions from STASH

Stash code	Emission
301	NOx surf emissions
302	CH4 surf emissions
303	CO surf emissions
304	HCHO surf emissions
305	C2H6 surf emissions
306	C3H8 surf emissions
307	ME2CO surf emissions
308	MECHO surf emissions
309	C5H8 surf emissions
310	BC fossil fuel surf emissions
311	BC biofuel surf emissions
312	OC fossil fuel surf emissions
313	OC biofuel surf emissions
314	Monoterpene surf emissions
315	NVOC surf emissions
322	BC BIOMASS 3D EMISSION
323	OC BIOMASS 3D EMISSION
340	NOX AIRCRAFT EMS IN KG/S/GRIDCELL

Code in ukca_setd1defs.F90

The species emitted are set in two places, firstly in the definition of an array called **em_chem_spec** which is scheme specific, and secondly in a block of code which searches through the *em_chem_spec* array and assigns a STASH number to it (as defined by the list above).

For example, for the *CheST/StratTrop* chemistry (not using aerosol chemistry), em_chem_spec is set to

```
           em_chem_spec =                                              &
               (/'NO        ','CH4       ','CO        ','HCHO      ', &
                 'C2H6      ','C3H8      ','Me2CO     ','MeCHO     ', &
                 'C5H8      ','NO_aircrft'/)
```

This can be found in the **Stratospheric Chemistry** section, controlled by the IF block where `(L_ukca_strattrop .AND. .NOT. L_ukca_achem)`. If you are using aerosol chemistry then em_chem_spec is set to

```
           em_chem_spec =                                              &
               (/'NO        ','CH4       ','CO        ','HCHO      ', &
                 'C2H6      ','C3H8      ','Me2CO     ','MeCHO     ', &
                 'C5H8      ','BC_fossil ','BC_biofuel','OC_fossil ', &
                 'OC_biofuel','Monoterp  ','NVOC      ','SO2_low   ', &
                 'SO2_high  ','DMS       ','NH3       ','SO2_nat   ', &
                 'BC_biomass','OC_biomass','NO_aircrft'/)
```

in the section of the IF block controlled by `(L_ukca_strattrop .AND. L_ukca_achem)`.

Further down the code there is this block of code:

```
    J = n_use_tracers
    IF (n_chem_emissions+n_3d_emissions+n_mode_emissions > 0) THEN
      DO i=1,n_chem_emissions + n_3d_emissions
        UkcaD1Codes(J+i)%section    = 0
        UkcaD1Codes(J+i)%item       = n_emiss_first+i-1  ! trop chemistry
        UkcaD1Codes(J+i)%len_dim1   = row_length         ! uses stash codes
        UkcaD1Codes(J+i)%len_dim2   = rows               ! 301-309 for
        UkcaD1Codes(J+i)%required   = .true.             ! surface emissions
        UkcaD1Codes(J+i)%prognostic = .true.             ! from Section 0
! Special cases, emissions already available in UM
        IF (em_chem_spec(i)(1:7) == 'SO2_low') THEN
          UkcaD1Codes(J+i)%item     = 58
          IF (.NOT. L_SO2_SURFEM .AND. (L_ukca_aerchem .OR.           &
                                        L_ukca_achem)) THEN
            cmessage='SO2 surface emissions from UM are not flagged'
            errcode=58

            CALL EREPORT('UKCA_SETD1DEFS',errcode,cmessage)
          ENDIF
        ELSEIF (em_chem_spec(i)(1:7) == 'SO2_nat') THEN
          UkcaD1Codes(J+i)%item     = 121
          UkcaD1Codes(J+i)%len_dim3 = tr_levels
          IF (.NOT. L_SO2_NATEM .AND. (L_ukca_aerchem .OR.            &
                    L_ukca_achem)) THEN
            cmessage='SO2 natural emissions from UM are not flagged'
            errcode=121

            CALL EREPORT('UKCA_SETD1DEFS',errcode,cmessage)
          ENDIF
        ELSEIF (em_chem_spec(i)(1:8) == 'SO2_high') THEN
          UkcaD1Codes(J+i)%item     = 126
          IF (.NOT. L_SO2_HILEM .AND. (L_ukca_aerchem .OR.            &
                                       L_ukca_achem)) THEN
            cmessage='SO2 high-level emissions are not flagged'
            errcode = UkcaD1Codes(J+i)%item
            CALL EREPORT('UKCA_SETD1DEFS',errcode,cmessage)
          ENDIF
        ELSEIF (em_chem_spec(i)(1:3) == 'NH3') THEN
          UkcaD1Codes(J+i)%item     = 127
          IF (.NOT. L_NH3_EM .AND. (L_ukca_aerchem .OR.               &
                                    L_ukca_achem)) THEN
            cmessage='NH3 surface emissions from UM are not flagged'
            errcode = UkcaD1Codes(J+i)%item
            CALL EREPORT('UKCA_SETD1DEFS',errcode,cmessage)
          ENDIF
        ELSEIF (em_chem_spec(i) == 'BC_fossil ') THEN
          UkcaD1Codes(J+i)%item = 310
        ELSEIF (em_chem_spec(i) == 'BC_biofuel') THEN
          UkcaD1Codes(J+i)%item = 311
        ELSEIF (em_chem_spec(i) == 'OC_fossil ') THEN
```

```
          UkcaD1Codes(J+i)%item = 312
        ELSEIF (em_chem_spec(i) == 'OC_biofuel') THEN
          UkcaD1Codes(J+i)%item = 313
        ELSEIF (em_chem_spec(i) == 'Monoterp  ') THEN
          UkcaD1Codes(J+i)%item = 314
        ELSEIF (em_chem_spec(i) == 'NVOC      ') THEN
          UkcaD1Codes(J+i)%item = 315
        ELSEIF (em_chem_spec(i) == 'BC_biomass') THEN
          UkcaD1Codes(J+i)%item = 322
          UkcaD1Codes(J+i)%len_dim3 = tr_levels
        ELSEIF (em_chem_spec(i) == 'OC_biomass') THEN
          UkcaD1Codes(J+i)%item = 323
          UkcaD1Codes(J+i)%len_dim3 = tr_levels
        ELSEIF (em_chem_spec(i) == 'SO2_biomas') THEN
          UkcaD1Codes(J+i)%item = 324
          UkcaD1Codes(J+i)%len_dim3 = tr_levels
        ELSEIF (em_chem_spec(i)(1:3) == 'DMS') THEN
          UkcaD1Codes(J+i)%section  = 17
          UkcaD1Codes(J+i)%item     = 205
          UkcaD1Codes(J+i)%prognostic = .false.
          IF (.NOT. L_DMS_EM .AND. (L_ukca_aerchem .OR.               &
                    L_ukca_achem)) THEN
            cmessage='DMS surface emissions from UM are not flagged'
            errcode = UkcaD1Codes(J+i)%section*1000 +                  &
                      UkcaD1Codes(J+i)%item
            CALL EREPORT('UKCA_SETD1DEFS',errcode,cmessage)
          ENDIF
        ELSEIF (em_chem_spec(i)(1:7) == 'NO_airc') THEN
          UkcaD1Codes(J+i)%item     = 340
          UkcaD1Codes(J+i)%len_dim3 = tr_levels
        ENDIF
      ENDDO
    ENDIF
```

This block of code is rather complicated, but what it essentially means is that for the STASH codes 301-309, the emissions are assumed to be in the order of the species in *em_chem_spec*, but for the other emissions the STASH numbers are explicitly defined. As you can see from the table above, for 2D (surface) emissions the numbers 301-315 are reserved, and for 3D emissions the numbers 322, 323, and 340 are reserved.

This means that if you are adding in a new surface emission(s) it is best to use the slots 316-320, unless you need more than 5 slots. For 3D emissions you have more leeway.

Emissions STASHmaster File

Now that you have selected your slot(s), you need to create a new STASH specification for it/them. The easiest way to do this is to copy the existing user STASHmaster file that defines your current (possible) emissions, and extend that. This is found in **Model Selection → Atmosphere → STASH → User-STASHmaster files. Diags, Progs & Ancills**. This will be a different STASHmaster file to the one that contains the UKCA tracers, and e.g. in the UKCA Tutorial job, is called **emiss_TCMIM_Aero.presm**.

This contains entries like

```
#
1|    1 |    0 |  301 |NOx surf emissions                  |
2|    2 |    0 |    1 |    1 |    5 |   -1 |   -1 |    0 |    0 |    0 |    0 |
3| 00000000000000000000000000000000 | 00000000000000000001 |    3 |
4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
5|    0 |  531 |    0 |  129 |    0 |    0 |    0 |    0 |    0 |
#

#
1|    1 |    0 |  340 |NOX AIRCRAFT EMS IN KG/S/GRIDCELL   |
2|    2 |    0 |    1 |    1 |    2 |   10 |   11 |    0 |    0 |    0 |    0 |
3| 00000000000000000000000000000000 | 00000000000000000001 |    3 |
4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
5|    0 |  520 |   20 |   65 |    0 |    0 |    0 | 9999 |    0 |
#
```

The top entry (*NOx surf emissions*) defines a 2D field, and the other (*NOX AIRCRAFT EMS IN KG/S/GRIDCELL*) defines a 3D field. You can see that there are differences in the numbers (other than the 301/340 item number) in various places in these specifications, which effectively (in this instance) tell STASH if the field is 2D or 3D.

Full details on what each of these numbers mean can be found in appendix 3 of *Unified Model Documentation Paper C4* which can be found on the Met Office Collaboration Twiki (password required) (http://collab.metoffice.gov.uk/twiki/bin/view/Support/Umdp) .

You should copy either the 2D or 3D specification, depending on what type of emission you are adding in, and edit only the **STASH item number**, **name of field**, and the **field code**. These can be found here:

```
#
1|    1 |    0 |  301 |NOx surf emissions                  |
2|    2 |    0 |    1 |    1 |    5 |   -1 |   -1 |    0 |    0 |    0 |
```

```
3| 00000000000000000000000000000000 | 00000000000000000001 |     3 |
4|     1 |     0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
5|     0 |   531 |     0 |   129 |     0 |     0 |     0 |     0 |     0 |
#
```

For the field code (531 above), if you were making a new emission at 316, it is advisable that you increase the current code above by 15 as well, e.g. *546*. Xconv uses this field code for the name of the variable, so it is best to have them different. A listing of available field codes can be found on the NCAS CMS website (http://cms.ncas.ac.uk/documents/IDL/@fcodes.txt) , although these may be a little out of date.

You should make these changes to your copy of the original emissions user STASHmaster file, save this, and then replace the original file in the UMUI's *Model Selection → Atmosphere → STASH → User-STASHmaster files. Diags, Progs & Ancills* table with your new file. Now go to **Model Selection → Atmosphere → STASH → Initialisation of User Prognostics** and scroll down the table until you find your new emission. Set the value in the **Option** column to **2** (*Initialise to User Ancillary File*).

As you need to make up a new ancillary file, you should copy this user STASHmaster file onto the supercomputer, as it will be needed by Xancil when you make the new ancillary file. It is advisable to put it in the same directory as the one containing your new emission(s) file.

Using Xancil

Extract your current emissions

Due to current limitations of the UM, you can only have one user single-level and one multi-level file. If you want to add a new emissions field, you must include the existing emissions in your new file along with it.

You can use Xconv to extract these fields. You must first find the location of the current file(s). To do this, go to **Model Selection → Atmosphere → Ancillary data and input data files → Climatologies and potential climatologies** and either open the **User multi-level ancillary file & fields** or **User single-level ancillary file & fields** panel. This will give two boxes giving the **directory name or environment variable** and the **file name**. This first box will contain an environment variable which sets the directory location. You can find the value of this environment variable in **Model Selection → Input/Output Control and Resources → Time Convention and SCRIPT Environment Variables**.

Once you have found the required file, extract all the fields within it to one of your own directories (ideally the one containing the netCDF file of your new emission(s)).

Make a new emissions ancillary file

Xancil is installed on both ARCHER at

```
/work/n02/n02/hum/bin/xancil
```

and on MONSooN (the postproc machine) at

```
/projects/um1/linux/bin/xancil
```

You may already have this location in your PATH and so can just launch Xancil from the command line by typing `xancil`. When you do this it will load up the Xancil window, which is initially rather empty. You should click on the **Xancil** text in the top-left corner. This will give 4 options:

- Configuration
- Atmosphere Ancillary Files
- Ocean Ancillary Files
- Generalised Ancillary Files

When using UKCA you will need to make use of all of these options, with the exception of the *Ocean Ancillary Files*.

You should view the *Using Xancil* page for more details on each of these sub-menus.

Load your netCDF and user STASHmaster files into the **Xancil → Configuration** panel, and define the vertical levels if you are making a 3D ancillary file.

Click on the *Atmosphere Ancillary Files* text and either open the **Multi-level User Fields** or **Single-level User Fields** panels. You should now

- Set the output file name
- Define the dates. If the ones in the netCDf file are fine to use then you can use them, or you can specify them. For this latter option you can either use the dates from the grid configuration panel, or you can define them again here
- Enter the number of ancillary fields. This will need to be the total of the number of fields in the original ancillary file, plus the number of new emissions you are adding
- For each individual field you should select it with the mouse, then
  - Define which netCDF file to use (it will remember the preference from the previous field)
  - Enter the STASH code (the PP code should be filled in automatically)
  - Either enter or confirm the variable name. Xancil takes the variable name from the *field code* defined in the STASHmaster file specification for each field. If there are multiple fields in the netCDF file with the same field code then Xancil may select the wrong one. If the name does not match a field code you will need to select it manually.

Once you have entered all the data for all fields you should

- Use the **Save/Save As** button to save the job, as it may be needed later
- Click the **Output Anc. Files** button to create the new ancillary file. Any errors will appear in the **Output messages** window, or to the terminal.

Use your new emissions file in the UMUI

Now that you have created your new emissions file you can use this in the UMUI. Go to **Model Selection → Atmosphere → Ancillary data and input data files → Climatologies and potential climatologies** and either open the **User multi-level ancillary file & fields** or **User single-level ancillary file & fields** panel, and set the **Directory name or Environment Variable** to the directory containing your new emissions file, and the **file name** to the name of your new file.

Note: On ARCHER this directory must be located on **/work** as the */home* directory cannot be read at run time. This includes any symbolic links from /work to /home and *vice versa*.

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

Please see this page for a solution to Task 5.1

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. No run-time processing of this surface emission is required. You will also need to add-in the molar-mass of **ALICE**.

Hint [hide]
You can calculate the molar mass from the mass of air and the conversion factor defined in Task 5.2.

Note: If you were unable to successfully complete Task 6.1 above, then please take a copy of the **e** job from the Tutorial experiment (*Tutorial: solution to Task 5.1 - adding a new chemical emissions ancillary file*) and work from there, as this will allow you to make only the changes required for this task. Please also make a new branch and merge-in branch **fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns** at revision number **14692** to allow you to proceed.

You can also find a copy of an emissions ancillary file, with the required emissions, at

```
/work/n02/n02/ukca/Tutorial/vn8.4/Task5.1/solution/Task5.1_AR5_2000.anc
```

on ARCHER, and at

```
~~/projects/ukca/Tutorial/vn8.4/Task5.1/solution/Task5.1_AR5_2000.anc~~
```

on MONSooN.

Part 2. UKCA Code Changes

Changes to ukca_setd1defs.F90

The ukca_setd1defs.F90 tells UKCA what fields it should expect to find from the UM to allow it to run. Previously you edited this routine so that UKCA knew about your new tracers, now you must edited to tell it about your new emissions. This is done in two places

Add the species to em_chem_spec

You will need to find the **em_chem_spec** definition for the scheme that you are using. For example, the *CheST/StratTrop* chemistry is located in the *Stratospheric Chemistry* section and contained in the IF block controlled by `(L_ukca_strattrop .AND. .NOT. L_ukca_achem)` if you are not using aerosol chemistry, and by `(L_ukca_strattrop .AND. L_ukca_achem)` if you are using aerosol chemistry. For the former case this is defined by

```
           n_chem_emissions = 9
           n_3d_emissions = 1       ! aircraft NOX

           ALLOCATE(em_chem_spec(n_chem_emissions+n_3d_emissions))
           em_chem_spec =                                          &
               (/'NO        ','CH4       ','CO        ','HCHO      ', &
                 'C2H6      ','C3H8      ','Me2CO     ','MeCHO     ', &
                 'C5H8      ','NO_aircrft'/)
```

and for the latter case by

```
           n_chem_emissions = 21      ! em_chem_spec below
           n_3d_emissions   = 2       ! volc SO2 & aircraft NOX
           ALLOCATE(em_chem_spec(n_chem_emissions+n_3d_emissions))
           em_chem_spec =                                          &
               (/'NO        ','CH4       ','CO        ','HCHO      ', &
                 'C2H6      ','C3H8      ','Me2CO     ','MeCHO     ', &
                 'C5H8      ','BC_fossil ','BC_biofuel','OC_fossil ', &
                 'OC_biofuel','Monoterp  ','NVOC      ','SO2_low   ', &
                 'SO2_high  ','DMS       ','NH3       ','SO2_nat   ', &
                 'BC_biomass','OC_biomass','NO_aircrft'/)
```

You should edit the equivalent block. You should first increase the value of **n_chem_emissions** (for surface emissions) or **n_3d_emissions** (for 3D emissions) by the number of emissions that you are adding, and you should then add the names of the species (as they appear in the **nm_spec** array in *ukca_setd1defs.F90*, and how they appear in the *CHCH_DEFS* specification in the *ukca_chem_scheme.F90* routine) that you are emitting into. Convention is that these are in ascending order by STASH code. The first 9 species in the above list are STASH codes 301-309, and `'NO_aircrft'` is STASH code 340, so all new species should be placed before `'NO_aircrft'` if you are not using aerosol chemistry, and before the `'SO2_low '` if you are. You should make sure that the size of *em_chem_spec* is correct for the number of species within it.

Tell UKCA the STASH code associated with your new emission

In the previous ukca_setd1defs.F90 discussion above we saw the following block of code

```
    J = n_use_tracers
    IF (n_chem_emissions+n_3d_emissions+n_mode_emissions > 0) THEN
      DO i=1,n_chem_emissions + n_3d_emissions
        UkcaD1Codes(J+i)%section    = 0
        UkcaD1Codes(J+i)%item       = n_emiss_first+i-1  ! trop chemistry
        UkcaD1Codes(J+i)%len_dim1   = row_length         ! uses stash codes
        UkcaD1Codes(J+i)%len_dim2   = rows               ! 301-309 for
        UkcaD1Codes(J+i)%required   = .true.             ! surface emissions
        UkcaD1Codes(J+i)%prognostic = .true.             ! from Section 0
! Special cases, emissions already available in UM
        IF (em_chem_spec(i)(1:7) == 'SO2_low') THEN
          UkcaD1Codes(J+i)%item     = 58
          IF (.NOT. L_SO2_SURFEM .AND. (L_ukca_aerchem .OR.           &
                                        L_ukca_achem)) THEN
            cmessage='SO2 surface emissions from UM are not flagged'
            errcode=58

            CALL EREPORT('UKCA_SETD1DEFS',errcode,cmessage)
          ENDIF
        ELSEIF (em_chem_spec(i)(1:7) == 'SO2_nat') THEN
          UkcaD1Codes(J+i)%item     = 121
          UkcaD1Codes(J+i)%len_dim3 = tr_levels
          IF (.NOT. L_SO2_NATEM .AND. (L_ukca_aerchem .OR.            &
                    L_ukca_achem)) THEN
            cmessage='SO2 natural emissions from UM are not flagged'
            errcode=121

            CALL EREPORT('UKCA_SETD1DEFS',errcode,cmessage)
          ENDIF
        ELSEIF (em_chem_spec(i)(1:8) == 'SO2_high') THEN
          UkcaD1Codes(J+i)%item     = 126
          IF (.NOT. L_SO2_HILEM .AND. (L_ukca_aerchem .OR.            &
                                       L_ukca_achem)) THEN
            cmessage='SO2 high-level emissions are not flagged'
            errcode = UkcaD1Codes(J+i)%item
            CALL EREPORT('UKCA_SETD1DEFS',errcode,cmessage)
          ENDIF
        ELSEIF (em_chem_spec(i)(1:3) == 'NH3') THEN
          UkcaD1Codes(J+i)%item     = 127
          IF (.NOT. L_NH3_EM .AND. (L_ukca_aerchem .OR.               &
                                    L_ukca_achem)) THEN
            cmessage='NH3 surface emissions from UM are not flagged'
            errcode = UkcaD1Codes(J+i)%item
            CALL EREPORT('UKCA_SETD1DEFS',errcode,cmessage)
          ENDIF
        ELSEIF (em_chem_spec(i) == 'BC_fossil ') THEN
          UkcaD1Codes(J+i)%item = 310
        ELSEIF (em_chem_spec(i) == 'BC_biofuel') THEN
          UkcaD1Codes(J+i)%item = 311
        ELSEIF (em_chem_spec(i) == 'OC_fossil ') THEN
          UkcaD1Codes(J+i)%item = 312
        ELSEIF (em_chem_spec(i) == 'OC_biofuel') THEN
          UkcaD1Codes(J+i)%item = 313
        ELSEIF (em_chem_spec(i) == 'Monoterp  ') THEN
          UkcaD1Codes(J+i)%item = 314
        ELSEIF (em_chem_spec(i) == 'NVOC      ') THEN
          UkcaD1Codes(J+i)%item = 315
        ELSEIF (em_chem_spec(i) == 'BC_biomass') THEN
          UkcaD1Codes(J+i)%item = 322
          UkcaD1Codes(J+i)%len_dim3 = tr_levels
        ELSEIF (em_chem_spec(i) == 'OC_biomass') THEN
          UkcaD1Codes(J+i)%item = 323
          UkcaD1Codes(J+i)%len_dim3 = tr_levels
        ELSEIF (em_chem_spec(i) == 'SO2_biomas') THEN
          UkcaD1Codes(J+i)%item = 324
          UkcaD1Codes(J+i)%len_dim3 = tr_levels
        ELSEIF (em_chem_spec(i)(1:3) == 'DMS') THEN
          UkcaD1Codes(J+i)%section  = 17
          UkcaD1Codes(J+i)%item     = 205
          UkcaD1Codes(J+i)%prognostic = .false.
          IF (.NOT. L_DMS_EM .AND. (L_ukca_aerchem .OR.               &
                    L_ukca_achem)) THEN
            cmessage='DMS surface emissions from UM are not flagged'
            errcode = UkcaD1Codes(J+i)%section*1000 +                  &
                      UkcaD1Codes(J+i)%item
            CALL EREPORT('UKCA_SETD1DEFS',errcode,cmessage)
          ENDIF
        ELSEIF (em_chem_spec(i)(1:7) == 'NO_airc') THEN
          UkcaD1Codes(J+i)%item     = 340
```

```
          UkcaD1Codes(J+i)%len_dim3 = tr_levels
        ENDIF
      ENDDO
    ENDIF
```

You will need to add code at the end of this block to tell UKCA what STASH code is associated with which species. The best thing to do is to copy one of the explict blocks at the end, and adjust accordingly, e.g.

For surface (2D) emissions you should add in the following

```
        ELSEIF (em_chem_spec(i) == 'Your Species') THEN
          UkcaD1Codes(J+i)%item = Your Species STASH code (301-320)
```

and for 3D emissions you should add in the following

```
        ELSEIF (em_chem_spec(i) == 'Your Species') THEN
          UkcaD1Codes(J+i)%item = Your Species STASH code (321-340)
          UkcaD1Codes(J+i)%len_dim3 = tr_levels
```

Remember that the character string length for a UKCA species is **10 characters**.

Changes to ukca_constants.F90

As you are adding in an emission you may need to define the molar mass (in g/mol) of the species that you are emitting in to, if it is not already defined. Add this definition into **ukca_constants.F90**, e.g.

```
    ...
    REAL, PARAMETER :: m_co      =  28.
    REAL, PARAMETER :: m_hcho    =  30.
    ...
```

Changes to ukca_emission_ctl.F90

You will need to ensure that the **molmass** array is filled for the tracer that you are emitting in to. This is filled in the **WHERE** block which has entries like

```
          ...
          WHERE (em_chem_spec == 'CO        ') molmass = m_co
          WHERE (em_chem_spec == 'HCHO      ') molmass = m_hcho
          ...
```

You should add an additional line for each new emission.

For surface emissions, unless your new emissions data requires further run-time processing, such as adding a diurnal cycle (see how isoprene (C5H8) is treated below), then you will not need to make too many additional changes to the **ukca_emission_ctl.F90** routine. If you do need to add functionality such as this then you will need edit the following IF block:

```
      DO l=1,n_emissions
        IF (advt(k) == em_chem_spec(l) .AND.                          &
          em_chem_spec(l) == 'NO      ' ) THEN
!         Convert from kg NO2/m2/s to kg NO/m2/s
          em_field(:,:,k) = emissions(:,:,l)*m_no/m_no2
        ELSE IF (advt(k) == em_chem_spec(l)(1:3) .AND.                 &
          em_chem_spec(l) == 'SO2_low ' ) THEN
!         Convert from kg S/m2/s to kg SO2/m2/s and take off sulphate fraction
          em_field(:,:,k) = emissions(:,:,l)*                          &
                            (1.0 - mode_parfrac/100.0)*m_so2/m_s
        ELSE IF (advt(k) == em_chem_spec(l) .AND.                      &
          em_chem_spec(l) == 'DMS     ' ) THEN
!         Convert from kg S/m2/s to kg DMS/m2/s
          em_field(:,:,k) = emissions(:,:,l)*m_dms/m_s
        ELSE IF (advt(k) == 'MeOH      ' .AND.                         &
          em_chem_spec(l) == 'NVOC      ' ) THEN
!         Convert from kg C/m2/s to kg CH3OH/m2/s
          em_field(:,:,k) = emissions(:,:,l)*meoh_factor*              &
              m_meoh/(m_c*3.0)
        ELSE IF (advt(k) == em_chem_spec(l) .AND.                      &
          em_chem_spec(l) == 'Monoterp  ' ) THEN
!         Convert from kg C/m2/s to kg C10H16/m2/s
          em_field(:,:,k) = emissions(:,:,l)*m_monoterp/(m_c*10.0)
!          === biogenic emissions ===
        ELSE IF (advt(k) == em_chem_spec(l) .AND.                      &
                 em_chem_spec(l) == 'C5H8      ') THEN
          IF (L_ukca_diurnal_isopems) THEN
            tmp_in_em_field(:,:) = emissions(:,:,l)*(m_isop/(5.0*m_c))
```

```
! DEPENDS ON: ukca_diurnal_isop_ems
!                testdcycl = .TRUE.
               CALL UKCA_DIURNAL_ISOP_EMS(row_length, rows,             &
                         tmp_in_em_field,  cos_zenith_angle,            &
                         int_zenith_angle,                              &
                         sin_theta_latitude, FV_cos_theta_latitude,     &
                         tan_theta_latitude, timestep, tmp_out_em_field,&
                         testdcycl)
               em_field(:,:,k) = tmp_out_em_field(:,:)
             ELSE
               em_field(:,:,k) = emissions(:,:,l)*(m_isop/(5.0*m_c))
             END IF
           ELSE IF (advt(k) == em_chem_spec(l) ) THEN
             em_field(:,:,k) = emissions(:,:,l)
           ENDIF             ! end advt(k)
```

If nothing needs to be done to the emissions field, then the final section

```
           ELSE IF (advt(k) == em_chem_spec(l) ) THEN
             em_field(:,:,k) = emissions(:,:,l)
```

just adds the emissions field from the ancillary file into the correct place to be emitted to the tracer.

After this is done the UM boundary layer mixing scheme is called (*TR_MIX*) for each tracer, and this is also where the emissions are added to the tracer field.

For 3D emissions you will need to explicitly add this into the required tracer using the *TRSRCE* subroutine. This is done for lightning emissions (calculated on-line) and aircraft emissions (read-in from the multi-level user ancillary file). You should copy what is done for e.g. aircraft emissions and adapt accordingly.

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

Please see this page for a solution to Task 5.2

Written by Luke Abraham 2014

Using Xancil

From UKCA

Xancil is a graphical package that is used to create Unfied Model ancillary files from netCDF files. Prior to using Xancil to create your files, you should ensure that the input netCDF files are on the correct grid (both horizontal and vertical) for the UM configuration that you are using.

Xancil is installed on ARCHER at

```
/work/n02/n02/hum/bin/xancil
```

and on MONSooN (on the postproc03 machine) at

```
/projects/um1/linux/bin/xancil
```

Contents

- 1 Xancil Settings
  - 1.1 Configuration
    - 1.1.1 General Configuration
    - 1.1.2 Grid Configuration
  - 1.2 Atmosphere Ancillary Files
  - 1.3 Generalised Ancillary Files
  - 1.4 Buttons

Xancil Settings

You may already have this location in your PATH and so can just launch Xancil from the command line by typing `xancil`. When you do this it will load up the Xancil window, which is initially rather empty. You should click on the **Xancil** text in the top-left corner. This will give 4 options:

- Configuration
- Atmosphere Ancillary Files
- Ocean Ancillary Files
- Generalised Ancillary Files

Figure 1 shows the Xancil GUI with some of the menus open. You will probably not need to make up any ocean files, as HadGEM3 uses the NEMO ocean, but you will need to make use of the Atmosphere and Generalised Ancillary files when using UKCA.

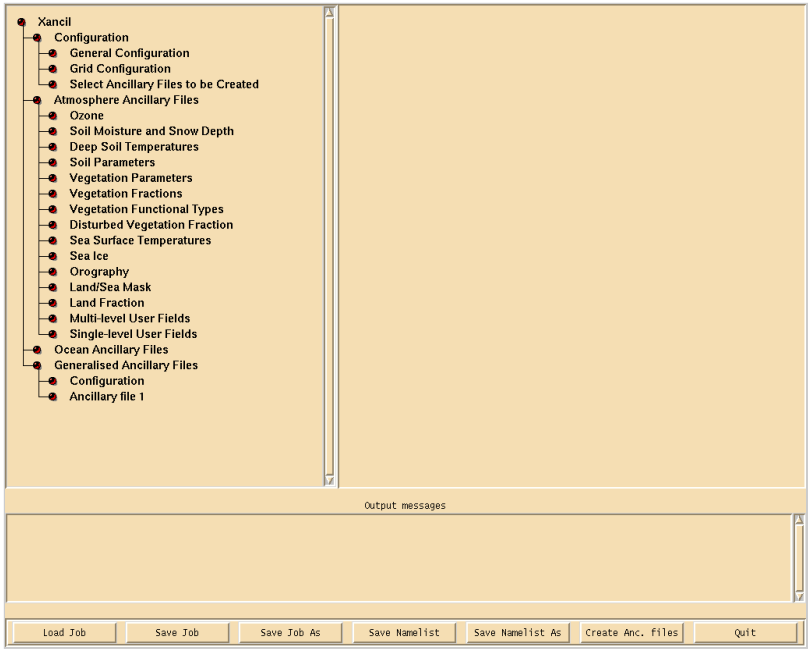


Figure 1: The Xancil GUI

Configuration

General Configuration

In this panel you need to set the UM **version number** for the ancillary file you are creating (e.g. 8.4 etc). Failiure to do this will cause an error when running (as the default is UM4.5).

If you are using nudging you will need to change the calendar, and you may also want to **output well-formed ancillary files**.

On this panel you should tell Xancil which netCDF files it should read (which contain the data you want to make into ancillary files) and which corresponding STASHmaster files it should use that describe this data.

Grid Configuration

This panel is used to define the vertical grid being used, as well as being able to set the times used in the output ancillary files (although the time information can also be set in the individual ancillary file panels).

The vertical settings are only important for 3D ancillary files (e.g. UKCA aircraft emissions, which are on the theta grid), but in this case you will need use the **specify atmosphere vertical levels**, which takes the vertical level structure from the UM namelist that contains this information. The location of this namelist is specified in the UMUI in **Model Selection → Atmosphere → Model Resolution and Domain → Vertical**.

If you made up your input netCDF file using Xconv, and then you try to use the option which uses the name of the vertical levels from the netCDF file you may get the following error:

```
ERROR: variable hybrid_ht from NetCDF file
name of netCDF input file
 doesn't contain the standard_name attribute needed to
 calculate vertical level values.
```

as Xconv does not output vertical levels which Xancil can read.

The ability to set the date and the number of times in the ancillary file is useful here, if needed.

Atmosphere Ancillary Files

This list gives the list of all the ancillary files that Xancil currently has defined settings for. We will not go through these in detail, but there are some options that you should be aware of.

At the top of each file panel will be the option **"Is field to output periodic in time?"** with *Yes* and *No* buttons. If you are making time-series ancillary file (e.g. data from 1960-2010) then the answer is **No**, but if you are making a climatology (e.g. 12 monthly values (Jan-Dec) or 4 seasonal values, or 1 annual value) then the answer is **Yes**.

You can use the **Select** button to choose the required field from the netCDF file. Usually Xancil will match up the name of the field in the file with the *field code* from the STASHmaster file, but if the name does not match (or matches a different field) then you will need to select this manually.

Generalised Ancillary Files

The *Atmosphere Ancillary Files* menu has only a sub-set of the number of ancillary files that you may wish to make up. If you want to make up a file that is not catered for you should use the *Generalised Ancillary Files* option.

There is feature of the *Generalised Ancillary Files* settings that you may need to be aware of. In the **Generalised Ancillary Files → Configuration** sub-menu you can select the number of different generalised ancillary files that you want to make (by default there is only one). If you make more than one, and then save your job using the **Save** button along the bottom of the GUI, when loading it again only the first file panel will appear. You will need to go back to the *Generalised Ancillary Files → Configuration* panel (which will show the correct number of files you wanted to make) increase the number by one, then decrease it back by one again. The full number of files (and their internal settings) will now be shown.

This panel is useful for making UKCA initial conditions.

Buttons

Using the buttons along the bottom of the GUI you can **Save** (or **Save As**) the current job you are working on or **Load** an existing *.job* file (as well as the namelists associated with the files). You should use the **Create Anc. Files** button to create the ancillary files when you are ready to do this.

Note: When you are using Xancil from scratch with your own STASHmaster file, Xancil will change the name in the Ancillary File panels (but not the file itself, when it is created) to that in your STASHmaster file. Another feature of Xancil is that often, when re-loading a saved .job file, the name of the fields defined in the user STASHmaster file has been lost, and instead it reverts to the name in the STASHmaster_A file. This does not affect the data in each field however.

When making the ancillary files, you will be given messages both to the **Output messages** window, and to the terminal.

Written by Luke Abraham 2013

Solution to UKCA Chemistry and Aerosol Tutorial 5 Task 5.1

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Back to the adding new chemical emissions tutorial

Contents

- 1 Task
- 2 Solution
  - 2.1 Regrid your emissions dataset
  - 2.2 Make your new STASH item
  - 2.3 Extract the original emissions data
  - 2.4 Make the new ancillary file
  - 2.5 Use your new Ancillary File in the UMUI
- 3 Output
- 4 Worked Solution

Task

You were asked to

> In the
>
> ```
> /work/n02/n02/ukca/Tutorial/vn8.4/Task5.1
> ```
>
> directory on ARCHER, or the
>
> ~~`/projects/ukca/Tutorial/vn8.4/Task5.1`~~
>
> directory on MONSooN, 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 **N96**, and then make a new surface emissions ancillary file with this as slot **316**. You should then use this new file, and the corresponding user STASHmaster file, in your UMUI job.

Solution

Before you start you should make a new directory to hold your files. For ease, this should be on the **/work** directory on ARCHER, and the **/projects** directory if you are using MONSooN.

Regrid your emissions dataset

Open the *Emissions_of_ALICE.nc* file using Xconv. Select the field (called "Emissions of ALICE in kg/m^2/s") and click on the **Trans** button on the top right of the GUI. You should then

1. Select **area weighted interpolation** (the second option down)
2. You should then scroll down this panel and
   1. Set the number of columns to 192
   2. Set the first longitude to 0.000000
   3. Set the column spacing to 1.875000
   4. Set the number of rows to 145
   5. Set the first latitude to -90.000000 (i.e. the South Pole)
   6. Set the row spacing to 1.250000

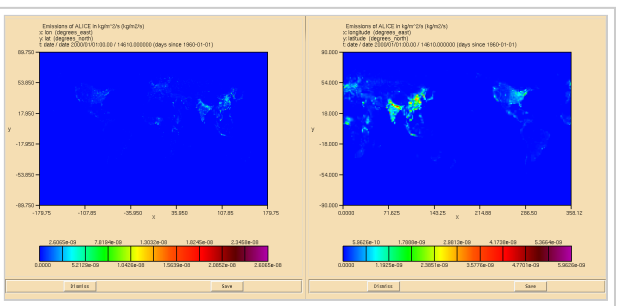


Figure 1: Comparison of the 0.5x0.5 degree data and the N96 regridded data. Note the shift in the grid introduced by changing the *first longitude* from -179.5 to 0.0.

Now click **Apply**. The window should read

```
Area weighted interpolation from 720x360 Regular grid to 192x1
```

and the **nx** and **ny** values should read 192 and 145. Now output this field to a netCDF file in your directory. A comparison of the 0.5x0.5 degree data and the N96 data can be seen in Figure 1.

An example of this file can be found at

```
/work/n02/n02/ukca/Tutorial/vn8.4/Task5.1/solution/Task5.1_Emissions_of_ALICE_N96.nc
```

on ARCHER and at

~~`/projects/ukca/Tutorial/vn8.4/Task5.1/solution/Task5.1_Emissions_of_ALICE_N96.nc`~~

on MONSooN.

Make your new STASH item

You are using the file

```
/home/ukca/userprestash/VN8.4/emiss_TCMIM_Aero.presm
```

on PUMA to define your emissions. Take a copy of this file and add the following entry just after field 315:

```
#
1|     1 |     0 |   316 |ALICE surf emissions                 |
2|     2 |     0 |     1 |     1 |     5 |    -1 |    -1 |     0 |     0 |     0 |
3| 00000000000000000000000000000000 | 00000000000000000001 |     3 |
4|     1 |     0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
5|     0 |   546 |     0 |   129 |     0 |     0 |     0 |     0 |     0 |
#
```

Save this file and use it in your job in **Model Selection → Atmosphere → STASH → User-STASHmaster files. Diags, Progs & Ancills**.

An example file can be found at

```
/home/ukca/userprestash/VN8.4/Tutorial/Task5.1_emiss_TCMIM_Aero.presm
```

Now go to **Model Selection → Atmosphere → STASH → Initialisation of User Prognostics** and set the **Option** column for this field to **2**.

Copy this new STASHmaster file into the directory on the supercomputer that contains your regridded emissions data.

Extract the original emissions data

Open the original emissions ancillary file. This can be found at

```
/work/n02/n02/ukca/ANCILS/N96L85/AR5_2000_MEGAN-MeOH
```

on ARCHER and at

```
~~/projects/ukca/inputs/ancil/N96L85/emiss/AR5_2000_MEGAN-MeOH~~
```

on MONSooN. Extract this data (using Xconv) to a new netCDF file in the same directory as your regridded data.

Make the new ancillary file

In your directory on the supercomputer containing the two emissions netCDF files and your new user STASHmaster file load up Xancil. You should just be able to type

```
xancil
```

on the command-line, but the full path is

```
/work/n02/n02/hum/bin/xancil
```

on ARCHER and

```
/projects/um1/linux/bin/xancil
```

on the MONSooN postproc.

On loading this up, go to the **Xancil → Configuration → General Configuration** and set

- The UM version to **8.4**
- Set to output **well-formed ancillary files**
- Input the netCDF files and STASHmaster files that you have created

Now go to **Xancil → Atmosphere Ancillary Files → Single-level User Fields**

- Set to **create** this file
- Set the **name** to one of your choosing
- Set to **specify the file dates**
- Increase the number of **ancillary fields** to **16**
- For each of the 16 fields you will need to
    - Set the **netCDF file**. For the first 15 this will be the file containing the fields extracted from *AR5_2000_MEGAN-MeOH*, and for field 16 this will be your regridded ALICE emissions field
    - Set the **STASH code** to **301-316** for each of the 16 fields
    - **Select** the **variable name**. You should do (or at least confirm) this manually as Xconv may put the wrong field in a STASH slot

You can now **Save** this if you wish. Now click **Create Anc. Files** and your new ancillary file will be created.

An example ancillary file can be found at

```
/work/n02/n02/ukca/Tutorial/vn8.4/Task5.1/solution/Task5.1_AR5_2000.anc
```

on ARCHER and at

```
~~/projects/ukca/Tutorial/vn8.4/Task5.1/solution/Task5.1_AR5_2000.anc~~
```

on MONSooN. A corresponding Xancil job file can be found at

```
/work/n02/n02/ukca/Tutorial/vn8.4/Task5.1/solution/Task5.1_Xancil.job
```

on ARCHER and at

```
~~/projects/ukca/Tutorial/vn8.4/Task5.1/solution/Task5.1_Xancil.job~~
```

on MONSooN.

Use your new Ancillary File in the UMUI

Go to **Model Selection → Atmosphere → Ancillary and input data files → Climatologies & potential climatologies → User single-level ancillary file & fields** and set the **Directory name or Environment Variable** box to the directory containing the new ancillary file, and set the name in the **file name** box to the file name of your file.

Output

If you open the **pa** file in your job directory you will find that it still contains the fields

```
0    : 192   145   85    1     Stash code = 34001
1    : 192   145   85    1     Stash code = 34064
2    : 192   145   85    1     Stash code = 34065
```

If you open 34064 and 34065 you will find that they still only contain zeros. This will change in the next task as we allow UKCA to take the emissions from this ancillary file and add them to the ALICE tracer.

Sample output from this job can be found in

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Task5.1/
```

on ARCHER, and in

```
~~/projects/ukca/Tutorial/vn8.4/sample_ouput/Task5.1/~~
```

on MONSooN.

Worked Solution

There is a worked solution to this problem in the UMUI Tutorial experiment. This is job **e**: *Tutorial: solution to Task 5.1 - adding a new chemical emissions ancillary file*.

Written by Luke Abraham 2014

Solution to UKCA Chemistry and Aerosol Tutorial 5 Task 5.2

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Back to the adding new chemical emissions tutorial

Contents

- 1 Task
- 2 Solution
  - 2.1 Code changes to ukca_setd1defs.F90
    - 2.1.1 em_chem_spec changes
    - 2.1.2 STASH changes
  - 2.2 Code changes to ukca_constants.F90
  - 2.3 Code changes to ukca_emission_ctl.F90
- 3 Output
- 4 Worked Solution

Task

You were asked to

> You should now make the UKCA code changes to add your emission into the **ALICE** tracer. No run-time processing of this surface emission is required.

Solution

As no run-time processing of this field is required, the only changes that need to be made are those in **ukca_setd1defs.F90**

Code changes to ukca_setd1defs.F90

em_chem_spec changes

In **ukca_setd1defs.F90** you should scroll-down to the code block controlled by the

```
        ELSE IF (L_ukca_strattrop .AND. L_ukca_achem) THEN
```

statement and

- **increase** the value of **n_chem_emissions** to **22**.
- Edit **em_chem_spec** so that it now includes **ALICE**:

```
               (/'NO        ','CH4       ','CO        ','HCHO      ', &
                 'C2H6      ','C3H8      ','Me2CO     ','MeCHO     ', &
                 'C5H8      ','BC_fossil ','BC_biofuel','OC_fossil ', &
                 'OC_biofuel','Monoterp  ','NVOC      ','ALICE     ', &
                 'SO2_low   ', &
                 'SO2_high  ','DMS       ','NH3       ','SO2_nat   ', &
                 'BC_biomass','OC_biomass','NO_aircrft'/)
```

STASH changes

Edit the IF block controlling which STASH codes are assigned to which species, and add the following

```
        ELSEIF (em_chem_spec(i) == 'ALICE     ') THEN
          UkcaD1Codes(J+i)%item = 316
```

Code changes to ukca_constants.F90

You need to add the following line to define the molar mass of ALICE

```
    REAL, PARAMETER :: M_ALICE      = 28.97
```

Code changes to ukca_emission_ctl.F90

You need to add the following line to the WHERE block adding in the molar masses of the emitted species (defined in *em_chem_spec*) to the **molmass** array

```
           WHERE (em_chem_spec == 'ALICE     ') molmass = M_ALICE
```

Output

As the ALICE (and BOB) tracers are being output as daily means to the pb/UPB stream, you can examine the output (located in your **archive** directory) to see if the emission is being correctly applied. Opening the **pa** file there are 3 fields:

```
0    : 192   145   85    1     Stash code = 34001
1    : 192   145   85    1     Stash code = 34064
2    : 192   145   85    1     Stash code = 34065
```

The first (34001) is ozone, and 34065 is BOB (which will still contain zeros). View the surface of 34064 and you should now see that it is non-zero. An example of this is shown in Figure 1.

You should note that this will not match up exactly with the emission field in the ancillary file, partly because of the time interpolation that is done between the monthly emission fields, and partly because the field in the tracer is a daily mean of a tracer that has these emissions applied every timestep (and currently with no loss processes), and which is also undergoing boundary layer mixing to mix concentrations out of the surface layer.

Sample output from this job can be found in

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Task5.2/
```

on ARCHER, and in

```
~~/projects/ukca/Tutorial/vn8.4/sample_ouput/Task5.2/~~
```

on MONSooN.

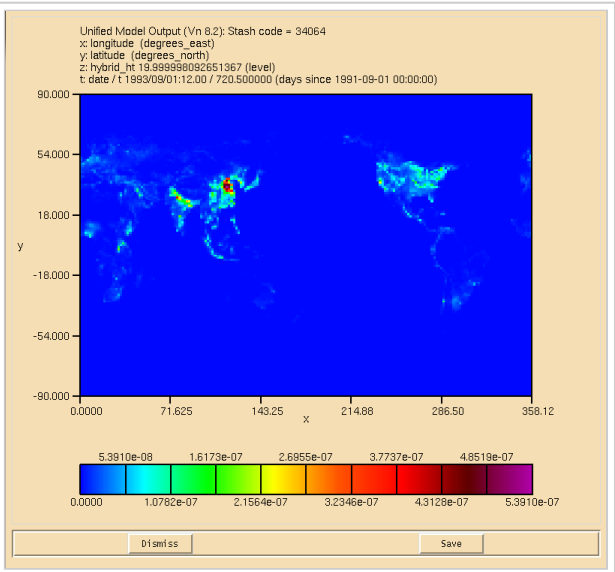


Figure 1: Surface plot of the ALICE tracer after emissions have been applied.

Worked Solution

There is a worked solution to this problem in the UMUI Tutorial experiment. This is job **f**: *Tutorial: solution to Task 5.2 - adding new chemical emissions in UKCA*.

The code changes can be viewed by using the following FCM command

```
fcm diff fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns@14692 fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns@14697
```

This gives the following (non-graphical) output:

```
Index: src/atmosphere/UKCA/ukca_setd1defs.F90
===================================================================
--- src/atmosphere/UKCA/ukca_setd1defs.F90      (revision 14692)
+++ src/atmosphere/UKCA/ukca_setd1defs.F90      (revision 14697)
@@ -273,14 +273,15 @@
              nr_phot        = 55         ! photolytic (ATA)

           ELSE IF (L_ukca_strattrop .AND. L_ukca_achem) THEN
-             n_chem_emissions = 21      ! em_chem_spec below
+             n_chem_emissions = 22      ! em_chem_spec below
              n_3d_emissions   = 2       ! volc SO2 & aircraft NOX
              ALLOCATE(em_chem_spec(n_chem_emissions+n_3d_emissions))
              em_chem_spec =                                          &
                  (/'NO        ','CH4       ','CO        ','HCHO      ', &
                    'C2H6      ','C3H8      ','Me2CO     ','MeCHO     ', &
                    'C5H8      ','BC_fossil ','BC_biofuel','OC_fossil ', &
-                   'OC_biofuel','Monoterp  ','NVOC      ','SO2_low   ', &
+                   'OC_biofuel','Monoterp  ','NVOC      ','ALICE     ', &
+                   'SO2_low   ', &
                    'SO2_high  ','DMS       ','NH3       ','SO2_nat   ', &
                    'BC_biomass','OC_biomass','NO_aircrft'/)
              n_aero_tracers = 12
@@ -624,6 +625,8 @@
             UkcaD1Codes(J+i)%item = 314
           ELSEIF (em_chem_spec(i) == 'NVOC      ') THEN
             UkcaD1Codes(J+i)%item = 315
+          ELSEIF (em_chem_spec(i) == 'ALICE     ') THEN
+            UkcaD1Codes(J+i)%item = 316
           ELSEIF (em_chem_spec(i) == 'BC_biomass') THEN
             UkcaD1Codes(J+i)%item = 322
             UkcaD1Codes(J+i)%len_dim3 = tr_levels
Index: src/atmosphere/UKCA/ukca_constants.F90
===================================================================
--- src/atmosphere/UKCA/ukca_constants.F90      (revision 14692)
+++ src/atmosphere/UKCA/ukca_constants.F90      (revision 14697)
```

```
@@ -385,6 +385,11 @@
       REAL, PARAMETER :: m_alkaooh  = 90.0
       REAL, PARAMETER :: m_aromooh  = 130.0
       REAL, PARAMETER :: m_mekooh   = 104.0
+
+!     UKCA Tutorial ALICE tracer - same  as mass of air in g/mol
+!     as the value of C_ALICE=1.0000 (above)
+!     Required for ukca_emission_ctl
+      REAL, PARAMETER :: M_ALICE      = 28.97

 !     The mass of organic nitrate is an approximation,
 !     calculated as the average of ORGNIT formed by two
Index: src/atmosphere/UKCA/ukca_emission_ctl.F90
===================================================================
--- src/atmosphere/UKCA/ukca_emission_ctl.F90   (revision 14692)
+++ src/atmosphere/UKCA/ukca_emission_ctl.F90   (revision 14697)
@@ -380,6 +380,8 @@
             WHERE (em_chem_spec == 'OC_fossil ') molmass = m_c
             WHERE (em_chem_spec == 'OC_biofuel') molmass = m_c
             WHERE (em_chem_spec == 'OC_biomass') molmass = m_c
+! UKCA Tutorial Tracer
+            WHERE (em_chem_spec == 'ALICE     ') molmass = M_ALICE

 ! Check if all the emitted species have a valid molecular weight
             IF (ANY(molmass(:) < 0.00001)) THEN
```

Written by Luke Abraham 2014

UKCA Chemistry and Aerosol Tutorial 6

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Contents

- 1 What you will learn in this tutorial
- 2 Task 6.1: Add a bimolecular reaction
- 3 Adding new Chemical Reactions
- 4 Biomolecular Reactions
  - 4.1 Bimolecular Reaction Definition
  - 4.2 Increase the size of JPBK (and JPNR)
- 5 Termolecular Reactions
  - 5.1 Termolecular Reaction Definition
  - 5.2 Increase the size of JPTK (and JPNR)
- 6 Heterogeneous Reactions
  - 6.1 Heterogeneous Reaction Definition
  - 6.2 Increase the size of JPHK (and JPNR)
- 7 Photolysis Reactions
  - 7.1 Tropospheric Off-Line Photolysis
  - 7.2 Stratospheric Look-Up Table Photolysis
  - 7.3 Interactive Photolysis
  - 7.4 Photolysis Reaction Definition
  - 7.5 Increase the size of JPPJ (and JPNR)
- 8 Solution to Task 6.1: Add a bimolecular reaction

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

Note: If you were unable to successfully complete Task 5.2, then please take a copy of the **f** job from the Tutorial experiment (*Tutorial: solution to Task 5.2 - adding new chemical emissions in UKCA*) and work from there, as this will allow you to only make the required changes. Please also make a new branch and merge-in branch **fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns/src** at revision number **14697** to allow you to proceed.

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 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

```
vn8.4_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_scheme.F90** routines using the **ratb_t** Fortran type specification, and are held in arrays. At the end of this routine the **ratb_defs_scheme** 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_t** type is

```
ratb_t('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), &
```

If fractional products are not required for a reaction, then the *fraction of each product* formed should be set to 0.000. 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 specifications of the individual reactions are done as, e.g.

```
ratb_t('O3        ','C5H8      ','HO2       ','OH        ','          ',& ! B133
'          ',  3.33E-15,  0.00,   1995.00, 0.750, 0.750, 0.000, 0.000), & ! B133 IUPAC2007*
...
ratb_t('OH        ','C5H8      ','ISO2      ','          ','          ',& ! B144
'          ',  2.70E-11,  0.00,   -390.00, 0.000, 0.000, 0.000, 0.000), & ! B144 IUPAC2009
...
ratb_t('OH        ','HCl       ','H2O       ','Cl        ','          ',& ! B159
'          ',  1.80E-12,  0.00,    250.00, 0.000, 0.000, 0.000, 0.000), & ! B159 JPL2011
```

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 reaction above takes its 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) . When adding new reactions you will need to increment the size of the array holding the `ratb_t` type.

To add new bimolecular reactions you will need to append equivalent lines for the new reactions to the end of the **ratb_defs_scheme** 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.

Increase the size of JPBK (and JPNR)

As well as adding these reactions to the *ukca_chem_scheme.F90* routine (and incrementing the size of the arrays in that routine accordingly, you will also need to increase the values of two parameters that UKCA needs. These are

- **JPBK** is the number of bimolecular reactions
- **JPNR** is the total number of reactions

These are set automatically in the UMUI (depending on what scheme is chosen), and are placed in the `&RUN_UKCA` namelist in **CNTLATM**. You will need to make a hand-edit to change these accordingly. The current values can be found by saving and processing the job, and then viewing the *CNTLATM* file in your `$HOME/umui_jobs/jobid` 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

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

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_scheme.F90** routines using the **ratt_t** Fortran type specification, and are usually held in one single array (there are not usually enough reactions to require splitting the reactions over several arrays).

To format of this **ratt_t** type is

```
ratt_t('Reactant 1','Reactant 2','Product 1 ','Product 2 ', f, &
k1,  α1,  β1, k2,  α2,  β2, Fraction of Product 1 produced, Fraction of Product 2 produced), &
```

and as in `ratb_t`, where the fraction of a product should be set to 0.000 if this functionality does not need to be used.

The f value is used to define the F_c value by

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

as F_c may or may not be highly temperature dependent.

Examples of these reactions are

```
ratt_t('N2O5      ','m          ','NO2       ','NO3       ',     0.3,    & ! T023
  1.30E-03, -3.50, 11000.00,  9.70E+14,  0.10, 11080.00, 0.000, 0.000), & ! T023 IUPAC 2002
ratt_t('NO        ','NO         ','NO2       ','NO2       ',     0.0,    & ! T024
  3.30E-39,  0.00,  -530.00,  0.00E+00,  0.00,     0.00, 0.000, 0.000)  & ! T024 IUPAC 2001
```

To add new termolecular reactions you will need to append equivalent lines for the new reactions to the end of the **ratt_defs_scheme** array (increasing the array sizes accordingly).

Increase the size of JPTK (and JPNR)

As with the bimolecular reactions, you will also need to increase the values of two parameters that UKCA needs. These are

- **JPTK** is the number of termolecular reactions
- **JPNR** is the total number of reactions

These are set automatically in the UMUI (depending on what scheme is chosen), and are placed in the `&RUN_UKCA` namelist in **CNTLATM**. You will need to make a hand-edit to change these accordingly. The current values can be found by saving and processing the job, and then viewing the *CNTLATM* file in your `$HOME/umui_jobs/jobid` 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_scheme.F90** routines using the **rath_t** Fortran type specification, usually in one array. To format of this **rath_t** type is

```
rath_t('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), &
```

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**. Examples of this type are

```
rath_t('ClONO2    ','H2O       ','HOCl      ','HONO2     ','          ', &
'          ', 0.000, 0.000, 0.000, 0.000), &
...
rath_t('SO2       ','H2O2      ','NULL0     ','          ','          ', & !HSO3+H2O2(aq)
'          ', 0.000, 0.000, 0.000, 0.000),                                 &
```

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

Increase the size of JPHK (and JPNR)

As with the bimolecular and termolecular reactions, you will also need to increase the values of two parameters that UKCA needs. These are

- **JPHK** is the number of heterogeneous reactions
- **JPNR** is the total number of reactions

These are set automatically in the UMUI (depending on what scheme is chosen), and are placed in the `&RUN_UKCA` namelist in **CNTLATM**. You will need to make a hand-edit to change these accordingly. The current values can be found by saving and processing the job, and then viewing the *CNTLATM* file in your `$HOME/umui_jobs/jobid` directory.

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 separate photolysis codes, **Fast-J** or **Fast-JX**, are used to interactively calculate the rate of reaction throughout the troposphere (for Fast-J) or 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 UM 8.4 these files (in ASCII format) can be found in

```
/work/n02/n02/hum/vn8.4/ctldata/UKCA/tropdata/photol
```

on ARCHER, and in

```
/projects/um1/vn8.4/ctldata/UKCA/tropdata/photol
```

on MONSooN. To use this scheme, in the UMUI go to **Model Selection → Atmosphere → Model Configuration → UKCA Chemistry and Aerosols → PHOTO** and click **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 interactive photolysis should be used instead. For the *CheS/Strat* or *CheST/StratTrop* schemes, Fast-JX should be used as this covers the stratosphere as well as the troposphere.

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, in the UMUI go to **Model Selection → Atmosphere → Model Configuration → UKCA Chemistry and Aerosols → PHOTO** and click **2D Photolysis Scheme**. The code for this scheme is held in **ukca_photolib.F90**.

References

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

Interactive Photolysis

The Fast-J scheme (Wild *et al*, 2000)[1] uses 7 different wavelength bins appropriate for the troposphere, and the Fast-JX scheme (Neu et al, 2007)[2] adds up to an extra 11 bins allowing use in the stratosphere.

To use these schemes, in the UMUI go to **Model Selection → Atmosphere → Model Configuration → UKCA Chemistry and Aerosols → PHOTO** and click either **FASTJ Photolysis Scheme** or **FASTJX Photolysis Scheme**. You will then need to give the location of several input data files used by these schemes. The code for Fast-J is in the `UKCA/` directory in the **fastj_*.F90** files (controlled by **ukca_fastj.F90**), and the code for Fast-JX is in the **fastjx_*.F90** files (controlled by **ukca_fastjx.F90**).

Further details on the Fast-JX scheme, and how it 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-J/Fast-JX data files are held in

```
/work/n02/n02/hum/vn8.4/ctldata/UKCA/fastj
```

on ARCHER, and

```
/projects/um1/vn8.4/ctldata/UKCA/fastj
```

on MONSooN.

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_scheme.F90** routines using the **ratj_t** Fortran type specification, usually in several arrays. To format of this **ratj_t** type is

```
ratj_t('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), &
```

The **Look-Up Label** is used to define the file used for the 2D photolysis, and is used by Fast-J/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.

Examples of this type are

```
ratj_t('H2O2      ','PHOTON    ','OH        ','OH        ','          ', &
     '          ',    0.0,   0.0,   0.0,   0.0, 100.000,'jh2o2     ') ,  &
ratj_t('HCHO      ','PHOTON    ','HO2       ','HO2       ','CO        ', &
     '          ',    0.0,   0.0,   0.0,   0.0, 100.000,'jhchoa    ') ,  &
```

Increase the size of JPPJ (and JPNR)

As with the bimolecular, termolecular, and heterogeneous reactions, you will also need to increase the values of two parameters that UKCA needs. These are

- **JPPJ** is the number of photolysis reactions
- **JPNR** is the total number of reactions

These are set automatically in the UMUI (depending on what scheme is chosen), and are placed in the `&RUN_UKCA` namelist in **CNTLATM**. You will need to make a hand-edit to change these accordingly. The current values can be found by saving and processing the job, and then viewing the *CNTLATM* file in your `$HOME/umui_jobs/jobid` directory.

Solution to Task 6.1: Add a bimolecular reaction

Please see this page for a solution to Task 6.1.

Written by Luke Abraham 2014

Solution to UKCA Chemistry and Aerosol Tutorial 6 Task 6.1

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Back to the adding new chemical reactions tutorial

Contents

- 1 Task
- 2 Solution
  - 2.1 Changes to ukca_chem_strattrop.F90
  - 2.2 Hand-edit to increase JPBK and JPNR
- 3 Output
- 4 Worked Solution

Task

You were asked to

> 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 |

Solution

Changes to ukca_chem_strattrop.F90

The only UKCA code changes that are required, are in **ukca_chem_strattrop.F90**. You should add the following line at the end of the **ratb_defs_strattrop05** array

```
ratb_t('ALICE     ','OH        ','BOB       ','Sec_Org   ','          ',&
'          ',  2.70E-11,  0.00,   -390.00, 0.000, 0.000, 0.000, 0.000)  &
```

(remembering to add a comma at the end of the line above) and increase the size of the **ratb_defs_strattrop05** array to **21** and the size of the **ratb_defs_strattrop_chem** master array to **201**.

Hand-edit to increase JPBK and JPNR

You will now need to make a hand-edit to increase the size of **JPBK** and **JPNR**, which are set in **CNTLATM**. This should look like

```
ed CNTLATM<<\EOF
/JPBK/
d
i
 JPBK=216,
.
/JPNR/
d
i
 JPNR=309,
.
w
q
EOF
```

You should make this script executable (`chmod a+rx script.ed`) and add it to the UMUI in the **Model Selection → Input/Output Control and Resource → User hand edit files** by placing it in the table and putting a **Y** in the **Include Y/N** column.

An example hand-edit can be found at

```
/home/ukca/hand_edits/VN8.4/Tutorial/Task6.1_incr_JPvals.ed
```

on PUMA.

Output

Now that we are forming BOB, this tracer field will be non-zero. If you open the **pb** file (located in your **archive** directory) in Xconv and plot the surface of 34065, you should see a field similar to that in Figure 1.

Sample output from this job can be found in

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Task6.1/
```

on ARCHER, and in

```
~~/projects/ukca/Tutorial/vn8.4/sample_ouput/Task6.1/~~
```

on MONSooN.

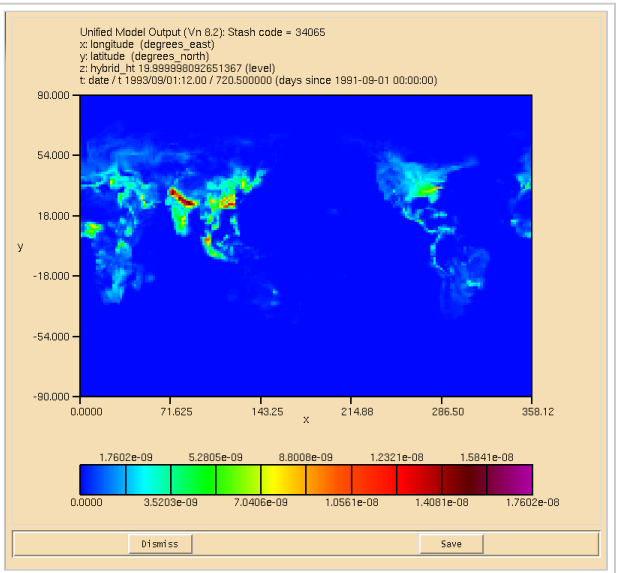


Figure 1: Surface plot of the BOB tracer after bimolecular reaction has been applied.

Worked Solution

There is a worked solution to this problem in the UMUI Tutorial experiment. This is job **g**: *Tutorial: solution to Task 6.1 - add a new bimolecular reaction*.

The code changes can be viewed by using the following FCM command

```
fcm diff fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns@14697 fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns@14703
```

This gives the following output:

```
Index: src/atmosphere/UKCA/ukca_chem_strattrop.F90
===================================================================
--- src/atmosphere/UKCA/ukca_chem_strattrop.F90 (revision 14697)
+++ src/atmosphere/UKCA/ukca_chem_strattrop.F90 (revision 14703)
@@ -399,7 +399,7 @@
 chch_t( 90,'BOB       ',  1,'TR        ','          ',  0,  0,  0)   &
   /)

-TYPE(RATB_T) :: ratb_defs_strattrop_chem(200)
+TYPE(RATB_T) :: ratb_defs_strattrop_chem(201)

 ! reactions found in either Trop or Strat but not both
 TYPE(RATB_T), PARAMETER :: ratb_defs_strattrop_aer(1:15)=(/              &
@@ -1011,7 +1011,7 @@
 TYPE(RATB_T) :: ratb_defs_strattrop02(45)
 TYPE(RATB_T) :: ratb_defs_strattrop03(45)
 TYPE(RATB_T) :: ratb_defs_strattrop04(45)
-TYPE(RATB_T) :: ratb_defs_strattrop05(20)
+TYPE(RATB_T) :: ratb_defs_strattrop05(21)

 REAL :: depvel_defs_strattrop01(360)
 REAL :: depvel_defs_strattrop02(360)
@@ -1649,7 +1649,9 @@
 ratb_t('ISO2      ','O2        ','OH        ','HCHO      ','MACR      ',& ! B198
 '          ',  4.00E-03,  0.00,      0.00, 0.000, 0.000, 0.000, 0.000) ,& ! B198  ATA HOx recyc - added by hand
 ratb_t('ISO2      ','N2        ','HO2       ','MGLY      ','HACET     ',& ! B199
-'OH        ',  8.00E-02,  0.00,      0.00, 1.000, 1.000, 0.650, 0.600) &  ! B199  ATA HOx recyc - added by hand
+-'OH        ',  8.00E-02,  0.00,      0.00, 1.000, 1.000, 0.650, 0.600) &  ! B199  ATA HOx recyc - added by hand
++'OH        ',  8.00E-02,  0.00,      0.00, 1.000, 1.000, 0.650, 0.600) ,& ! B199  ATA HOx recyc - added by hand
++ratb_t('ALICE     ','OH        ','BOB       ','Sec_Org   ','          ',& ! UKCA TUTORIAL RXN 01
++'          ',  2.70E-11,  0.00,   -390.00, 0.000, 0.000, 0.000, 0.000)  & ! UKCA TUTORIAL RXN 01
   /)

 !-----------------------------------------------------------------------
```

Written by Luke Abraham 2014

- This page was last modified on 15 December 2015, at 15:33.

UKCA Chemistry and Aerosol Tutorial 7

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Contents

- 1 What you will learn in this Tutorial
- 2 Task 7.1: adding new dry deposition values
- 3 Adding Dry Deposition
- 4 Chemistry Scheme Specification
- 5 2D Dry Deposition Scheme
- 6 Interactive Dry Deposition Scheme
  - 6.1 Changes to ukca_aerod.F90
  - 6.2 Changes to ukca_surfddr.F90
- 7 Increase the value of JPDD
- 8 Solution to Task 7.1: adding new dry deposition values

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.

Note: If you were unable to successfully complete Task 6.1, then please take a copy of the **g** job from the Tutorial experiment (*Tutorial: solution to Task 6.1 - add a new bimolecular reaction*) and work from there, as this will allow you to only make the changes required for this task. Please also make a new branch and merge-in branch **fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns/src** at revision number **14703** to allow you to proceed.

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 2D scheme. To choose the interactive dry deposition scheme in the UMUI, go to **Model Selection → Atmosphere → Model Configuration → UKCA Chemistry and Aerosols → COUPL** and select **UKCA interactive dry deposition scheme**.

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. As the 2D scheme is the default, it must always have values set.

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_scheme.F90** file. In the **chch_defs_scheme** array there are lines like

```
chch_t( 10,'HONO2     ',  1,'TR        ','          ',  1,  1,  0),  & !  10 DD: 7,WD: 4,
chch_t( 11,'H2O2      ',  1,'TR        ','          ',  1,  1,  0),  & !  11 DD: 8,WD: 5,
```

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_scheme** array, which is held in the **ukca_chem_scheme.F90** module. This is a large array made up of size (6,5) blocks. These blocks mean

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

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

```
!  1  O3 (Ganzeveld & Lelieveld (1995) note 1 (modified to same as Guang)
  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
!  2  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
```

Note: When adding new deposition values you should be careful. **UKCA assumes that the order of this array is the same as the order of the species in the chch_defs_scheme array.** If you are adding values for a species in the middle of the list then you will need to make sure that you slot it in to the appropriate place in the existing **depvel_defs_scheme** array (and change the size of this array accordingly). Also note that the dry deposition of the species associated with the aerosol chemistry is held in a separate array which is treated slightly differently. You may find that you need to change the order of the tracers in the `chch_defs` array to accomodate this.

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 documentation paper for vn8.4 of the MetUM (//www.ukca.ac.uk/wiki/index.php/File:Umdp_084-umdp84.pdf).

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 different surface types:

1. Broadleaved trees
2. Needleleaf trees
3. C3 Grass
4. C4 Grass
5. Shrub
6. Urban
7. Water
8. Bare Soil
9. Ice

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)
```

Increase the value of JPDD

When you added a new chemical reaction you needed to increment a counter which gave the number of reactions, when adding new dry deposition of a species you will need to increase the size of the **JPDD** counter. This is done with a hand-edit, the value of **JPDD** being set in the **CNTLATM** file in your `$HOME/umui_jobs/jobid` directory.

Solution to Task 7.1: adding new dry deposition values

Please see this page for a solution to Task 7.1.

Written by Luke Abraham 2014

Solution to UKCA Chemistry and Aerosol Tutorial 7 Task 7.1

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Back to the adding dry deposition tutorial

Contents

- 1 Task
- 2 Solution
  - 2.1 Changes to ukca_chem_strattrop.F90
    - 2.1.1 chch_defs array
      - 2.1.1.1 Re-order tracer list
    - 2.1.2 Additions for depvel_defs
  - 2.2 Changes to ukca_aerod.F90
  - 2.3 Changes to ukca_surfddr.F90
  - 2.4 Hand-edit to increase the value of JPDD
- 3 Output
- 4 Worked Solution

Task

You were asked to

> 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.

and 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.

Solution

You will need to make changes to 3 UKCA routines and make 1 hand-edit.

Changes to ukca_chem_strattrop.F90

chch_defs array

First, you will need to change the **0** in the 6th column of the `chch_defs_strattrop_aer` array to **1**:

```
!  88
chch_t( 88,'ALICE     ',  1,'TR        ','          ',  1,  0,  0),  &
```

Re-order tracer list

However, you may also need to change the order of the tracers in the chch_defs array. This is because of the fact that the chemical species required for the aerosol chemistry are dealt with slightly differently, and are included using the `depvel_defs_strattrop_aer` array (of size `ndry_st_aer`). There are two options:

1. Put the dry deposition of ALICE into the array dealing with the aerosol chemistry
2. Re-order the tracers so that the dry deposition of ALICE can be put in the array dealing with the main CheST/StratTrop chemistry

Either way would work, but if you wanted to turn off the aerosol chemistry at any point you would need to make further changes later. In this solution we will re-order the tracer list so that it now looks like this:

```
!   1
chch_t(  1,'O(3P)     ',  1,'TR        ','Ox        ',  0,  0,  0),  &
!   2
chch_t(  2,'O(1D)     ',  1,'SS        ','Ox        ',  0,  0,  0),  &
!   3 DD: 1,WD: 1,
chch_t(  3,'O3        ',  1,'TR        ','Ox        ',  1,  1,  0),  &
!   4
chch_t(  4,'N         ',  1,'TR        ','NOx       ',  0,  0,  0),  &
```

```
!   5 DD: 2,
chch_t(  5,'NO        ',  1,'TR        ','NOx       ',  1,  0,  0),  &
!   6 DD: 3,WD: 2,
chch_t(  6,'NO3       ',  1,'TR        ','NOx       ',  1,  1,  0),  &
!   7 DD: 4,      EM: 1
chch_t(  7,'NO2       ',  1,'TR        ','NOx       ',  1,  0,  1),  &
!   8 DD: 5,WD: 3,
chch_t(  8,'N2O5      ',  1,'TR        ','          ',  1,  1,  0),  &
!   9 DD: 6,WD: 4,
chch_t(  9,'HO2NO2    ',  1,'TR        ','          ',  1,  1,  0),  &
!  10 DD: 7,WD: 5,
chch_t( 10,'HONO2     ',  1,'TR        ','          ',  1,  1,  0),  &
!  11 DD: 8,WD: 6,
chch_t( 11,'H2O2      ',  1,'TR        ','          ',  1,  1,  0),  &
!  12 EM: 2
chch_t( 12,'CH4       ',  1,'TR        ','          ',  0,  0,  2),  &
!  13 DD: 9,      EM:  3
chch_t( 13,'CO        ',  1,'TR        ','          ',  1,  0,  3),  &
!  14 DD:10,WD: 7,EM:  4
chch_t( 14,'HCHO      ',  1,'TR        ','          ',  1,  1,  4),  &
!  15       WD: 8,
chch_t( 15,'MeOO      ',  1,'TR        ','          ',  0,  1,  0),  &
!  16 DD:11,WD: 9,
chch_t( 16,'MeOOH     ',  1,'TR        ','          ',  1,  1,  0),  &
!  17
chch_t( 17,'H         ',  1,'TR        ','HOx       ',  0,  0,  0),  &
!  18
chch_t( 18,'H2O       ',  1,'TR        ','          ',  0,  0,  0),  &
!  19
chch_t( 19,'OH        ',  1,'TR        ','HOx       ',  0,  0,  0),  &
!  20       WD:10,
chch_t( 20,'HO2       ',  1,'TR        ','HOx       ',  0,  1,  0),  &
!  21
chch_t( 21,'Cl        ',  1,'TR        ','Clx       ',  0,  0,  0),  &
!  22
chch_t( 22,'Cl2O2     ',  1,'TR        ','Clx       ',  0,  0,  0),  &
!  23
chch_t( 23,'ClO       ',  1,'TR        ','Clx       ',  0,  0,  0),  &
!  24
chch_t( 24,'OClO      ',  1,'TR        ','          ',  0,  0,  0),  &
!  25
chch_t( 25,'Br        ',  1,'TR        ','Brx       ',  0,  0,  0),  &
!  26
chch_t( 26,'BrO       ',  1,'TR        ','Brx       ',  0,  0,  0),  &
!  27
chch_t( 27,'BrCl      ',  1,'TR        ','          ',  0,  0,  0),  &
!  28       WD:11,
chch_t( 28,'BrONO2    ',  1,'TR        ','          ',  0,  1,  0),  &
!  29
chch_t( 29,'N2O       ',  1,'TR        ','          ',  0,  0,  0),  &
!  30 DD:12,WD:12,
chch_t( 30,'HCl       ',  1,'TR        ','          ',  1,  1,  0),  &
!  31 DD:13,WD:13,
chch_t( 31,'HOCl      ',  1,'TR        ','          ',  1,  1,  0),  &
!  32 DD:14,WD:14,
chch_t( 32,'HBr       ',  1,'TR        ','          ',  1,  1,  0),  &
!  33 DD:15,WD:15,
chch_t( 33,'HOBr      ',  1,'TR        ','          ',  1,  1,  0),  &
!  34       WD:16,
chch_t( 34,'ClONO2    ',  1,'TR        ','          ',  0,  1,  0),  &
!  35
chch_t( 35,'CFCl3     ',  1,'TR        ','          ',  0,  0,  0),  &
!  36
chch_t( 36,'CF2Cl2    ',  1,'TR        ','          ',  0,  0,  0),  &
!  37
chch_t( 37,'MeBr      ',  1,'TR        ','          ',  0,  0,  0),  &
!  38 DD:16,WD:17,
chch_t( 38,'HONO      ',  1,'TR        ','          ',  1,  1,  0),  &
!  39             EM: 5
chch_t( 39,'C2H6      ',  1,'TR        ','          ',  0,  0,  1),  &
!  40
chch_t( 40,'EtOO      ',  1,'TR        ','          ',  0,  0,  0),  &
!  41 DD:17,WD:18,
chch_t( 41,'EtOOH     ',  1,'TR        ','          ',  1,  1,  0),  &
!  42 DD:18,      EM: 6
chch_t( 42,'MeCHO     ',  1,'TR        ','          ',  1,  0,  1),  &
!  43
chch_t( 43,'MeCO3     ',  1,'TR        ','          ',  0,  0,  0),  &
!  44 DD:19,
chch_t( 44,'PAN       ',  1,'TR        ','          ',  1,  0,  0),  &
```

```
!  45             EM: 7
chch_t( 45,'C3H8       ',  1,'TR        ','          ',  0,  0,  1),  &
!  46
chch_t( 46,'n-PrOO     ',  1,'TR        ','          ',  0,  0,  0),  &
!  47
chch_t( 47,'i-PrOO     ',  1,'TR        ','          ',  0,  0,  0),  &
!  48 DD:20,WD:19,
chch_t( 48,'n-PrOOH    ',  1,'TR        ','          ',  1,  1,  0),  &
!  49 DD:21,WD:20,
chch_t( 49,'i-PrOOH    ',  1,'TR        ','          ',  1,  1,  0),  &
!  50 DD:22,
chch_t( 50,'EtCHO      ',  1,'TR        ','          ',  1,  0,  0),  &
!  51
chch_t( 51,'EtCO3      ',  1,'TR        ','          ',  0,  0,  0),  &
!  52             EM: 8
chch_t( 52,'Me2CO      ',  1,'TR        ','          ',  0,  0,  1),  &
!  53
chch_t( 53,'MeCOCH2OO  ',  1,'TR        ','          ',  0,  0,  0),  &
!  54 DD:23,WD:21,
chch_t( 54,'MeCOCH2OOH',  1,'TR        ','          ',  1,  1,  0),  &
!  55 DD:24,
chch_t( 55,'PPAN       ',  1,'TR        ','          ',  1,  0,  0),  &
!  56
chch_t( 56,'MeONO2     ',  1,'TR        ','          ',  0,  0,  0),  &
!  57             EM: 9
chch_t( 57,'C5H8       ',  1,'TR        ','          ',  0,  0,  1),  &
!  58
chch_t( 58,'ISO2       ',  1,'TR        ','          ',  0,  0,  0),  &
!  59 DD:25,WD:22,
chch_t( 59,'ISOOH      ',  1,'TR        ','          ',  1,  1,  0),  &
!  60 DD:26,WD:23,
chch_t( 60,'ISON       ',  1,'TR        ','          ',  1,  1,  0),  &
!  61 DD:27,
chch_t( 61,'MACR       ',  1,'TR        ','          ',  1,  0,  0),  &
!  62
chch_t( 62,'MACRO2     ',  1,'TR        ','          ',  0,  0,  0),  &
!  63 DD:28,WD:24,
chch_t( 63,'MACROOH    ',  1,'TR        ','          ',  1,  1,  0),  &
!  64 DD:29,
chch_t( 64,'MPAN       ',  1,'TR        ','          ',  1,  0,  0),  &
!  65 DD:30,WD:25,
chch_t( 65,'HACET      ',  1,'TR        ','          ',  1,  1,  0),  &
!  66 DD:31,WD:26,
chch_t( 66,'MGLY       ',  1,'TR        ','          ',  1,  1,  0),  &
!  67 DD:32,
chch_t( 67,'NALD       ',  1,'TR        ','          ',  1,  0,  0),  &
!  68 DD:33,WD:27,
chch_t( 68,'HCOOH      ',  1,'TR        ','          ',  1,  1,  0),  &
!  69 DD:34,WD:28,
chch_t( 69,'MeCO3H     ',  1,'TR        ','          ',  1,  1,  0),  &
!  70 DD:35,WD:29,
chch_t( 70,'MeCO2H     ',  1,'TR        ','          ',  1,  1,  0),  &
!  71
chch_t( 71,'H2         ',  1,'TR        ','          ',  0,  0,  0),  &
!  72 DD:36,WD:30,
chch_t( 72,'MeOH       ',  1,'TR        ','          ',  1,  1,  0),  &
!  73
chch_t( 73,'CO2        ',  1,'CT        ','          ',  0,  0,  0),  &
!  74
chch_t( 74,'O2         ',  1,'CT        ','          ',  0,  0,  0),  &
!  75
chch_t( 75,'N2         ',  1,'CT        ','          ',  0,  0,  0),  &
!  76 DD:37
chch_t( 76,'ALICE      ',  1,'TR        ','          ',  1,  0,  0),  &
!  77 WD:31
chch_t( 77,'BOB        ',  1,'TR        ','          ',  0,  1,  0),  &
!  78
chch_t( 78,'DMS        ',  1,'TR        ','          ',  0,  0,  0),  &
!  79 DD:38,WD:32,EM:10
chch_t( 79,'SO2        ',  1,'TR        ','          ',  1,  1,  1),  &
!  80
chch_t( 80,'H2SO4      ',  1,'TR        ','          ',  0,  0,  0),  &
!  81
chch_t( 81,'MSA        ',  1,'TR        ','          ',  0,  0,  0),  &
!  82 DD:39,WD:33
chch_t( 82,'DMSO       ',  1,'TR        ','          ',  1,  1,  0),  &
!  83 DD:40,WD:34,EM:11
chch_t( 83,'NH3        ',  1,'TR        ','          ',  1,  1,  1),  &
!  84
chch_t( 84,'CS2        ',  1,'TR        ','          ',  0,  0,  0),  &
```

```
!  85
chch_t( 85,'COS       ',  1,'TR        ','          ',  0,  0,  0),  &
!  86
chch_t( 86,'H2S       ',  1,'TR        ','          ',  0,  0,  0),  &
!  87
chch_t( 87,'SO3       ',  1,'TR        ','          ',  0,  0,  0),  &
!  88 DD:41,      EM: 12
chch_t( 88,'Monoterp  ',  1,'TR        ','          ',  1,  0,  1),  &
!  89 DD:42,WD:35
chch_t( 89,'Sec_Org   ',  1,'TR        ','          ',  1,  1,  0)   &
```

Additions for depvel_defs

To add in the values for the deposition velocity into the **depvel_defs_strattrop** array you will need to

- Increase the value of **ndry_strattrop** from 36 to **37**
- Increase the size of **depvel_defs_strattrop03** from 360 to **390**
- Add this code block to end of the **depvel_defs_strattrop03** array (remembering to add a comma to end of the line preceding this)

```
  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  & !     37.1
  0.03,  0.03,  0.03,  0.03,  0.03,  0.03,  & !     37.2
  0.03,  0.03,  0.03,  0.03,  0.03,  0.03,  & !     37.3
  0.03,  0.03,  0.03,  0.03,  0.03,  0.03,  & !     37.4
  0.00,  0.00,  0.00,  0.00,  0.00,  0.00   & !     37.5
```

Changes to ukca_aerod.F90

You will need to add the following block to the **CASE** statement within this routine

```
           CASE ('ALICE   ')
             d0(j) = d_h2o * SQRT(m_h2o / m_ALICE)
```

the value of m_ALICE having been set in Task 5.2 - adding new emissions to UKCA.

Changes to ukca_surfddr.F90

As has been noted in the *hint* for this task, the only change required to **ukca_surfddr.F90** is to add `'ALICE '` to the **CO** section of the **CASE** statement:

```
           CASE ('CO        ','ALICE     ')
       rsurf(:,n)=(/3700.,7300.,4550.,1960.,4550.0,r_null,r_null,     &
         4550.0,r_null /)  ! Shrub+bare soil set to C3 grass (guess)
```

Hand-edit to increase the value of JPDD

You will need to add a hand-edit to UMUI in the **Model Selection → Input/Output Control and Resource → User hand edit files** by placing it in the table and putting a **Y** in the **Include Y/N** column.

It should contain the following:

```
ed CNTLATM<<\EOF
/JPDD/
d
i
 JPDD = 42,
.
w
q
EOF
```

An example hand-edit can be found at

```
/home/ukca/hand_edits/VN8.4/Tutorial/Task7.1_incr_JPvals.ed
```

Output

If you view the ALICE (34064) field in the **pb** file (located in your **archive** directory) after these changes, and compare it to the equivalent field after the previous Task (6.1) you will see that there are some difference. However, we will not be able to quantify these differences correctly until Tutorial 9: Adding new UKCA diagnostics.

Sample output from this job can be found in

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Task7.1/
```

on ARCHER, and in

```
~~/projects/ukca/Tutorial/vn8.4/sample_ouput/Task7.1/~~
```

on MONSooN.

Worked Solution

There is a worked solution to this problem in the UMUI Tutorial experiment. This is job **h**: *Tutorial: solution to Task 7.1 - add new dry deposition* .

The code changes can be viewed by using the following FCM command

```
fcm diff fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns@14703 fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns@14711
```

This gives the following output:

```
Index: src/atmosphere/UKCA/ukca_surfddr.F90
===================================================================
--- src/atmosphere/UKCA/ukca_surfddr.F90        (revision 14703)
+++ src/atmosphere/UKCA/ukca_surfddr.F90        (revision 14711)
@@ -277,7 +277,7 @@
           500.0, 12500. /)
             CASE ('NH3       ')
         rsurf(:,n)=hno3
-            CASE ('CO        ')
+            CASE ('CO        ','ALICE     ')
         rsurf(:,n)=(/3700.,7300.,4550.,1960.,4550.0,r_null,r_null,     &
           4550.0,r_null /)  ! Shrub+bare soil set to C3 grass (guess)
             CASE ('CH4       ')
Index: src/atmosphere/UKCA/ukca_chem_strattrop.F90
===================================================================
--- src/atmosphere/UKCA/ukca_chem_strattrop.F90 (revision 14703)
+++ src/atmosphere/UKCA/ukca_chem_strattrop.F90 (revision 14711)
@@ -56,7 +56,7 @@
 INTEGER, PARAMETER, PUBLIC :: nhet_st_tpht   = 2         ! trophet rxns

 ! No of dry deposited species
-INTEGER, PARAMETER, PUBLIC :: ndry_strattrop = 36        ! Stratos chemistry
+INTEGER, PARAMETER, PUBLIC :: ndry_strattrop = 37        ! Stratos chemistry
 INTEGER, PARAMETER, PUBLIC :: ndry_st_aer    = 5         ! Aerosol chemistry

 ! No of wet deposited species
@@ -369,34 +369,34 @@
 chch_t( 74,'O2        ',  1,'CT        ','          ',  0,  0,  0),  &
 !  75
 chch_t( 75,'N2        ',  1,'CT        ','          ',  0,  0,  0),  &
-!  76
-chch_t( 76,'DMS       ',  1,'TR        ','          ',  0,  0,  0),  &
-!  77 DD:37,WD:31,EM:10
-chch_t( 77,'SO2       ',  1,'TR        ','          ',  1,  1,  1),  &
+!  76 DD:37
+chch_t( 76,'ALICE     ',  1,'TR        ','          ',  1,  0,  0),  &
+!  77
+chch_t( 77,'BOB       ',  1,'TR        ','          ',  0,  0,  0),  &
 !  78
-chch_t( 78,'H2SO4     ',  1,'TR        ','          ',  0,  0,  0),  &
-!  79
-chch_t( 79,'MSA       ',  1,'TR        ','          ',  0,  0,  0),  &
-!  80 DD:38,WD:32
-chch_t( 80,'DMSO      ',  1,'TR        ','          ',  1,  1,  0),  &
-!  81 DD:39,WD:33,EM:11
-chch_t( 81,'NH3       ',  1,'TR        ','          ',  1,  1,  1),  &
-!  82
-chch_t( 82,'CS2       ',  1,'TR        ','          ',  0,  0,  0),  &
-!  83
-chch_t( 83,'COS       ',  1,'TR        ','          ',  0,  0,  0),  &
+chch_t( 78,'DMS       ',  1,'TR        ','          ',  0,  0,  0),  &
+!  79 DD:38,WD:31,EM:10
+chch_t( 79,'SO2       ',  1,'TR        ','          ',  1,  1,  1),  &
+!  80
+chch_t( 80,'H2SO4     ',  1,'TR        ','          ',  0,  0,  0),  &
+!  81
+chch_t( 81,'MSA       ',  1,'TR        ','          ',  0,  0,  0),  &
+!  82 DD:39,WD:32
+chch_t( 82,'DMSO      ',  1,'TR        ','          ',  1,  1,  0),  &
```

```
+!  83 DD:40,WD:33,EM:11
+chch_t( 83,'NH3       ',  1,'TR        ','          ',  1,  1,  1),  &
 !  84
-chch_t( 84,'H2S       ',  1,'TR        ','          ',  0,  0,  0),  &
+chch_t( 84,'CS2       ',  1,'TR        ','          ',  0,  0,  0),  &
 !  85
-chch_t( 85,'SO3       ',  1,'TR        ','          ',  0,  0,  0),  &
-!  86 DD:40,      EM: 12
-chch_t( 86,'Monoterp  ',  1,'TR        ','          ',  1,  0,  1),  &
-!  87 DD:41,WD:34
-chch_t( 87,'Sec_Org   ',  1,'TR        ','          ',  1,  1,  0),  &
-!  88
-chch_t( 88,'ALICE     ',  1,'TR        ','          ',  0,  0,  0),  &
-!  89
-chch_t( 90,'BOB       ',  1,'TR        ','          ',  0,  0,  0)   &
+chch_t( 85,'COS       ',  1,'TR        ','          ',  0,  0,  0),  &
+!  86
+chch_t( 86,'H2S       ',  1,'TR        ','          ',  0,  0,  0),  &
+!  87
+chch_t( 87,'SO3       ',  1,'TR        ','          ',  0,  0,  0),  &
+!  88 DD:41,      EM: 12
+chch_t( 88,'Monoterp  ',  1,'TR        ','          ',  1,  0,  1),  &
+!  89 DD:42,WD:34
+chch_t( 89,'Sec_Org   ',  1,'TR        ','          ',  1,  1,  0)   &
   /)

 TYPE(RATB_T) :: ratb_defs_strattrop_chem(201)
@@ -1015,7 +1015,7 @@

 REAL :: depvel_defs_strattrop01(360)
 REAL :: depvel_defs_strattrop02(360)
-REAL :: depvel_defs_strattrop03(360)
+REAL :: depvel_defs_strattrop03(390)

 INTEGER(KIND=jpim), PARAMETER :: zhook_in  = 0
 INTEGER(KIND=jpim), PARAMETER :: zhook_out = 1
@@ -2579,7 +2579,13 @@
   0.83,  0.04,  0.44,  0.06,  0.05,  0.06,  & !     36.2
   0.63,  0.06,  0.35,  0.08,  0.06,  0.07,  & !     36.3
   0.03,  0.03,  0.03,  0.03,  0.03,  0.03,  & !     36.4
-  0.01,  0.01,  0.01,  0.01,  0.01,  0.01   & !     36.5
+  0.01,  0.01,  0.01,  0.01,  0.01,  0.01   & !     36.5
+  0.01,  0.01,  0.01,  0.01,  0.01,  0.01,  & !     36.5
+! 42 ALICE as CO (see Giannokopoulos (1998))
+  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  & !     37.1
+  0.03,  0.03,  0.03,  0.03,  0.03,  0.03,  & !     37.2
+  0.03,  0.03,  0.03,  0.03,  0.03,  0.03,  & !     37.3
+  0.03,  0.03,  0.03,  0.03,  0.03,  0.03,  & !     37.4
+  0.00,  0.00,  0.00,  0.00,  0.00,  0.00   & !     37.5
   /)


Index: src/atmosphere/UKCA/ukca_aerod.F90
===================================================================
--- src/atmosphere/UKCA/ukca_aerod.F90  (revision 14703)
+++ src/atmosphere/UKCA/ukca_aerod.F90  (revision 14711)
@@ -219,6 +219,8 @@
               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
 !
```

Written by Luke Abraham 2014

UKCA Chemistry and Aerosol Tutorial 8

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Contents

- 1 What you will learn in this Tutorial
- 2 Task 8.1: Add wet deposition of a species
- 3 Adding Wet Deposition
- 4 Turning on Wet Deposition for a Species
  - 4.1 Chemistry Scheme Specification
  - 4.2 Setting Henry's Law values
  - 4.3 Increase the value of JPDW
- 5 Solution to Task 8.1: Add wet deposition of a species

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

Note: If you were unable to successfully complete Task 7.1, then please take a copy of the **h** job from the Tutorial experiment (*Tutorial: solution to Task 7.1 - add new dry deposition*) and work from there, as this will allow you to only make the changes required for this task. Please also make a new branch and merge-in branch **fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns/src** at revision number **14711** to allow you to proceed.

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_scheme.F90** file. In the **chch_defs_scheme** array there are lines like

```
chch_t( 10,'HONO2     ',  1,'TR        ','          ',  1,  1,  0),  & !  10 DD: 7,WD: 4,
chch_t( 11,'H2O2      ',  1,'TR        ','          ',  1,  1,  0),  & !  11 DD: 8,WD: 5,
```

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_scheme.F90** the parameters required to calculate H_{eff} are held in the **henry_defs_scheme** array, and has format

$k(298)$	$-(\Delta H/R)$	$k(298)$ for the 1st dissociation	$-(\Delta H/R)$ for the 1st dissociation	$k(298)$ for the 2nd dissociation	$-(\Delta H/R)$ for the 2nd dissociation

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_scheme**, the order of **henry_defs_scheme** also assumes that the values are in the same order as the species (that wet deposit) in the **chch_defs_scheme** array.

Examples for this array are

```
0.2100E+06, 0.8700E+04, 0.2000E+02, 0.0000E+00, 0.0000E+00, 0.0000E+00,&    !     4  HONO2
0.8300E+05, 0.7400E+04, 0.2400E-11,-0.3730E+04, 0.0000E+00, 0.0000E+00,&    !     5  H2O2
```

Increase the value of JPDW

Similar to when adding dry deposition of a species you will need to increase the size of the **JPDW** counter. This is done with a hand-edit, the value of **JPDW** being set in the **CNTLATM** file in your `$HOME/umui_jobs/jobid` directory.

Solution to Task 8.1: Add wet deposition of a species

Please see this page for a solution of Task 8.1.

Written by Luke Abraham 2014

Solution to UKCA Chemistry and Aerosol Tutorial 8 Task 8.1

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Back to the adding wet deposition tutorial

Contents

- 1 Task
- 2 Solution
  - 2.1 Changes to ukca_chem_strattrop.F90
  - 2.2 Hand-edit
- 3 Output
- 4 Worked Solution

Task

You were asked to

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

Solution

Changes to ukca_chem_strattrop.F90

First, you will need to increase the value of **nwet_st_aer** from 34 to **35**.

Then, you will need to change the 0 in the 7th column of the **BOB** entry in the **chch_defs_strattrop_aer** array to **1**, e.g.

```
!  77
chch_t( 77,'BOB       ',  1,'TR        ','          ',  0,  1,  0),  &
```

Finally, you will need to add the following to the end of the **henry_defs_strattrop_aer** array

```
!   31  BOB
 0.2100E+06, 0.8700E+04, 0.2000E+02, 0.0000E+00, 0.0000E+00, 0.0000E+00,&
```

(there is a separate and complete array for when using just the CheST/StratTrop chemistry, *henry_defs_strattrop_chem*)

Hand-edit

You will need to make a hand-edit containing

```
ed CNTLATM<<\EOF
/JPDW/
d
i
 JPDW = 35,
.
w
q
EOF
```

and add this to the UMUI in the **Model Selection → Input/Output Control and Resource → User hand edit files** by placing it in the table and putting a **Y** in the **Include Y/N** column.

An example hand-edit can be found at

```
/home/ukca/hand_edits/VN8.4/Tutorial/Task8.1_incr_JPvals.ed
```

Output

If you view the BOB (34065) field in the **pa** file (located in your **archive** directory) after these changes, and compare it to the equivalent field after the previous Task (7.1) you will see that there are some differences. However, we will not be able to quantify these differences correctly until Tutorial 9: Adding new UKCA diagnostics.

Sample output from this job can be found in

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Task8.1/
```

on ARCHER, and in

```
~~/projects/ukca/Tutorial/vn8.4/sample_ouput/Task8.1/~~
```

on MONSooN.

Worked Solution

There is a worked solution to this problem in the UMUI Tutorial experiment. This is job **i**: *Tutorial: solution to Task 8.1 - add new wet deposition* .

The code changes can be viewed by using the following FCM command

```
fcm diff fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns@14711 fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns@14713
```

This gives the following output:

```
Index: src/atmosphere/UKCA/ukca_chem_strattrop.F90
===================================================================
--- src/atmosphere/UKCA/ukca_chem_strattrop.F90 (revision 14711)
+++ src/atmosphere/UKCA/ukca_chem_strattrop.F90 (revision 14713)
@@ -61,7 +61,7 @@

 ! No of wet deposited species
 INTEGER, PARAMETER, PUBLIC :: nwet_strattrop = 29        ! Stratos chemistry
-INTEGER, PARAMETER, PUBLIC :: nwet_st_aer    = 34        ! Aerosol chemistry
+INTEGER, PARAMETER, PUBLIC :: nwet_st_aer    = 35        ! Aerosol chemistry


 ! ATA NLA CheST Chemistry v1.2
@@ -371,19 +371,19 @@
 chch_t( 75,'N2        ',  1,'CT        ','          ',  0,  0,  0),  &
 !  76 DD:37
 chch_t( 76,'ALICE     ',  1,'TR        ','          ',  1,  0,  0),  &
-!  77
-chch_t( 77,'BOB       ',  1,'TR        ','          ',  0,  0,  0),  &
+!  77 WD:31
+chch_t( 77,'BOB       ',  1,'TR        ','          ',  0,  1,  0),  &
 !  78
 chch_t( 78,'DMS       ',  1,'TR        ','          ',  0,  0,  0),  &
-!  79 DD:38,WD:31,EM:10
+!  79 DD:38,WD:32,EM:10
 chch_t( 79,'SO2       ',  1,'TR        ','          ',  1,  1,  1),  &
 !  80
 chch_t( 80,'H2SO4     ',  1,'TR        ','          ',  0,  0,  0),  &
 !  81
 chch_t( 81,'MSA       ',  1,'TR        ','          ',  0,  0,  0),  &
-!  82 DD:39,WD:32
+!  82 DD:39,WD:33
 chch_t( 82,'DMSO      ',  1,'TR        ','          ',  1,  1,  0),  &
-!  83 DD:40,WD:33,EM:11
+!  83 DD:40,WD:34,EM:11
 chch_t( 83,'NH3       ',  1,'TR        ','          ',  1,  1,  1),  &
 !  84
 chch_t( 84,'CS2       ',  1,'TR        ','          ',  0,  0,  0),  &
@@ -395,7 +395,7 @@
 !  88 DD:41,      EM: 12
 chch_t( 88,'Monoterp  ',  1,'TR        ','          ',  1,  0,  1),  &
-!  89 DD:42,WD:34
+!  89 DD:42,WD:35
 chch_t( 89,'Sec_Org   ',  1,'TR        ','          ',  1,  1,  0)   &
   /)

@@ -975,13 +975,15 @@
  0.4700E+04, 0.6000E+04, 0.1800E-04, 0.0000E+00, 0.0000E+00, 0.0000E+00,&
 !   30  MeOH
  0.2300E+03, 0.4900E+04, 0.0000E+00, 0.0000E+00, 0.0000E+00, 0.0000E+00,&
-!   31  SO2
```

```
+!   31  BOB
+ 0.2100E+06, 0.8700E+04, 0.2000E+02, 0.0000E+00, 0.0000E+00, 0.0000E+00,&
+!   32  SO2
  0.1230E+01, 0.3020E+04, 0.1230E-01, 0.2010E+04, 0.6000E-07, 0.1120E+04,&
-!   32  DMSO
+!   33  DMSO
  0.5000E+05, 0.6425E+04, 0.0000E+00, 0.0000E+00, 0.0000E+00, 0.0000E+00,&
-!   33  NH3 (H*)
+!   34  NH3 (H*)
  0.1000E+07, 0.0000E+00, 0.0000E+00, 0.0000E+00, 0.0000E+00, 0.0000E+00,&
-!   34  SEC_ORG
+!   35  SEC_ORG
  0.1000E+06, 0.1200E+02, 0.0000E-00, 0.0000E+00, 0.0000E+00, 0.0000E+00 &
   /),(/  6,  nwet_st_aer/) )
```

Written by Luke Abraham 2014

UKCA Chemistry and Aerosol Tutorial 9

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Contents

- 1 What you will learn in this Tutorial
- 2 Task 9.1: Output new diagnostics
- 3 Adding New UKCA Diagnostics
- 4 Flux Definitions in asad_flux_dat.F90
  - 4.1 Diagnostic Type
  - 4.2 STASH Code
  - 4.3 Diagnostic Specification
  - 4.4 Mask
  - 4.5 Reaction number
  - 4.6 Number of Species
  - 4.7 Species
  - 4.8 Reactants and Products
  - 4.9 Addition of Diagnostics
  - 4.10 Changes to asad_flux_dat.F90
- 5 STASHmaster file
  - 5.1 2D Diagnostic Fields
  - 5.2 3D Diagnostic Fields
  - 5.3 STASHmaster file format
  - 5.4 UMUI Changes
- 6 Time Validity
  - 6.1 Why should I use these numbers?
- 7 Solution to Task 9.1: Output new diagnostics

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 $\mathrm{ALICE + OH \longrightarrow BOB + SEC\_ORG}$ to STASH code **34461**, the dry deposition of **ALICE** to STASH code **34462**, and the wet deposition of **BOB** to **34463**. They should be outputted as a daily mean to the **pa/UPA** stream.

Hint	[hide]
Remember to use the correct sampling frequency.	

Note: If you were unable to successfully complete Task 8.1 , then please take a copy of the **i** job from the Tutorial experiment (*Tutorial: solution to Task 8.1 - add new wet deposition*) and work from there, as this will allow you to only make the changes required for this task. Please also make a new branch and merge-in branch **fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns/src** at revision number **14713** to allow you to proceed.

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 spec
     (/'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)
- **EMS** to output the emission flux into 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)
- **RTE** to output the rate of a reaction (in gridcell/s)

STASH Code

This is a **5 digit integer** defining the STASH code that the diagnostic will be outputted to (e.g. 34301). Currently this must be in **section 34**.

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
- *EMS*
  - **S** to output a surface emission (2D)
  - **A** to output aircraft emissions (3D)
  - **L** to output lightning emissions (3D)
  - **V** to output volcanic SO2 emissions (3D)
  - **T** to output SO2 emissions (3D)
- *PSC*
  - **1** to output the fraction of Type 1 PSCs
  - **2** to output the fraction of Type 2 PSCs
- *RTE*
  - **B** to output the rate of a bimolecular reaction
  - **T** to output the rate of a termolecular reaction
  - **H** to output the rate of a heterogeneous reaction
  - **J** to output the rate of a photolysis reaction

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*, *EMS*, *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_scheme.F90** modules.

Addition of 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**.

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 make a new STASHmaster file for diagnostics that you have defined yourself. The easiest option is to copy an existing diagnostic specification from the *STASHmaster_A* file, which is located at

```
/work/n02/n02/hum/vn8.4/ctldata/STASHmaster/STASHmaster_A
```

on ARCHER, and at

```
/projects/um1/vn8.4/ctldata/STASHmaster/STASHmaster_A
```

on MONSooN.

2D Diagnostic Fields

If you are outputting a 2D field such as a surface emission or a dry deposition field (e.g. you are using the old 2D dry deposition scheme) then the fact that this is a 2D surface field is defined in the STASHmaster entry, which looks like this e.g.

```
#
1|     1 |    34 |   321 |Ox BUDGET: O3 DRY DEPOSITION (2D)    |
2|     0 |     0 |     1 |     1 |     5 |    -1 |    -1 |     0 |     0 |     0 |     0 |
3| 00000000000000000000000000000000 | 00000000000000000001 |     3 |
4|     1 |     0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
5|     0 |  1871 |     1 |   129 |     0 |     0 |     0 |     0 |     0 |
#
```

However, you can always output a surface diagnostic using a STASHmaster file that defines a 3D field, it is just that only the surface values will be non-zero (if the full field is output).

You will need to edit the two elements in red to match your new diagnostic.

3D Diagnostic Fields

A 3D diagnostic field has slightly different values for some of the elements of the STASHmaster definition. These can been seen by comparing the specification above with the one below

```
#
1|     1 |    34 |   321 |Ox BUDGET: O3 DRY DEPOSITION (3D)    |
2|     0 |     0 |     1 |     1 |     2 |    10 |    11 |     0 |     0 |     0 |     0 |
3| 00000000000000000000000000000000 | 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 |
#
```

You will need to edit the two elements in red to match your new diagnostic.

STASHmaster file format

As well as defining the STASH items in your new user STASHmaster file, you will also need to include the correct preamble and an end of file specifier. These are

Preamble:

```
H1| SUBMODEL_NUMBER=1
H2| SUBMODEL_NAME=ATMOS
H3| UM_VERSION=8.4
#
#|Model |Sectn | Item |Name                                 |
#|Space |Point | Time | Grid |LevelT|LevelF|LevelL|PseudT|PseudF|PseudL|LevCom|
#| Option Codes                  | Version Mask          | Halo |
#|DataT |DumpP | PC1  PC2  PC3  PC4  PC5  PC6  PC7  PC8  PC9  PCA |
#|Rotate| PPF  | USER | LBVC | BLEV | TLEV |RBLEVV| CFLL | CFFF |
#
```

which is also helpful in describing what the elements in the STASH specification are.

End of file specifier:

```
#
1|    -1 |    -1 |    -1 |END OF FILE MARK                     |
2|     0 |     0 |     0 |     0 |     0 |     0 |     0 |     0 |     0 |     0 |
3| 00000000000000000000000000000000 | 00000000000000000000 |     0 |
4|     0 |     0 | -99  -99  -99  -99  -30  -99  -99  -99  -99  -99 |
5|     0 |     0 |     0 |     0 |     0 |     0 |     0 |     0 |     0 |
#
```

UMUI Changes

After you have made your STASHmaster file with the required diagnostics, you should add it to the UMUI in **Model Selection → Atmosphere → STASH → User-STASHmaster files. Diags, Progs & Ancils**. You will also need to give it an initial value in **Model Selection → Atmosphere → STASH → Initialisation of User Prognostics** (you can just set it to **zero**, i.e. **3** in the **Option** column). After you have done this you can select the diagnostic in the **Model Selection → Atmosphere → STASH → STASH. Specification of Diagnostics requirements** panel.

Time Validity

When outputting UKCA reaction and deposition fluxes, you need to consider what time-points these diagnostics are valid at. Tracer fields are valid at all timesteps, but for reaction fluxes this is not the case. For all the chemistry schemes which use ASAD/Newton-Raphson (*CheT/TropIsop*, *CheS/Strat*, and *CheST/StratTrop*), the UKCA chemical timestep is every hour.

This means that when you output this diagnostic through STASH you need to change the *sampling frequency*. To do this, go to the STASH panel in the UMUI (**Model Selection → Atmosphere → STASH → STASH. Specification of Diagnostic requirements**) and **copy** the **time profile** you would like to use for the diagnostic to a new profile (giving it a similar, but slightly different name) and then change the **sampling frequency** to be

- **Frequency (every)** to be **3**
- **Offset** to be **2**

This will then sample the diagnostic on the UKCA timesteps. If you don't sample the diagnostic using this sampling frequency, then the output may be incorrect.

The following diagnostics are valid on **all timesteps**:

- *MAS*
- *OUT*
- *TPM*
- *STE*
- *EMS*

The following diagnostics are only valid on **chemical timesteps**:

- *RXN*
- *DEP*
- *RTE*
- *PSC*

Why should I use these numbers?

The standard N96L85 GA4.0 job has a timestep of 20-minutes (72 timesteps per day). This means that 1-hour is 3 timesteps, and the first timestep of a day is the 1st. An off-set of 2 will then mean outputting on the hour.

Note: If you are using a job with a different timestep you will need to change this sampling, as by default UKCA will still run with a 1-hour timestep.

Solution to Task 9.1: Output new diagnostics

Please see this page for a solution to Task 9.1.

Written by Luke Abraham 2014

Solution to UKCA Chemistry and Aerosol Tutorial 9 Task 9.1

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Back to the adding new diagnostics tutorial

Contents

- 1 Task
- 2 Solution
  - 2.1 Changes to asad_flux_dat.F90
  - 2.2 STASHmaster file
  - 2.3 UMUI STASH table changes
    - 2.3.1 Create a new time profile
    - 2.3.2 Output the diagnostics
- 3 Output
- 4 Worked Solution

Task

You were asked to

> Output diagnostics of the reaction $\mathrm{ALICE + OH \longrightarrow BOB + SEC\_ORG}$ to STASH code **34461**, the dry deposition of **ALICE** to STASH code **34462**, and the wet deposition of **BOB** to **34463**. They should be outputted as a daily mean to the **pa/UPA** stream.

and were given the hint

> Remember to use the correct sampling frequency.

Solution

As well as making some code changes and making a new user STASHmaster file, you will also need to make some changes to the STASH table to output the diagnostics correctly.

Changes to asad_flux_dat.F90

The only code changes that need to be made are in **asad_flux_dat.F90**, where you will need to add the following diagnostic specification blocks

```
      asad_flux_defn('RXN',34461,'B',.FALSE.,0,4,                       &
      (/'ALICE     ','OH        '/),                                    &
      (/'BOB       ','Sec_Org   ','          ','          '/)),         &
      asad_flux_defn('DEP',34462,'D',.FALSE.,0,1,                       &
      (/'ALICE     ','          '/),                                    &
      (/'          ','          ','          ','          '/)),         &
      asad_flux_defn('DEP',34463,'W',.FALSE.,0,1,                       &
      (/'BOB       ','          '/),                                    &
      (/'          ','          ','          ','          '/))          &
```

It is probably best to make this as a separate array, rather than add this to the end of an existing array. You should then add this array into the **asad_chemical_fluxes** master array that holds all the possible diagnostics. You should also increase the **n_chemical_fluxes** integer by 3 to **235** to take account of these extra fluxes.

STASHmaster file

You will need to make a user STASHmaster file which looks similar to

```
H1| SUBMODEL_NUMBER=1
H2| SUBMODEL_NAME=ATMOS
H3| UM_VERSION=8.2
#
#|Model |Sectn | Item |Name                                |
#|Space |Point | Time | Grid |LevelT|LevelF|LevelL|PseudT|PseudF|PseudL|LevCom|
#| Option Codes                  | Version Mask         | Halo |
#|DataT |DumpP | PC1  PC2  PC3  PC4  PC5  PC6  PC7  PC8  PC9  PCA |
#|Rotate| PPF  | USER | LBVC | BLEV | TLEV |RBLEVV| CFLL | CFFF |
#
#================================================================================
#
1|    1 |   34 |  461 |UKCA Tutorial: ALICE+OH->BOB+Sec_Org|
2|    0 |    0 |    1 |    1 |    2 |   10 |   11 |    0 |    0 |    0 |    0 |
3| 00000000000000000000000000000000 | 00000000000000000001 |    3 |
4|    1 |    0 | -99  -99  -99  -99  -99  -99  -99  -99  -99  -99 |
```

```
5|     0 | 1871 |     0 |    65 |     0 |     0 |     0 |     0 |
#
1|     1 |    34 |   462 |UKCA Tutorial: ALICE Dry Dep (3D)    |
2|     0 |     0 |     1 |     1 |     2 |    10 |    11 |     0 |     0 |     0 |
3| 00000000000000000000000000000 | 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 |    34 |   463 |UKCA Tutorial: BOB Wet Dep (3D)      |
2|     0 |     0 |     1 |     1 |     2 |    10 |    11 |     0 |     0 |     0 |
3| 00000000000000000000000000000 | 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 |    -1 |    -1 |END OF FILE MARK                     |
2|     0 |     0 |     0 |     0 |     0 |     0 |     0 |     0 |     0 |     0 |
3| 00000000000000000000000000000 | 00000000000000000000 |     0 |
4|     0 |     0 | -99  -99  -99  -99  -30  -99  -99  -99  -99  -99 |
5|     0 |     0 |     0 |     0 |     0 |     0 |     0 |     0 |     0 |
#
```

This should be included in the UMUI in **Model Selection → Atmosphere → STASH → User-STASHmaster files. Diags, Progs & Ancils** and the three diagnostics should be initialised to zero (**Option 3**) in **Model Selection → Atmosphere → STASH → Initialisation of User Prognostics**.

An example STASHmaster file can be found at

```
/home/ukca/userprestash/VN8.4/Tutorial/Task9.1_Tutorial_Diags.presm
```

UMUI STASH table changes

Once you have added in your new user STASHmaster file, you can now output these diagnostics through the STASH panel, by placing entries in the STASH table. However, to do this properly you will need to make a new STASH **Time Profile**.

Go to **Model Selection → Atmosphere → STASH → STASH. Specification of Diagnostics requirements** to make these changes.

Create a new time profile

We want to output these diagnostics as daily means, but we cannot use the existing **TDAYM** time profile, as the *sampling frequency* is incorrect. We also don't want to alter the existing *TDAYM* profile as this will then affect all diagnostics currently outputted using this profile. To make the required changes you need to

1. Select the **TDAYM** profile (it should now highlight yellow)
    1. Go to **Profiles → Copy Profile → Copy time**
    2. You will now be asked for a new name to copy this profile to (e.g. **TDYMUKCA**)
2. Now select this new profile
    1. Edit it by going to **Profiles → Edit Profile → Edit time**
    2. Set the **Frequency (every)** to **3**
    3. Change the **Sampling offset** to **2**
    4. Click **Close** to save these changes

You have now created a new time profile with the correct sampling frequency for UKCA diagnostics.

Output the diagnostics

Now that you have created the new time profile (**TDYMUKCA**), you should use it to output the diagnostics as

```
34 461 UKCA Tutorial: ALICE+OH->BOB Flux   TDYMUKCA DALLTH   UPB      Y +  Y
34 462 UKCA Tutorial: ALICE Dry Dep (3D)   TDYMUKCA DALLTH   UPB      Y +  Y
34 463 UKCA Tutorial: BOB Wet Dep (3D)     TDYMUKCA DALLTH   UPB      Y +  Y
```

Output

Your **pa** file (located in your job directory) should now contain the following fields

```
0     : 192   145   85    1      Stash code = 34001
1     : 192   145   85    1      Stash code = 34064
2     : 192   145   85    1      Stash code = 34065
3     : 192   145   85    1      Stash code = 34461
4     : 192   145   85    1      Stash code = 34462
5     : 192   145   85    1      Stash code = 34463
```

The surface plot of the flux through the $\mathrm{ALICE} + \mathrm{OH} \longrightarrow \mathrm{BOB} + \mathrm{Sec\_Org}$ can be seen in Figure 1, the dry deposition flux of ALICE is in Figure 2, and the wet deposition flux of BOB is in Figure 3. All fluxes are in moles/gridcell/s.

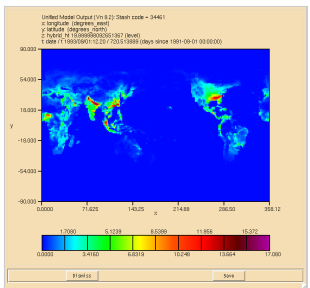


Figure 1: Flux through the ALICE+OH->BOB Reaction in the lowest model level.

Sample output from this job can be found in

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Task9.1/
```

on ARCHER, and in

```
~~/projects/ukca/Tutorial/vn8.4/sample_ouput/Task9.1/~~
```

on MONSooN.

Worked Solution

There is a worked solution to this problem in the UMUI Tutorial experiment. This is job **j**: *Tutorial: solution to Task 9.1 - add new diagnostics*.

The code changes can be viewed by using the following FCM command

```
fcm diff fcm:um_br/dev/luke/vn8.4_UKCA_Tutorial_Solns@14713 fcm:um_br/dev/luke/vn8.4_
```

This gives the following output:

```
Index: src/atmosphere/UKCA/asad_flux_dat.F90
===================================================================
--- src/atmosphere/UKCA/asad_flux_dat.F90       (revision 14713)
+++ src/atmosphere/UKCA/asad_flux_dat.F90       (revision 14714)
@@ -102,7 +102,7 @@
       CHARACTER(LEN=10) :: blank0 = '          '   ! Defines null product

 ! Number of chemical fluxes defined below
-      INTEGER, PARAMETER :: n_chemical_fluxes = 232
+      INTEGER, PARAMETER :: n_chemical_fluxes = 235

 ! Do we load standard diagnostics as done below (T), or use only reactions
 !  specified in STASH (F) ?
@@ -923,6 +923,21 @@
        /)


+! UKCA Tutorial fluxes
+       TYPE(asad_flux_defn), PARAMETER, PUBLIC ::                        &
+                                    UKCA_Tutorial_Fluxes(3) = (/    &
+       asad_flux_defn('RXN',34461,'B',.FALSE.,0,4,                       &
+       (/'ALICE     ','OH        '/),                                    &
+       (/'BOB       ','Sec_Org   ','          ','          '/)),         &
+       asad_flux_defn('DEP',34462,'D',.FALSE.,0,1,                       &
+       (/'ALICE     ','          '/),                                    &
+       (/'          ','          ','          ','          '/)),         &
+       asad_flux_defn('DEP',34463,'W',.FALSE.,0,1,                       &
+       (/'BOB       ','          '/),                                    &
+       (/'          ','          ','          ','          '/))          &
+       /)
+
+
       PUBLIC :: ASAD_LOAD_DEFAULT_FLUXES
       INTERFACE ASAD_LOAD_DEFAULT_FLUXES
         MODULE PROCEDURE ASAD_LOAD_DEFAULT_FLUXES
@@ -1005,7 +1020,8 @@
        asad_strat_oh_loss,          & ! 26 181
        asad_strat_o3_budget,        & ! 20 201
        asad_strat_o3_misc,          & ! 15 216
-       asad_aerosol_chem            & ! 16 232
+       asad_aerosol_chem,           & ! 16 232
+       UKCA_Tutorial_Fluxes         & !  3 235
        /)

       ELSE
```

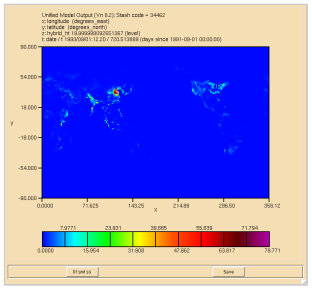


Figure 2: Dry deposition of ALICE in the lowest model level.

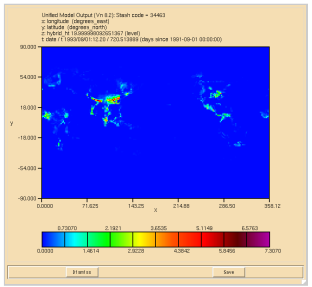


Figure 3: Wet deposition of BOB in the lowest model level.

Written by Luke Abraham 2014

- This page was last modified on 15 December 2015, at 15:20.

UKCA Chemistry and Aerosol Tutorial 10

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Contents

- 1 What you will learn in this Tutorial
- 2 Task 10.1: What is the aerosol optical depth?
- 3 Task 10.2: Understand the principles behind GLOMAP and how aerosol optical properties are derived via RADAER
- 4 Task 10.3: Output daily-mean Aerosol Optical Depths from your UKCA model run
  - 4.1 Worked Solution

What you will learn in this Tutorial

In this tutorial you will learn about the Aerosol Optical Depth and how the RADAER module diagnoses aerosol optical properties from the GLOMAP aerosol microphysics scheme included in UKCA.

Note: The GLOMAP aerosol tutorials use a slightly different base job. Please take a copy of **`xjrnk`** and work from that for these tasks. Example output from this job can be found on ARCHER in the directory

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Base_Aerosol
```

Task 10.1: What is the aerosol optical depth?

Aerosol particles affect the Earth's radiative balance by scattering and absorbing solar radiation and, where they are large enough, can also act similarly to a greenhouse gas by absorbing outgoing terrestrial long-wave radiation.

The aerosol optical depth (AOD), sometimes referred to as aerosol optical thickness (AOT), is often used in atmospheric science to indicate the overall strength of aerosol-radiation interactions at a particular wavelength.

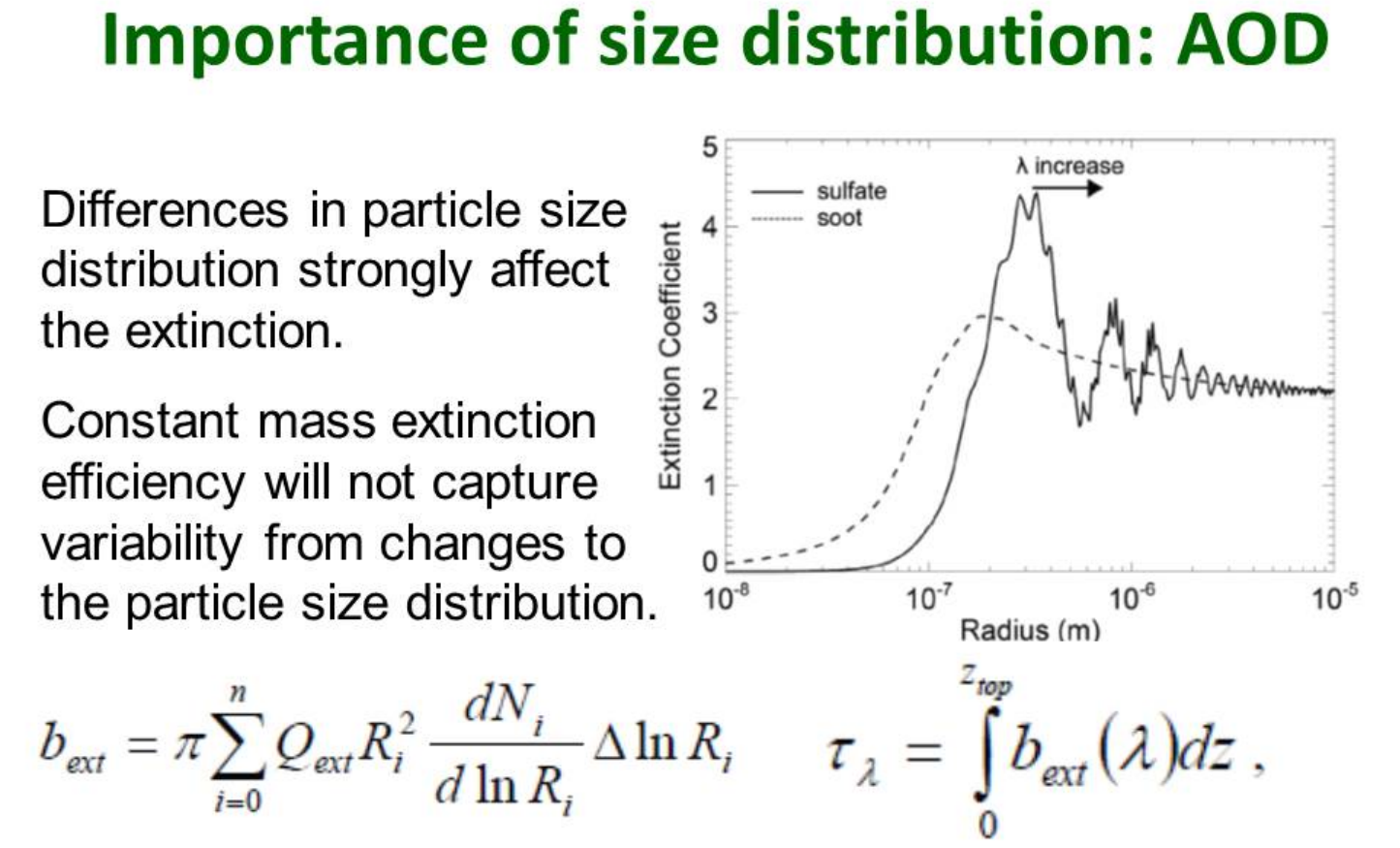
The AOD is defined as the vertical integral of the monochromatic (single-wavelength) extinction (the sum of scattering plus absorption) through the atmospheric column. It is common also to refer to an absorption AOD which represents the integral of just the single-wavelength absorption.

The Unified Model (UM) radiation scheme divides the shortwave and longwave spectra into wavebands. To enable UKCA simulated gases or aerosols to interact with the UM radiation scheme, their optical properties need to be integrated across each of these wavebands. For aerosols, the radiation scheme requires the specific scattering and absorption coefficients, which describe the strength of aerosol scattering and absorption processes per unit aerosol mass (in $m^2\ kg^{-1}$), and the asymmetry parameter, which describes in a simplified way the angular dependence of the scattering (dimensionless). The specific scattering and absorption coefficients, and the asymmetry parameter, are hereafter referred to as the *aerosol optical properties*.

Task 10.2: Understand the principles behind GLOMAP and how aerosol optical properties are derived via RADAER

Mie theory describes the scattering and absorption of light by spherical particles. Essentially, the scattering/absorption efficiency can be parameterized in terms of two parameters: the (complex) refractive index of the particle and the Mie parameter, which describes the particle size dependence in relation to the wavelength of light under consideration.

The Figure in the slide below shows the dependence on particle size of the extinction efficiency for mid-visible (550nm, green) light for two different types of particle (refractive index).



The aerosol scheme used for UK climate model simulations in CMIP5 was called the Coupled Large-scale Aerosol Simulator for Studies In Climate (CLASSIC). CLASSIC is a simpler scheme than the microphysical GLOMAP scheme included in UKCA but has a comprehensive representation of the main aerosol sources, tracking up to eight tropospheric aerosol species: ammonium sulphate, mineral dust, sea salt, fossil fuel black carbon, fossil fuel organic carbon, biomass burning aerosols, secondary organic (also called biogenic), and ammonium nitrate aerosols.

However, the size distribution and refractive index for each of the CLASSIC transported aerosol types was prescribed to have globally and temporally uniform values. Some variation in aerosol properties was resolved by having fresh and aged sub-types for each aerosol type, which then allow size and refractive index to vary from values near-source (fresh) to remote (aged).

With this approach, optical properties for each of the CLASSIC aerosol types is derived based on Mie calculations using the prescribed size distributions and refractive indices for each of the transported aerosol masses and assuming hygroscopic growth factors for the water associated with each.

The GLOMAP-mode aerosol scheme in UKCA simulates the evolution of size-resolved aerosol properties, including microphysical processes such as new particle formation, coagulation, condensation (gas-to-particle-transfer) and cloud processing.

Whereas CLASSIC simulated only the mass of several aerosol types, the transported tracers in GLOMAP are particle number and mass in different size classes spanning the particle size range from 3nm up to around 20 microns dry diameter.

Processes such as condensation and aqueous sulphate production grow particles by increasing the mass in a size class while conserving particle number.

As was as microphysical processes, GLOMAP also includes size-resolved representations of primary emissions (e.g. sea-salt, dust and carbonaceous particles) and of removal processes including particle dry deposition, sedimentation, nucleation scavenging (rainout) and impaction scavenging (washout).

GLOMAP therefore simulates the evolution of particle number and composition across the size spectrum over several different components as determined by the aerosol processes included.

The original version of GLOMAP (known as GLOMAP-bin) uses a two-moment sectional aerosol dynamics approach, with typically 20 bins spanning the size spectrum, but when tracking several aerosol types, becomes too expensive for running multi-decadal integrations as required in a climate model. For UKCA, a new computationally cheaper version of GLOMAP was developed (GLOMAP-mode), which has the same process representations, but using log-normal modes as its size classes. Each size mode in GLOMAP-mode covers one of four size ranges, with additional separation among soluble and insoluble modes.

GLOMAP has been developed to follow a flexible multi-component approach, with the same code able to be run with different levels of composition/size sophistication with FORTRAN-90 modules providing alternative aerosol *mode set-up* arrays. The GLOMAP-mode aerosol scheme in UKCA not only simulates the dry aerosol mass, but also the mass of water attached to the aerosol, and the aerosol number concentrations.

The calculation of aerosol optical properties from GLOMAP is carried out via the RADAER module within UKCA.

Compared to CLASSIC, the GLOMAP-mode scheme introduces three important changes which are relevant for simulated aerosol-radiation interactions:

- The mean radius of each size class (mode) varies in time and space as determined by transport and aerosol processes.
- There is a refractive index for each size class (mode) which varies according to its internally-mixed composition.
- The amount of aerosol water in each soluble mode varies interactively consistent with its composition.

RADAER relies on pre-computed look-up tables of monochromatic optical properties, covering all realistic combinations of modal radii and refractive indices. At runtime, remaining tasks are:

- To compute the modal refractive index out of the simulated chemical composition of each mode.
- To obtain the monochromatic properties from the look-up tables at selected wavelengths within each shortwave and longwave waveband.
- To integrate across each waveband to obtain the waveband-averaged optical properties, which can be used by the radiation code.

As well as providing the waveband-averaged scattering, extinction and asymmetry parameters from GLOMAP, RADAER also diagnoses monochromatic Aerosol Optical Depth (AOD) for each of the GLOMAP modes.

Task 10.3: Output daily-mean Aerosol Optical Depths from your UKCA model run

In this task you will update your copy of the UKCA tutorial job (**`xjrnk`**) adding extra STASH requests to output daily-mean AODs from GLOMAP.

UM-UKCA diagnoses the AOD from each of the GLOMAP modes as a separate STASH item.

All of the GLOMAP AOD diagnostics are contained with STASH section 2 (long-wave radiation) in items numbers near to those used for the AOD diagnostics from CLASSIC.

Section 2 items 300-305 contain the AOD for the Aitken-soluble, accumulation-soluble, coarse-soluble, Aitken-insoluble, accumulation-insoluble and coarse-insoluble modes respectively.

The dimensions for each of these STASH items is 2D global in longitude and latitude but there is also a third dimension containing 6 pseudo levels for the 6 monochromatic AODs stored by the model (at 0.38, 0.44, 0.55, 0.67, 0.87, 1.02 micron wavelengths).

Note that there is no nucleation mode AOD as those particles are too small to significantly scatter or absorb radiation at these wavelenghts.

Note that although the absorption AOD in these 6 modes can usually be requested via STASH section 2 items 240 to 245, in the job here these are not available.

There are also separate "stratospheric AOD" diagnostics in section 2 items 251 to 256 which can also be requested to store the AOD in levels above the tropopause.

To output AODs from GLOMAP, you need to add in some extra STASH requests for the section 2 items 300 to 305.

To output as a daily-mean select the usage profile "UPA" to output to the .pa files and the time profile "TDAYM" for daily-means. For AOD STASH requests, you need to select the domain profile DIAGAOT to output over the longitude, latitude and 6 wavelength pseudo-levels.

Section 34 items 106, 110, 116, 121 and 126 contain the OM mmr's in the Aitken-soluble, accumulation-soluble, coarse-soluble, Aitken-insoluble and nucleation-soluble modes respectively.

When requesting the OM mmr's you should use the DALLTH domain profile to request the variable on the full 3D model grid.

So now you should have an equivalent version of the UKCA tutorial job with extra daily-mean fields requested.

Since you specified the UPA usage profile in the job, these fields will be output to the so-called *.pa files*.

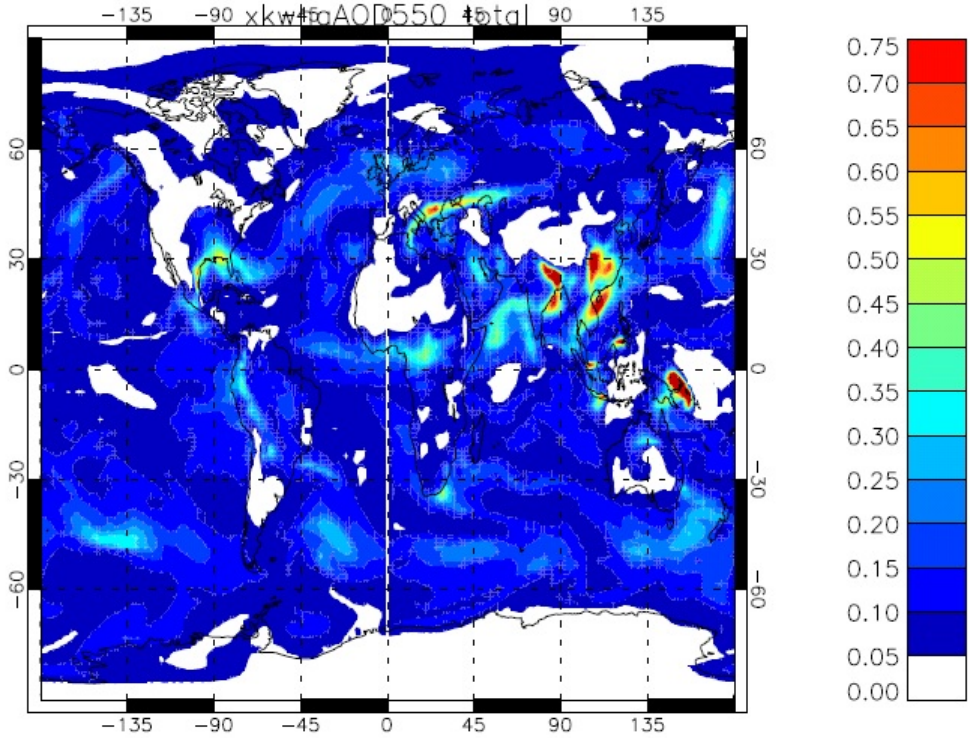
These can be found in your /work/n02/n02/ directory on ARCHER. You should have a file like **jobid**a.pa19991201 for the daily-mean fields for the 1 day that the job was set to run for.

In the first instance you should use xconv to open the file although you can use convsh or other tools to extract required fields from the .pa file and then manipulate them in IDL or python.

As an example I have put below a jpeg showing a global map produced in IDL of the daily-mean AOD at 550nm (total over all modes) for 1st December 1999 from the UKCA tutorial job (worked solution **`xjrnl`**).

In preparation for the next task, you should also re-run the job adding also daily-mean STASH requests for the following mass mixing ratios (mmr's) for the Organic Matter (OM) in each mode (34106, 34110, 34116, 34121 and 34126).

Once you have re-run the job with these extra fields available in the .pa file, you can use the cdo operator *cdo add* to sum up each of the OM mmrs to give a total OM mmr in a separate netCDF file.



Worked Solution

A worked soltion to **Task 10.3** is job **`xjrnl`**.

You can use **cdo** as so to add the 5 separate netCDF files containing the OC mass-mixing ratios:

```
$ cdo add xkvxm_aitken_sol_OCmmr.nc -add xkvxm_accum_sol_OCmmr.nc -add xkvxm_coarse_sol_OCmmr.nc -add xkvxm_aitken_insol_OCmmr.nc
xkvxm_nuc_sol_OCmmr.nc xkvxm_OCmmr.nc
cdo add: Started child process "add xkvxm_accum_sol_OCmmr.nc -add xkvxm_coarse_sol_OCmmr.nc -add xkvxm_aitken_insol_OCmmr.nc
xkvxm_nuc_sol_OCmmr.nc (pipe1.1)".
cdo(2) add: Started child process "add xkvxm_coarse_sol_OCmmr.nc -add xkvxm_aitken_insol_OCmmr.nc xkvxm_nuc_sol_OCmmr.nc (pipe2.1)".
cdo(3) add: Started child process "add xkvxm_aitken_insol_OCmmr.nc xkvxm_nuc_sol_OCmmr.nc (pipe3.1)".
cdo(4) add: Processed 4732800 values from 2 variables over 2 timesteps ( 0.23s )
cdo(3) add: Processed 4732800 values from 2 variables over 2 timesteps ( 0.25s )


cdo(2) add: Processed 4732800 values from 2 variables over 2 timesteps ( 0.27s )
cdo add: Processed 4732800 values from 2 variables over 2 timesteps ( 0.28s )
```

When opening the output file (`xkvxm_OCmmr.nc` in the example above) the variable name is taken from the first file in the list.

Example output can be found on ARCHER in the directory

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Task10.3
```

Written by Graham Mann 2014

UKCA Chemistry and Aerosol Tutorial 11

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Contents

- 1 What you will learn in this Tutorial
- 2 Task 11.1: Understand how the GLOMAP aerosol module tracks aerosol species and modes
- 3 Task 11.2: Run a copy of the standard UKCA job which tracks two OC components in the GLOMAP modes
  - 3.1 Worked Solution
- 4 Task 11.3 Examine the simulated total organic carbon in the original and two-cpt OC configurations

What you will learn in this Tutorial

In this tutorial you will learn about the GLOMAP-mode aerosol module and how it tracks different aerosol types within several size classes. You will understand the standard configuration used in the UKCA jobs so far whereby the mass mixing ratios of sulphate, sea-salt, black carbon and organic matter in each mode are transported via separate tracers. GLOMAP-mode is an aerosol microphysics scheme and therefore, as well as transporting the mass of several components in the modes, the scheme also transports the number concentrations of particles in each mode.

Task 10 already introduced the basic concepts behind the GLOMAP-mode aerosol microphysics scheme and how it differs from the mass-based CLASSIC scheme which preceded UKCA.

Initially developed in the TOMCAT CTM environment (see Manktelow et al., 2007; Mann et al., 2010; Mann et al., 2012), the GLOMAP code then became the aerosol module for the UKCA sub-model of the UM (see Bellouin et al., 2013; Kipling et al., 2013; West et al., 2014; Mann et al., 2014; Dhomse et al., 2014).

GLOMAP is now also implemented into the ECMWF Integrated Forecasting System as part of the "Composition IFS" module (C-IFS) where it will be used in combination with data assimilation of satellite Aerosol Optical Depth to provide forecasts and re-analyses of atmospheric composition and boundary conditions for regional air quality models.

The GLOMAP-mode code allows several alternative "aerosol configurations" to be run using the same set of FORTRAN subroutines.

In section 12 of the UKCA UMDP, Table 18 shows the standard configuration for GLOMAP in all 3 of these modelling frameworks (TOMCAT, UM-UKCA and C-IFS-GLOMAP).

There the model runs with 7 modes each containing mixtures of up to 5 different aerosol components (sulphate, black carbon, organic matter, sea-salt and dust).

In the full configuration (known as setup 8) the model runs has 7 number mixing ratios (one for each mode) and a total of 19 component mass mixing ratios.

When GLOMAP is run within UM-UKCA, dust is handled by the existing 6-bin UM scheme, and GLOMAP is configured to use the "5-mode configuration" (known as setup 2) covering only 4 of the above 5 components (sulphate, black carbon, organic matter and sea-salt).

The scheme can also be reduced to cover just the sulphate and sea-salt components in 4 modes (known as setup 1) or extended to track two separate components for organic matter (OM) to track the mass of primary OM and secondary OM in each mode.

Section 12.2 of the UMDP has a more detailed explaination of these configurations with Table 19 showing how these 4 different *GLOMAP-mode setups* map onto the model tracers.

In this task you will take a copy of the standard UKCA job (which uses GLOMAP-mode setup 2, MS2) and change it to use GLOMAP-mode setup 4 (MS4) to track two separate organic matter (OM) components rather than the usual 1. With the 2-component OC configuration, the model tracks primary (emitted) organic carbon in the usual OM component and secondary organic matter (formed following oxidation in the atmosphere) separately in a 2nd OM component.

Task 11.1: Understand how the GLOMAP aerosol module tracks aerosol species and modes

TASK 11.1: Read section 12 (page 32) of the v8.4 UM Documentation Paper and refer to Tables 18, 19 and 20 on pages 33, 34 and 35.

Task 11.2: Run a copy of the standard UKCA job which tracks two OC components in the GLOMAP modes

Copy your copy of the standard UKCA tutorial job (xjrnk) from the UMUI and change the settings from the default UM-UKCA configuration for GLOMAP (setup 2) to instead use the 2-component OM configuration (setup 4).

To run the 2-component GLOMAP configuration, you will need to change an existing hand-edit in the UMUI to specify that you wish to run the model with additional aerosol tracers switched on, and change the switch controlling the GLOMAP setups from 2 to 4.

First, open the hand-edits panel in the UMUI and find the line specifying to use the hand-edit

```
~mdalvi/umui_jobs/hand_edits/vn8.4/config_plume_scav_on_st.ed
```

If you open this file in an editor you can get an idea of what the hand-edit does.

The hand-edit begins by introducing a new logical variable and parameter for the convective scavenging module in UKCA.

But the relevant section of the hand-edit for this task is the where it edits the file *SIZES* setting the values of the array *TC_UKCA* which specifies which of the UKCA tracers are switched on (=1) or off (=0).

The order of the tracers here matches that specified in the code in the array *nm_spec* in the routine *ukca_init.F90*.

If you look in that file (e.g. checkout the package branch used in the model) you can check which tracers and switched on and off.

In the standard tracer configuration for UKCA specified in this *config_plume_scav_on_st.ed* hand-edit, tracers *Ait_SOL_OC* (index 106), *Acc_SOL_OC* (110), *Cor_SOL_OC* (116), *Ait_INS_OC* (121) and *Nuc_SOL_OC* (126) are set to 1 but tracers *Nuc_SOL_SO* (128), *Ait_SOL_SO* (129), *Acc_SOL_SO* (130) and *Cor_SOL_SO* (131) are set to zero.

To run with the 2-component OC configuration, the additional "SO" tracers to store the 2nd organic matter component in each mode need to be switched on.

Also, the hand-edit needs to change the value of *I_MODE_SETUP* via the GLOMAP UMUI panel (labelled *MODE*) from the selection of Atmosphere-->Model Configuration--> UKCA --> MODE and the button-selector *Set Aerosol Species and Modes*).

Currently, there are two options for *Set Aerosol Species and Modes* which are *H2SO4, NaCl in 4 modes* for GLOMAP setup 1 (MS1) and *H2SO4, NaCl, BC and OC in 5 modes* for GLOMAP setup 2 (MS2), which set the value of *I_MODE_SETUP* to be 1 or 2 accordingly.

The UMUI does not yet have the option to automatically select GLOMAP setup 4 (*I_MODE_SETUP*=4). So that also needs to done via the hand-edit.

The hand-edit edits the file *CNTLATM* to change the switch *I_MODE_SETUP* from the value of 2 set in the UMUI to instead be set to 4 for the 2-component OC configuration.

An equivalent hand-edit for the 2-component OC configuration to apply these changes has already been produced which you can find in the file:

```
~gmann/stashfiles/config_plume_scav_on_st_MS4.ed
```

You can use a graphical difference tool like tkdiff or xxdiff to see the differences between the two files.

You see that running with the standard configuration of GLOMAP (I_MODE_SETUP=2) requires 83 tracers in the UKCA CheST configuration, with 20 coming from GLOMAP.

To run with the 2-component OM configuration of GLOMAP (I_MODE_SETUP=4) and the UKCA CheST chemistry required 3 addititional aerosol tracers, giving 86 in total.

If you compare against *config_plume_scav_on_st.ed*, you see that in *config_plume_scav_on_st_MS4.ed* TC_UKCA tracers 128, 129, 130 and 131 are set to 1 for the SO components in each of the soluble modes and the nucleation soluble OC mmr is no longer required as it has been replaced with SO mmr.

So to configure your copy of your initial UKCA tutorial job to run with 2-component OM, change the line in the UMUI hand-edits panel to point to

```
~gmann/stashfiles/config_plume_scav_on_st_MS4.ed
```

rather than

```
~mdalvi/umui_jobs/hand_edits/vn8.4/config_plume_scav_on_st.ed
```

Similarly, you also need to use an updated version of the RADAER hand-edit *raderv2_vn84_ARCHER.ed* to allow the 2-component OM configuration of GLOMAP to couple to the UM radiation scheme.

For this replace the hand-edit

```
~ukca/hand_edits/VN8.4/raderv2_vn84_ARCHER.ed
```

with the version for GLOMAP setup 4:

```
~gmann/stashfiles/raderv2_vn84_MS4_ARCHER.ed
```

The update to the hand-edit adds STASH requests to make available to RADAER the values of the partial volumes from the 2nd OM components at each radiation timestep to ensure they are included when calculating the GLOMAP aerosol optical properties.

To be able to add STASH requests for the additional OM components you also need to change the User-STASHmaster file in Atmosphere --> STASH --> User STASHmaster files, Diags, Progs and Ancils from

```
~ukca/userprestash/VN8.4/UKCA_Tr_StratTropAeroMODE.prestash
```

to instead use

```
~gmann/stashfiles/UKCA_Tr_StratTropAeroMODE_MS4.prestash
```

Once you have done this you should then add daily-mean STASH requests for section 34 items 128, 129, 130 and 131 for the SO component mmr's in each soluble mode as you already did for the OC mmr items in Task 10.

Finally, since you have asked the model to run with new tracers, you also need to specify how these should be initialised. Go to the Initialisation of User Prognostics panel off the STASH window. Scroll down until you see items 34128, 34129, 34130, 34131 and set the *Option* column to 3 so that these tracers are initialised to zero values for an NRUN.

Worked Solution

The worked solution to this task can be found in job **`xjrnm`**. Sample output can be found on ARCHER in the directory

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Task11.2
```

Task 11.3 Examine the simulated total organic carbon in the original and two-cpt OC configurations

In the above Task 11.2 you ran a 2-component OC version of the UKCA tutorial job (**xjrnk**). You can also refer to the worked solution **xjrnm** which I have configured in this way. See that **xjrnl** was the same as the UKCA tutorial job **xjrnk** except that I have added the extra STASH requests as in Task 10.3.

So by now you should have equivalent standard (as **xjrnl**) and 2-component OM (as **xjrnm**) versions of the UKCA tutorial job. In these jobs you have requested numerous daily-mean fields to be output in the .pa files.

So in your /work/n02/n02/ directory on ARCHER you should have **jobid**a.pa19991201 files for your standard and 2-component OM jobs.

Included in the extra STASH requests are the mass mixing ratios of OC (the standard organic component) and SO (the 2nd organic component) in each mode.

The OC mmrs are STASH section 34, items 126 (nucleation mode), 106 (Aitken-soluble), 110 (accumn-soluble), 116 (coarse-soluble) and 121 (Aitken-insoluble).

The SO mmrs are STASH section 34, items 128 (nucleation mode), 129 (Aitken-soluble), 130 (accumn-soluble), 131 (coarse-soluble).

These STASH item numbers and the details of the standard and 2-component GLOMAP configurations can be found in the UKCA UMDP section 12 Tables 19 and 20.

Note that there is no SO in the Aitken-insoluble mode as this contains only primary carbonaceous particles. Any SO or OC condensing onto the particles in the insoluble modes is immediately transferred over to the corresponding soluble mode following the "condensation-ageing" approach used by the model. This OC or SO condensing onto the insoluble particles is a kind of "coating" for the particles making the particles hygroscopic/soluble.

You could also try adding STASH requests for the mmr of the gas phase species MONOTER and SEC_ORG (STASH section 34, items 91 and 92).

As an example I have put here a link to a pdf OMcomparison (//www.ukca.ac.uk/wiki/index.php/File:GlobalMap_2cptOM.pdf) showing global maps comparing surface OM fields between the worked solutions **xjrnl** (top-left, labelled as xkwhg) and **xjrnm** (top-right, labelled as xkwhh).

Page 1 of the pdf compares the "total POM mmr" at the surface which is the total particulate organic matter (POM) summing up the mass of OC and SO in each mode.

Pages 2 and 3 show comparisons of "total POM1 mmr" and "total POM2 mmr" which are the sum of the 1st and 2nd organic component over all the modes.

You can see from the example that the "total POM2 mmr" in **xjrnl** is zero everywhere. That's because in this job GLOMAP has the standard configuration with just one organic component.

By contrast the "total POM2 mmr" for **xjrnm** has considerable concentrations in vegetated continental regions. In this "I_MODE_SETUP=4" configuration, the "SEC_ORG" species (which contains the secondary organics from monoterpene oxidation) condenses into the "SO" component, whereas in **xjrnl** SEC_ORG condenses into the "OC" component.

The bottom left panel on each page shows a global map of the ratio of the field for the two model runs. One can use this kind of approach to track the fraction of the OM that is biogenic and anthropogenic.

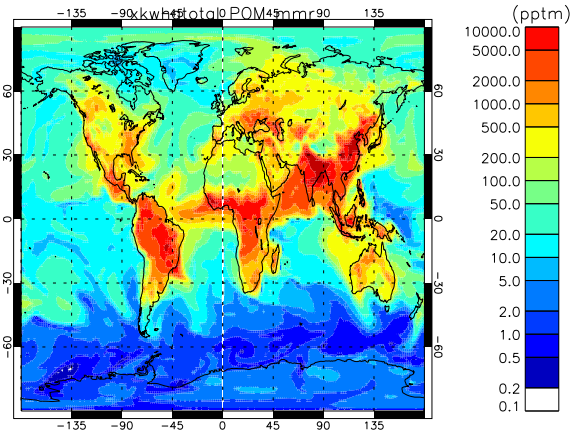
Note however that we initialised the SO mmr's to zero at the start of the 1-day run. So the OC1 mmrs will be spinning down and the SO mmrs will be spinning up. The daily-mean values are averaging over values on each the 1-hour timesteps over which the UKCA chemistry and aerosol processes are integrated. So although the ratio shown in the bottom-left on page 2 is indicative of the biogenic fraction it should be treated with caution as the fields will not have spun-up/down yet.

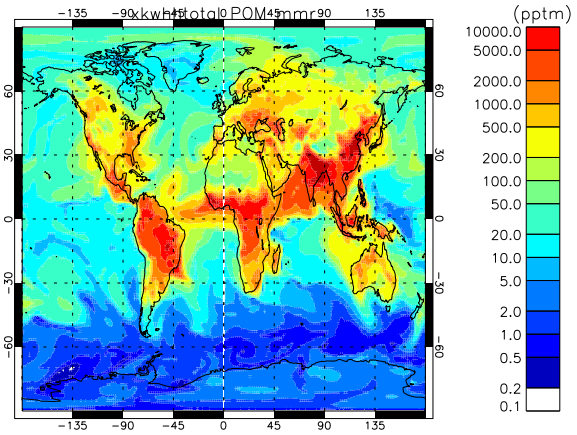
This task illustrates how one can separate out the aerosol mass from different sources and track them separately via a different aerosol component.

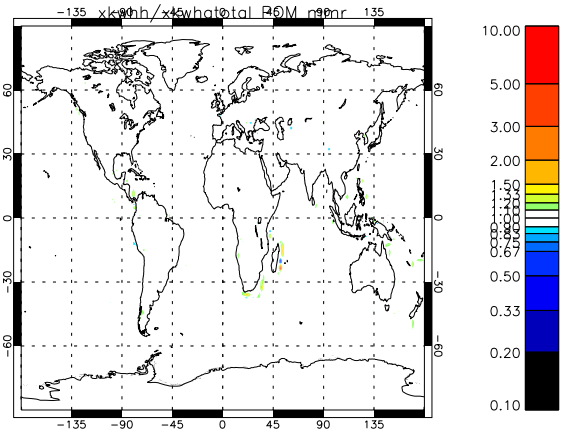
One could also introduce a 2nd gas phase species like "SEC_ORG" to track different types of SOA. For example one could configure the model so that such a 2nd "SEC_ORG2" species held semi-volatile oxidised organic species with very low volatility oxidised organics held in the usual "SEC_ORG" species.

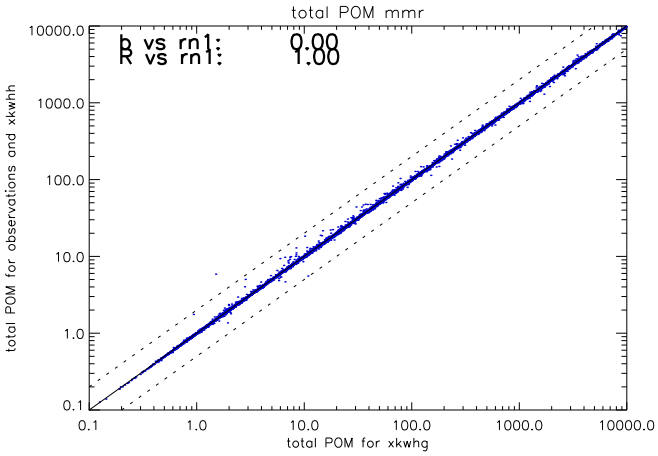
Written by Graham Mann 2014

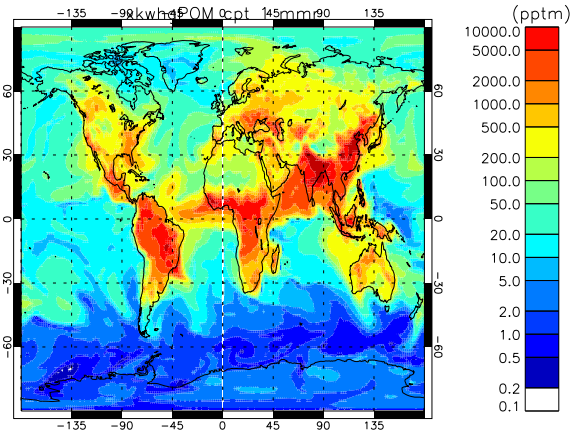
- This page was last modified on 7 January 2016, at 15:08.

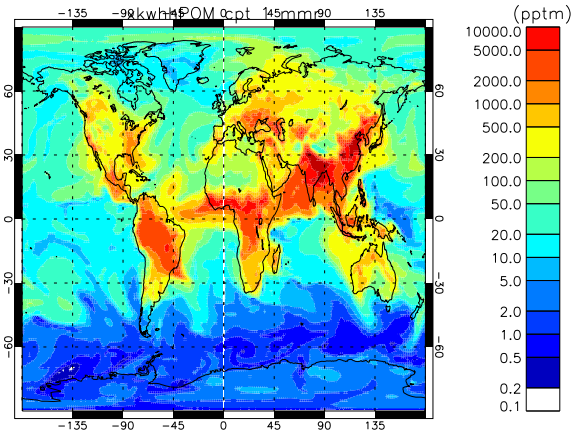

xkwhh total POM mmr
(pptm)
10000.0
5000.0
2000.0
1000.0
500.0
200.0
100.0
50.0
20.0
10.0
5.0
2.0
1.0
0.5
0.2
0.1

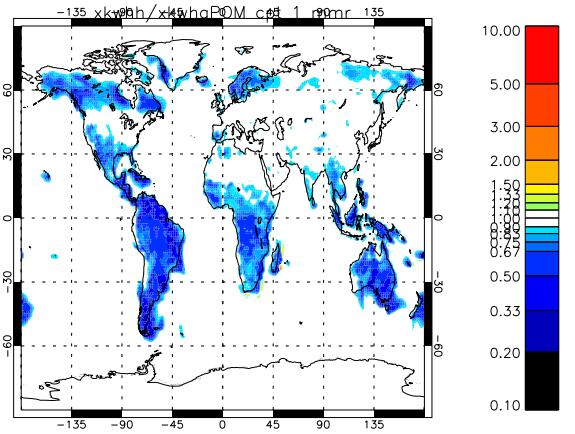

xkwhg total POM mmr
(pptm)
10000.0
5000.0
2000.0
1000.0
500.0
200.0
100.0
50.0
20.0
10.0
5.0
2.0
1.0
0.5
0.2
0.1

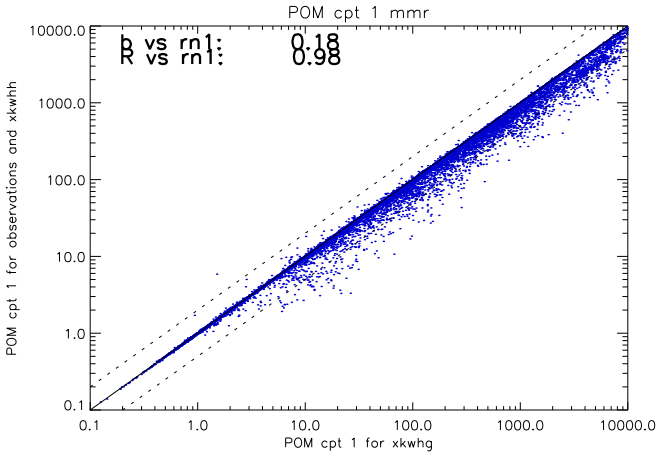

xkwhh/xkwhg total POM mmr
10.00
5.00
3.00
2.00
1.50
0.67
0.50
0.33
0.20
0.10

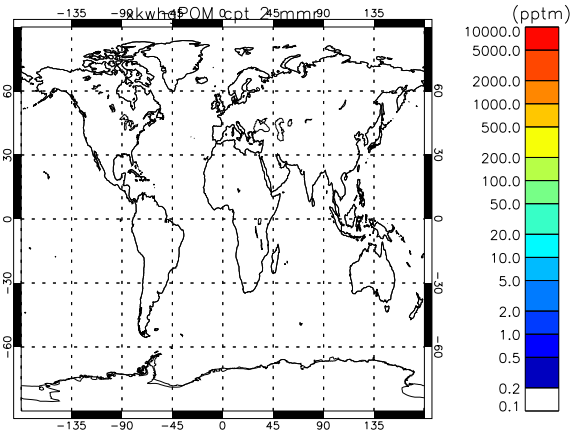

total POM mmr
b vs rn1: 0.00
R vs rn1: 1.00
total POM for observations and xkwhh
total POM for xkwhg
0.1
1.0
10.0
100.0
1000.0
10000.0

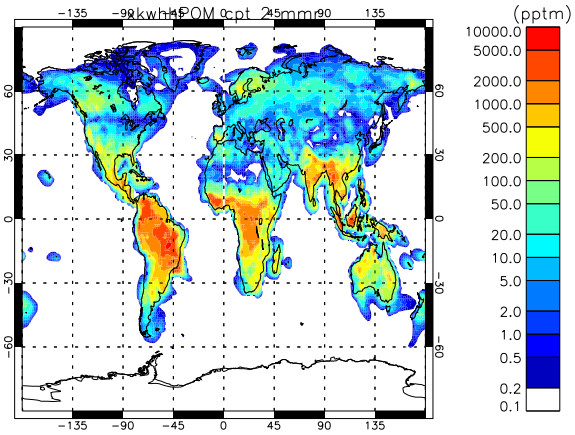

xkwhh/xkwhg POM cpt 1 mmr
(pptm)
10000.0
5000.0
2000.0
1000.0
500.0
200.0
100.0
50.0
20.0
10.0
5.0
2.0
1.0
0.5
0.2
0.1

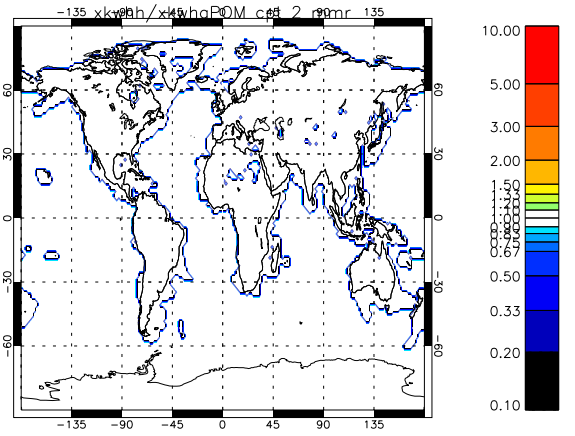

xkwhh/xkwhg POM cpt 1 mmr
(pptm)
10000.0
5000.0
2000.0
1000.0
500.0
200.0
100.0
50.0
20.0
10.0
5.0
2.0
1.0
0.5
0.2
0.1


xkwhh/xkwhg POM cpt 1 mmr
10.00
5.00
3.00
2.00
1.50
0.67
0.50
0.33
0.20
0.10


POM cpt 1 mmr
b vs rn1: 0.18
R vs rn1: 0.98
POM cpt 1 for observations and xkwhh
POM cpt 1 for xkwhg
10000.0
1000.0
100.0
10.0
1.0
0.1


kwhaPOM cpt 2 mmr
(pptm)
10000.0
5000.0
2000.0
1000.0
500.0
200.0
100.0
50.0
20.0
10.0
5.0
2.0
1.0
0.5
0.2
0.1


kwhaPOM cpt 2 mmr
(pptm)
10000.0
5000.0
2000.0
1000.0
500.0
200.0
100.0
50.0
20.0
10.0
5.0
2.0
1.0
0.5
0.2
0.1


10.00
5.00
3.00
2.00
1.50
0.67
0.50
0.33
0.20
0.10

UKCA Chemistry and Aerosol Tutorial 12

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

Contents

- 1 What you will learn in this Tutorial
- 2 Task 12.1: Update your copy of the UKCA tutorial job to diagnose Top Of the Atmosphere (TOA) radiative fluxes
  - 2.1 Example Output
- 3 Task 12.2 Configure the UKCA tutorial job to run as a *double-call* job diagnosing aerosol radiative effects
  - 3.1 Worked Solution

What you will learn in this Tutorial

In this tutorial you will learn about how to quantify the radiative effects of aerosol simulated by GLOMAP-mode in UM-UKCA.

In the first task you will update your copy of the UKCA tutorial job to request radiative fluxes allowing the radiative flux perturbation (or effective radiative forcing) to be diagnosed based on difference in the fluxes between a pair of UM-UKCA jobs with some difference (e.g. pre-industrial and present-day emissions jobs).

The second task involves configuring a copy of the UKCA tutorial job to run in *double-call configuration* whereby the aerosol radiative effects can be diagnosed at each radiation timestep.

Task 12.1: Update your copy of the UKCA tutorial job to diagnose Top Of the Atmosphere (TOA) radiative fluxes

In this task you will add STASH requests for SW and LW outgoing radiative fluxes at the top of the atmosphere to enable the radiative forcing from a particular change to be diagnosed.

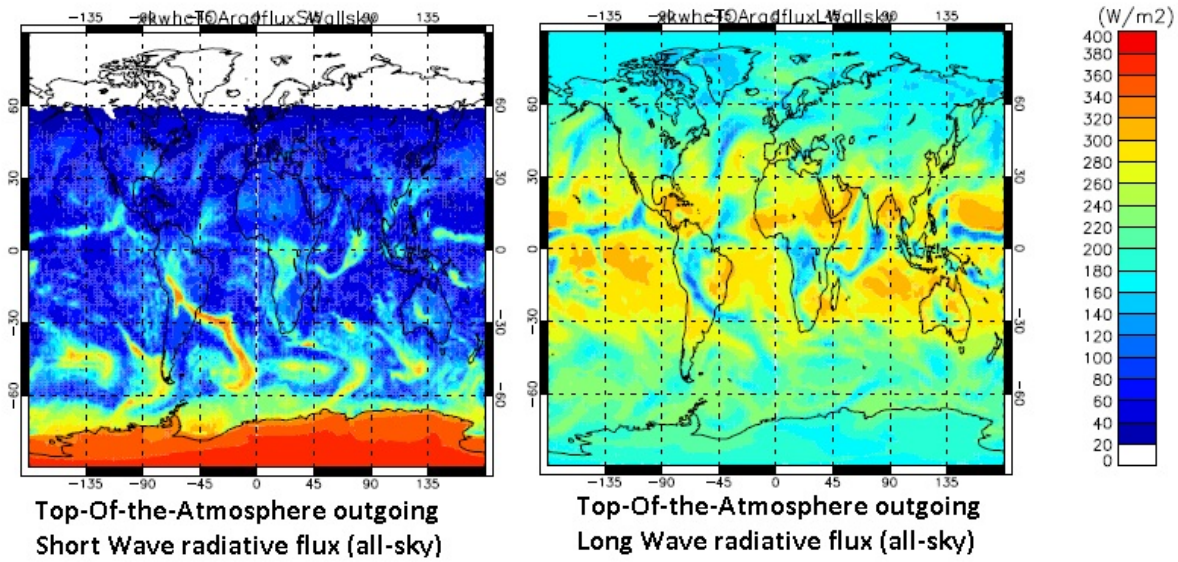
The user should note however that to illustrate the task we are adding these requests to the UKCA tutorial job which is just a 1-day simulation.

One would need to average the flux-difference between the pair of simulations over an appropriate timescale (e.g. multi-annual monthly-means) in order to diagnose an effective radiative forcing appropriately.

Noting the above caveat, proceed and add daily-mean STASH requests for section 1 item 208 (all-sky outgoing short wave flux at the top-of-the-atmosphere) and section 2 item 205 (all-sky outgoing long wave flux at the top-of-the-atmosphere) to your copy of the UKCA tutorial job (xkvxe).

The radiative fluxes are 2-dimensional diagnostics (longitude by latitude) so you should use the DIAG domain profile in this case. For daily-means use the TDAYM time profile. Again, since we require the daily-mean fluxes to be output to the .pa file you should request the diagnostics with the UPA usage profile.

The Figure below shows the daily-mean SW and LW all-sky TOA radiative fluxes from the UKCA tutorial job for 1st December 1999 (gmann job xkwhc).



Example Output

Example output for Task12.1 can be found on ARCHER in the following directory:

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Task12.1
```

Task 12.2 Configure the UKCA tutorial job to run as a *double-call* job diagnosing aerosol radiative effects

In this task you will copy your copy of the standard tutorial job (**`xjrnk`**) and configure it to run with *double-call* to the radiation scheme to diagnose the radiative effects of the aerosol simulated by GLOMAP in UM-UKCA.

The UM has been coded to allow the user to diagnose radiative effects of a particular forcing agent by calling the radiation scheme twice with one of the calls setting the agent's concentration to zero. Special *forcing* STASH items are included within the UM which store the difference in the radiative fluxes between the two radiation calls.

In the UMUI go to Atmosphere --> Scientific Parameters and Sections --> Section by section choices and then choose *Section 1: SW radiation.*

In the SW Radiation UMUI panel that opens, you see that at the top there is an "Options for multiple calls to radiation" button-selector.

The UKCA tutorial job is set to "Timestepping scheme" which is the recommended way of running the model. In this configuration the UM has a single call to the radiation scheme every radiation timestep (here 3 hours) with a 2nd reduced-call being applied on other timesteps (for more details see Manners et al., QJRMS 2009). The *single-call* option is the same as the Timestepping scheme but does not apply the reduced radiation call on interim timesteps.

The other option supported here is to select "Diagnose radiative forcings" which activates the *double-call* approach where the radiation scheme is called twice on each radiation timestep with and without the forcing agent.

By default, if one selects the *Diagnose radiative forcings* option, then the model diagnoses the radiative forcing based on the *advancing call* including the forcing agent as usual, and the species is set to zero in the 2nd *diagnostic call* to the radiation.

This operation is applied via the *SLWForc* panel which is available after selecting the *Gen2* follow-on window. See that it is possible to individually select each of the CLASSIC aerosol types to diagnose their radiative effects whereas for GLOMAP-mode it only makes sense to diagnose the effects over all the types considered since the different types become internally mixed within each size class. Note that the user needs to be very careful to specify exactly how the effects should be applied in the second radiation call and this is specified in the *Call2* follow-on window.

We strongly recommend only making changes to the default settings after discussing with relevant experts within NCAS or the Met Office.

Although the default UM setting for the double-call is to set the species mixing ratio to zero in the diagnostic call, it is often very useful to be able to suppress the fast feedbacks from the forcing agent in question by reversing the operation of the double-call including the aerosol radiative effects only in the diagnostic call and setting the species mixing ratio to zero in the advancing call.

With this *double-call radiative forcing* configuration, the difference in radiative fluxes between the two calls provides the aerosol radiative perturbation with respect to an atmosphere containing no aerosols. One can diagnose the present-day to pre-industrial aerosol radiative forcing by taking the difference between two parallel double-call simulations with aerosol and precursor emissions set to 1850 and 2000. All other forcing agents, such as greenhouse gases or land-use change, remain fixed at a reference time period.

Often nudging to meteorological re-analysis winds and temperatures is applied in tandem with the double-call configuration in which case the composition-climate model is being run in a similar way to an offline chemistry-transport model. This approach has been used extensively in aerosol forcing intercomparisons (e.g. the AeroCom direct forcing experiments, Myhre et al., 2013, ACP) with the radiative forcings diagnosed from each model with fast feedbacks disabled.

To run UM-UKCA with this *double-call radiative forcing* configuration, you will need to add an extra FCM branch to the job and also add an extra hand-edit in the UMUI to configure the double-call.

So first, in the FCM panel, add an entry to point to revision 17632 of the following FCM branch:

```
fcm:um-br/dev/gmann/vn8.4_RADAERupdates_for_dblcalaerforc/src
```

Then, in the hand-edits panel in the UMUI add the hand-edit to configure the double-call:

```
~gmann/umui_jobs/hand_edits/vn8.4/c2c_dustADE_glomapADEandAIE1_v84.ed
```

One of the things the hand-edit sets is the value of a switch C2C_AER_DIAGCAL which controls whether the setting of the forcing agent to zero is applied on the advancing call or the diagnostic call. You see that the hand-edit sets C2C_AER_DIAGCAL to .TRUE. so that the forcing agent is set to zero on the diagnostic call.

When the double-call forcing configuration is selected in the SW (or LW) radiation panels it synchronises to the setting in the LW (or SW) panel and a submitted job adds a series of other *C2C* switches to the *RADFCDIA* namelist in the file CNTLATM according to the buttons selected in the *SLWForc* panel.

In our *double-call forcing* job however, we will keep the *SLWForc" UMUI panel unchanged with all* C2C switched set to be false.

Instead we are using the above hand-edit to set the values of the C2C switches in CNTLATM.

If you view the c2c_dustADE_glomapADEandAIE1_v84.ed hand-edit you see it changes the C2C_DUST_D, C2C_UKCA_D and C2C_UKCA_I switches to be true so that the model is configured to diagnose the flux-difference (forcing) based on including the direct radiative effects from the total of the CLASSIC simulated dust combined with the direct and (1st) indirect effects from the GLOMAP simulated aerosol properties.

Configuring the radiation scheme for the double-call requires more than just selecting the *Diagnose radiative forcings* option.

You will need to update your job making also the following changes:

First, in the SW radiation window change the *Number of times a day ot calculate increments (Diagnostic)* from 24 to 8. This reverts the diagnostic call to only be carried out on radiation timesteps (3 hourly) rather than every hour as the reduced-radiation-call was applied in the timestepping configuration. Make the same change in the *Section 2: LW radiation window*.

Second, in the "Gen2" follow-on window for the SW radiation, change the *Diagnostic File* to be the same shortwave spectral file as for the *Prognostic File*. I.e. change *spec_sw_cloud3_0* to *spec_sw_ga3_0*. Do the same for the *Gen2* follow-on window under the LW radiation panel changing the *Diagnostic File* from *spec_lw_cloud3_0* to *spec_lw_ga3_0*.

Finally, in the *Call2* follow-on window (from the SW or LW Radiation panels) the radiation settings for the double-call need to be set to match those used in the main model. In order to achieve this, in the make the following changes:

- change the *method for representing horizontal water content variability* from *Homogeneous* to *McICA*.
- change the *option for overlapping clouds* from *Maximum-random* to *Exponential-random*.
- switch on the button to include SW absorption by O2
- switch on the buttons to include LW absorption by CFC113, CFC114, CFC11, CFC12, HCFC22, HCFC125, HFC134A, CH4 and N2O.

All that remains then is to add in the extra STASH requests for the double-call forcing diagnostics and to note a change to the operation of the Aerosol Optical Depth diagnostics.

The approach taken to index the STASH numbers for the radiative forcing items (the flux-difference between the two radiation calls) is to apply an offset of +200 to the item number to the corresponding item for the conventional radiative fluxes.

In task 12.1 we added STASH requests for the all-sky TOA outgoing SW and LW radiative fluxes which are referenced in STASH as section 1 item 208 and section 2 item 205.

To request the all-sky TOA outgoing SW and LW radiative forcings (between the two radiation calls) the corresponding item numbers are section 1 item 408 and section 2 item 405. Unfortunately however, at UM v8.4, section 2 item 405 is not available from the UMUI. In this task we will therefore request instead the clear-sky TOA outgoing SW and LW radiative forcing diagnostics (section 1 item 409 and section 2 item 406). Note also that one needs to request both the radiation flux and radiation forcing diagnostic in these runs so you should add 4 daily-mean STASH requests for section 1 items 209 and 409 and section 2 items 206 and 406.

Go to the *STASH Specification of Diagnostic requirements* window and add the daily-mean (TDAYM) requests for both of these flux-forcing pairs of diagnostics. As in Task 10.1, the domain profile should be set as DIAG and the usage profile as UPA.

Since we have now configured the model to run with GLOMAP aerosol set to zero in the advancing call, the conventional AOD diagnostics introduced in tutorial 10 (section 2 items 300 to 305) will now contain zero values when the model is run.

The UM therefore has a second set of AOD diagnostics giving the aerosol optical depth as calculated in the diagnostic call.

The approach for the double-call AOD diagnostics is the same as for the forcing diags, i.e. to apply an offset of +200 to the item number to find the corresponding AOD item in the diagnostic call.

The double-call GLOMAP AOD diagnostics are therefore found in section 2 items 500 to 505.

At v8.4 it is necessary to add an extra user-STASHmaster file to enable these STASH items to be requested in the job. In the *User-STASHmaster files. Diags, Progs & Ancils* window off the STASH panel you need to add in the following file

```
~gmann/stashfiles/dblecall_aods_only.stash
```

You will also need to add the hand-edit

```
~gmann/umui_jobs/hand_edits/vn8.4_nosulphateAOD.ed
```

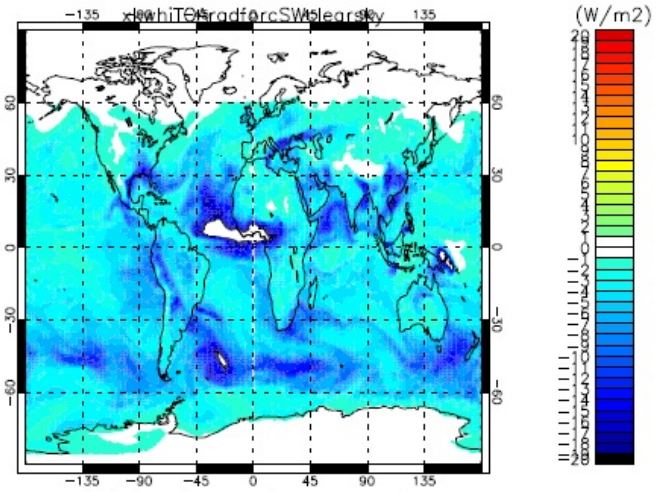
which removes the CLASSIC sulphate AOD as it causes a crash in the double-call forcing configuration. You should also remove this STASH number (2-284) as well as the mineral dust optical depth (2-285) from the STASH requests panel. This causes a crash on ARCHER, but not on MONSooN.

If you view that file you see that as well as providing the STASH settings for the GLOMAP double-call AOD diagnostics for each mode, it also provides the information to the UMUI to allow double-call AOD diagnostics to be requested for each of the CLASSIC aerosol types.

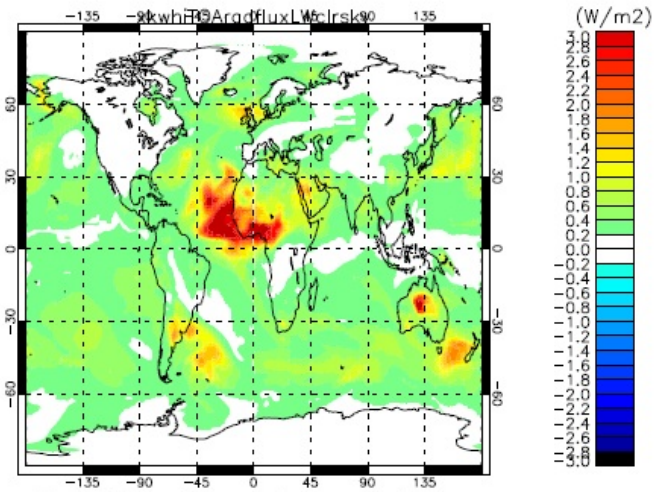
Once you have have added the *dblecall_aods_only.stash* user-STASHmaster file you should proceed to the *STASH Specification of Diagnostic requirements* window and add daily-mean (TDAYM) requests for the double-call GLOMAP AODs (section 2 items 500 to 505) with usage profile UPA and domain profile DIAG_AOT.

The simulation will then output daily-mean SW and LW clear-sky forcings and double-call AOD diagnostics to the .pa file for your UM-UKCA job.

The Figure below shows daily-mean TOA SW-clearsky and LW-clearsky radiative effect fields for the double-call-modified version of the UKCA tutorial (gmann job xkwhi).



Top-Of-the-Atmosphere outgoing
SW aerosol radiative effect (clear-sky)



Top-Of-the-Atmosphere outgoing
LW aerosol radiative effect (clear-sky)

Worked Solution

A worked solution to Task 12.2 can be found in job **`xjrnn`**. Sample output from a copy of this job can be found on ARCHER in the directory

```
/work/n02/n02/ukca/Tutorial/vn8.4/sample_output/Task12.2
```

Written by Graham Mann 2014

UKCA Chemistry and Aerosol Tutorial Rose Example

From UKCA

Back to UKCA Chemistry and Aerosol Tutorials

This tutorial will give you a brief example of a Rose job at vn10.2.

You should not use the UKCA Tutorial Rose Suite for scientific studies. It is designed to be used as a training aid.

Running Rose

In **PUMA**, do

```
rosie go --prefix=puma
```

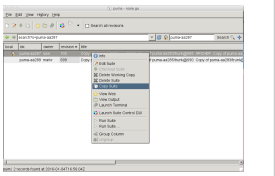
and in the search box search for **puma-aa287**.

You can then right-click and then copy this job. Click *Forward* and *OK* to do this.

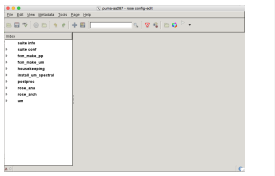
Double-click on this job and you should then have the Rose GUI open-up.

Rose is organised rather differently from the UMUI, but it is searchable. Search for **ncastr** and then put your NCAS training account in the correct box, then click the save icon.

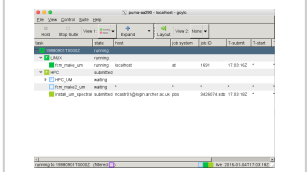
Now you will be able to run the job - click the **Play** icon (the right-facing triangle). The Cylc GUI will appear and show the progress of your job. You can click to expand all the steps.



Copying a Rose Suite.



The Rose GUI.



The Cylc GUI.

Questions

- **1)** What is the run length of this run?

Answer [hide]
1 day. This is defined by the **P1D** value ("plus one day"). Similarly, P1M would be a 1-month run, and P10Y4M would be a 10-year, 4 month run etc.

- **2)** What is the start date of this run?

Hint: Go to the **suite conf**: *Initialisation and cycling* panel. What do you see?

Answer [hide]
Midnight GMT on the 1st September 1988 (19880901T0000Z).

- **3)** Find the UKCA panel, and compare this to the UMUI UKCA panel. Are there any differences? What are they?
- **4)** Where is the data held on ARCHER?

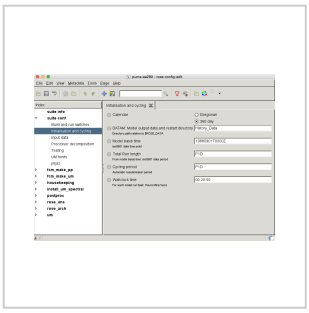
Answer [hide]
/work/n02/n02/$USER/cylc-run/puma-xxYYY/share/data/History_Data

- **5)** Can you find the ozone field that has been output?

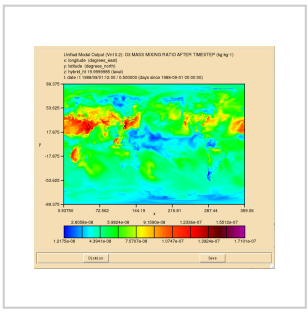
Answer [hide]
It is at the end of the xxYYYa.pa19880901 file.

- **6)** Compare this field to the ozone field from your tutorial job. What differences stand out?

Answer [hide]
Apart from the fact that the values are different, the grid is also different. This is because the vn8.4 job is New Dynamics (**192x145** gridpoints: -90.0 to 90.0 in steps of 1.25, and 0 to 358.125 in steps of 1.875), whereas the vn10.2 suite is based in GA6.0+ and uses ENDGAME (**192x144** gridpoints: -89.375 to 89.375 in steps of 1.25, and 0.9375 to 359.0625 in steps of 1.875).



The Rose initialisation and cycling panel.



Surface ozone from puma-aa287

- This page was last modified on 6 January 2016, at 16:13.