

This discussion paper is/has been under review for the journal Geoscientific Model Development (GMD). Please refer to the corresponding final paper in GMD if available.

The infrastructure MESSy submodels GRID (v1.0) and IMPORT (v1.0)

A. Kerkweg¹ and P. Jöckel²

¹Institut für Physik der Atmosphäre, Johannes Gutenberg-Universität Mainz,
55099 Mainz, Germany

²Deutsches Zentrum für Luft- und Raumfahrt (DLR), Institut für Physik der Atmosphäre,
82234 Oberpfaffenhofen, Germany

Received: 27 July 2015 – Accepted: 1 September 2015 – Published: 8 October 2015

Correspondence to: A. Kerkweg (kerkweg@uni-mainz.de)

Published by Copernicus Publications on behalf of the European Geosciences Union.

8607

Abstract

The coupling of Earth system model components, which work on different grids, into an Earth System Model (ESM) provokes the necessity to transfer data from one grid to another. Additionally, each of these model components might require data import onto its specific grid. Usually, one of two approaches is used: Either all input data is pre-processed to the employed grid, or the imported data is interpolated on-line, i.e. during model integration to the required grid. For the former, each change in the model resolution requires the re-preprocessing of all data. The latter option implies that in each model integration computing time is required for the grid mapping. If all components of an ESM use only one single point of import and the same mapping software, only one software package needs to be changed for code optimisation, inclusion of additional interpolation methods or the implementation of new data formats.

As the Modular Earth Submodel System (MESSy) is mainly used for research purposes which require frequent changes of the model setup including the model resolution or the application of different sets of input data (e.g., different emission scenarios), the idea of a common procedure for data import was implemented in MESSy in form of the infrastructure submodel IMPORT. Currently, IMPORT consists of two submodels: IMPORT_TS for reading and processing abstract time series data and IMPORT_GRID, utilising the infrastructure submodel GRID which provides procedures for grid transformations using the remapping software packages NREGRID (Jöckel, 2006) and SCRIP (Jones, 1999). Grid information is stored in a standardised structure as geo-hybrid grids. Based on this unified definition a standardised interface for the grid transformations is provided, thus simplifying the implementation of grid transformations in the model code.

This article describes the main functionalities of the two MESSy infrastructure submodels GRID and IMPORT. The Supplement of this article contains stand-alone tools of both IMPORT subsubmodels, IMPORT_TS and IMPORT_GRID. Their handling is explained in detail in the IMPORT User Manual which is also part of the Supplement.

8608

1 Introduction

An important part of Earth System Model (ESM) infrastructure is input and output of data (I/O). While the output is centralised in most models, input is often performed directly where it is needed, i.e., corresponding routines are spread throughout the model. Usually ESMs require lots of input, e.g., the land-sea mask, land types, leaf area indices or – for atmospheric chemistry models – emission maps, and so forth. For many models (e.g., ECHAM, COSMO, CESM1) these external data need to be preprocessed for each attributable model resolution and (in case of a regional model) domain, because the model requires the input data to be on its own grid. This is the fastest method with respect to model run-time performance and therefore might be the best solution for operational application of models, e.g., in weather prediction. In contrast to an operational model, for a research model a change of the input data is a frequent procedure, e.g., a change of the horizontal resolution or of an emission inventory. In terms of high flexibility of the system, it seems a more desirable design of a research model, to store all possible input data only in the finest available resolution, while the model itself transforms the data to the respective model grid. This approach is less storage space consumptive and more flexible as the repeated preprocessing of all required input data. On the other side, more computing time is required during the model integration for the on-line remapping. This was already implemented for the EMAC¹ model by the infrastructure submodel NCREGRID (Jöckel, 2006)², but NCREGRID was used for data import throughout the model, i.e., it was called individually by

¹The ECHAM/MESSy Atmospheric Chemistry (EMAC) model is a numerical chemistry and climate simulation system that includes sub-models describing tropospheric and middle atmosphere processes and their interaction with oceans, land and human influences (Jöckel et al., 2010). It uses the second version of the Modular Earth Submodel System (MESSy2) to link multi-institutional computer codes. The core atmospheric model is the 5th generation European Centre Hamburg general circulation model (ECHAM5, Roeckner et al., 2006).

²Note: The infrastructure submodel previously used in EMAC is named NCREGRID, while the remapping algorithm itself is called NREGRID.

8609

each submodel, which required off-line gridded input data. The new MESSy submodel IMPORT (v1.0) establishes one single point of data import for the entire model system. This has several advantages:

- It is much easier to keep track of the data imported, as all imported data is listed in one namelist.
- All data is handled consistently and the usage of additional or new import data is less error-prone.
- The outline of the model code is much clearer, as not each model part depending on input data needs to include importing and regridding routines.
- In case of import optimisation only one source code needs to be changed.
- For code extensions, i.e., introduction of new file formats or new mapping routines, the corresponding routines have to be added only at one point in the model code.

Figure 1 illustrates the current layout of the IMPORT and GRID submodels and their connection to other generic MESSy submodels. The generic submodels of MESSy are those submodels, which are not dedicated to simulate a specific process or diagnostic, but provide the technical infrastructure required by those regular submodels. Currently, IMPORT supports two different types of data: First, 0- or 1-dimensional (i.e. abstract) time series data are read by a standard interface in the IMPORT subsubmodel IMPORT_TS. Secondly, for gridded³ (time dependent or static) data the submodel IMPORT_GRID uses the transformation routines provided by the GRID submodel. GRID comprises the in EMAC well established standard mapping tool NREGRID (Jöckel, 2006) and an implementation of the SCRIP software (Jones, 1999) providing transformations to/from curvi-linear or unstructured grids.

In order to unify the usage of the different mapping softwares, the so-called “geo-hybrid grid” structure, as already defined in NCREGRID, is extended and conversion

³Gridded here implies geo-referenced.

8610

routines between the grid definitions required by NREGRID and SCRIP are provided. The mapping routines are not only used during data import, but also for grid transformations within the model, e.g., for mapping between the ocean and the atmospheric grid when MPIOM is used as a MESSy submodel in EMAC (Pozzer et al., 2011), or for the 2-way on-line coupling between EMAC and COSMO/MESSy (Kerkweg and Jöckel, 2012b). Thus, the submodel GRID provides the required grid definition and grid handling routines and its submodel GRID_TRAFO contains the transformation routines including standard interfaces for the usage of these routines.

In the following the functionality of the generic MESSy submodels GRID and IMPORT are described. Information about the general usage of these submodels is provided here. Further details and more technical information, required for model developers to implement the remapping routines into their own code, are supplied in the user manuals in the Supplement of this article. Section 2 gives an overview of the GRID submodel. Section 3 describes the IMPORT submodel. IMPORT consists of two submodels: the first imports time dependent or static gridded data automatically transforming them to a target grid. A general view of this subsubmodel is given in Sect. 3.1 including examples for gridded data import. The second one imports abstract time series data and is described in Sect. 3.2. Both sections explain the respective namelists and show how the submodels are to be used in a simulation. Before the summary an overview of the history of data import in MESSy is provided (Sect. 4).

2 The generic MESSy submodel GRID

The generic MESSy submodel GRID builds the basis for all required grid transformations. Most of its internal data types follow the netCDF data format definitions. The hierarchical data structures follow mostly those of NCREGRID (Jöckel, 2006).

The submodel core layer (SMCL) of GRID contains the definition of the geo-hybrid grid structure, i.e. a grid defined horizontally by geographical longitude and latitude and vertically by hybrid pressure coefficients. The structure provides all information re-

8611

quired for the grid conversion. For different types of grids different containers for the definition of the horizontal grid are specified. The remapping algorithms automatically applies the correct conversion routines, depending on the containers filled. The details are explained in the “GRID User Manual”, which is part of the Supplement. The GRID SMCL routines also comprise subroutines for the handling of the grid structures, i.e., routines for initialising, copying, importing, exporting and printing a variable of the grid structure type. Beyond that, routines necessary for defining a grid, storing it in a concatenated list, locating an already defined grid within this list, and for comparing grids are part of the GRID SMCL.

2.1 GRID_TRAFO

The main target of the GRID submodel is to provide routines for the transformation of gridded geo-located data. So far, two different transformation algorithms are part of GRID_TRAFO: NREGRID and SCRIP. While the core mapping algorithms differ, GRID_TRAFO provides unified interfaces for the conversion between different grids.

2.1.1 NREGRID

NREGRID, the mapping algorithm and the core of NCREGRID, is a recursive algorithm, which is applicable to arbitrary orthogonal (including curvi-linear) grids of any dimension. The algorithm does not apply a point-to-point interpolation, but a transformation based on overlaps between the different grid volumes. Details about the algorithm applied have been published by Jöckel (2006).

2.1.2 SCRIP

As NREGRID is limited to the remapping between orthogonal grids, the implementation of an algorithm able to interpolate between different curvi-linear or even unstruc-

tured grids became necessary. To reach this aim the SCRIP software⁴ (Jones, 1999) version 1.4 provided by the Los Alamos National Laboratory has been utilised. SCRIP (a Spherical Coordinate Remapping and Interpolation Package) “is a software package used to generate interpolation weights for remapping fields from one grid to another in spherical geometry. The package currently supports four types of remappings. The first is a conservative remapping scheme that is ideally suited to a coupled model context where the area-integrated field (e.g. water or heat flux) must be conserved. The second type of mapping is a basic bilinear interpolation which has been slightly generalized to perform a local bilinear interpolation. A third method is a bicubic interpolation similar to the bilinear method. The last type of remapping is a distance-weighted average of nearest-neighbor points. The bilinear and bicubic schemes can only be used with logically-rectangular grids; the other two methods can be used for any grid in spherical coordinates.” (Quoted from: SCRIP Users Guide, Introduction, Jones, 1998)

Unfortunately, SCRIP is only a software for horizontal grid transformation. The easiest way to add vertical remapping is to use NREGRID for the vertical grid transformation, after the horizontal remapping via SCRIP is conducted. Additional vertical interpolation schemes can be easily added in the future.

2.2 The basemodel interface layer of GRID

The backbone of each model is its grid, e.g., for an atmospheric model, the horizontal space is given by a definition of the longitudes and latitudes of the models grid midpoints and the grid corners. The vertical space is defined by a height or pressure coordinate. As this grid is the reference for most submodels and processes, this grid is defined in the basemodel interface layer (BMIL) for the usage in all MESSy submodels. Most importantly, it is used by IMPORT as the default target grid for data import. Additionally, the BMIL of GRID allows to broadcast the geo-hybrid grid structure. This

⁴<http://oceans11.lanl.gov/trac/SCRIP> (last access: 23 June 2015). The official link named in the SCRIP users guide (<http://climate.acl.lanl.gov/software/SCRIP>) is not available any more.

8613

is required if a geo-hybrid grid is initially defined on one parallel task, but is used by all parallel tasks later in the simulation.

3 The generic MESSy submodel IMPORT

IMPORT supplies MESSy with a standardised interface for data import. So far IMPORT includes submodels for import of abstract time series data (IMPORT_TS) and for gridded (time-dependent or static) data (IMPORT_GRID). If required, IMPORT can be easily expanded by additional subsubmodels to import other data representations. In this way, all data traffic into the model is managed by IMPORT, while the generic MESSy submodel CHANNEL (see Jöckel et al., 2010, and the Supplement thereof) performs the model wide memory management including the output. Centralising input and output of a model facilitates the optimisation of the model performance, as in- and output are time critical, especially regarding scaling of the model performance with the number of parallel tasks.

Both, IMPORT_TS and IMPORT_GRID are namelist controlled. The following sections give an overview of the submodels and explain basic setups of the IMPORT namelists. Further details about the namelist settings and additional information for model developers are provided in the “IMPORT User Manual”, which is part of the Supplement of this article.

3.1 IMPORT_GRID

Currently, two horizontal mapping algorithms (NREGRID and SCRIP) are available in IMPORT_GRID. The default scheme depends on the basemodel, e.g., for the regional COSMO/MESSy model SCRIP is automatically chosen, as the COSMO model domain is usually defined on a rotated grid and thus NREGRID is not applicable. NREGRID is the default for the global EMAC model.

The imported data is made available to the other MESSy submodels as CHANNEL objects using the data infrastructure submodel CHANNEL⁵. Additional meta-information (e.g., units, emission heights) available along with the imported data are forwarded to other submodels in form of attributes of the respective CHANNEL objects (see Fig. 1).

As default, IMPORT_GRID assumes the basemodel grid to be the target grid for the imported data. This grid is defined in the BMIL of GRID (Sect. 2.2). But the destination grid can be changed by namelist entry. In this case the target grid has to be defined by any other submodel using it (e.g., an ocean or a finer land surface grid). For the definition, a unique name is assigned to the grid. This name is required to specify the target grid in the IMPORT_GRID namelist. All grid definitions are internally stored in a concatenated list. If a grid name is given in the namelist, the concatenated list of geo-hybrid grids is searched, and the stored grid is used as target grid for the grid transformation.

3.1.1 The IMPORT_GRID namelist

The mechanism driving IMPORT_GRID is the same as described for OFFLEM (Kerkweg et al., 2006b) and NCREGRID (Jöckel, 2006). In the namelist file `import.nml` a list of “regrid events” (&RGTEVENTS) is given. The User Manual to OFFLEM in the Supplement of Kerkweg et al. (2006b) and the Supplement of this article contain a detailed description. Here, we provide only a simple example:

```
&RGTEVENTS
!
! CH3I
!
! TIMER          NAME      COUNTER  ACTION STRING
RG_TRIG(1) = 1, 'months', 'first', 0, 'EMIS1', 13, 1, 24, 2, 'NML=CH3I_20001-200012.nml;',
/
```

Each namelist entry consists of four different parts: the TIMER information, the name, the counter and the action string.

⁵For more information see Jöckel et al. (2010) and the Supplement thereof.

- The TIMER information directly relates to the definition of an *event* as defined in the MESSy submodel TIMER (see Jöckel et al., 2010, and the Supplement thereof). In this example the event (import of methyl iodine emissions) is triggered each month.
- The name, here ‘EMIS1’, will be the first part of the name of the CHANNEL object as defined by IMPORT. The second part (appended after an underscore) is given by the name of the regridded variable (see below).
- The counter provides the information which time steps from the data file are to be read. In this example, the second time step would be read at model start, subsequently the time step is increased by 1 until it reaches 24. Afterwards, the program continues with step 13 asf.
- The action string contains the information for the remapping process. In the example, it only contains the name of the namelist file to be processed by IMPORT_GRID (see below). Likewise, the name of the regridding algorithm and the name of the target grid is given here, if the defaults should not be used. The remapping algorithm can be changed by naming the interpolation method (IPOL) in the action string. The following methods are available:
 - IPOL=NRGT selects NREGRID.
 - IPOL=SCRIP selects SCRIP.
 - IPOL=NONE imports the data on its original grid as defined in the input file.

Furthermore, adding the identifier ‘GRID=MYGRID’ results in remapping to a grid named “MYGRID” defined by some MESSy submodel. All possible contents of the action string are listed in the IMPORT User Manual in the Supplement of this article.

The regridding namelist for the example is:

```

&regrid
infile      = 'CH3I_20001-200012.nc',
i_tmem      = 'MONTH_REG',
i_latm      = 'LAT',
5 i_latr     = -90.0,90.0,
i_lonm      = 'LON',
i_lonr      = -180.0,180.0,
var         = 'CH3I=E_CH3I,0.5',
/

```

10 The individual regridding namelists contain

- the name and the path of the input file (`infile`),
- the name of the time variable (`i_tmem`),
- the names of the longitude (`i_lonm`) and the latitude (`i_latm`) variables,
- the respective ranges of the longitude and latitude axes (`i_lonr` and `i_latr`) and
- 15 – the quantity to be remapped from the file, including an optional scaling factor and a new name. Here, the original field `E_CH3I` is regridded, scaled by the factor 0.5 and renamed to `CH3I` (resulting in a channel object “EMIS1_CH3I” in the `IMPORT_GRID` channel).

20 3.1.2 Examples: import of gridded data

Figure 2 shows an example of mapping to a global Gaussian grid. The upper left panel displays the original field read in by `IMPORT_GRID`. This field has a resolution of $1^\circ \times 1^\circ$. The middle and right columns display the results after regridding with `NREGRID` and `SCRIP`, respectively, to a T106 (upper) and a T42 (lower) grid. The fields after transformation by `NREGRID` and `SCRIP`, respectively, look very similar. The differences

8617

between the T42 fields resulting from `NREGRID` and `SCRIP` remapping (lower left) are 5 orders of magnitude smaller than the data.

For regional rotated grids only `SCRIP` is applicable. Figure 3 displays an original road traffic CO emissions field (upper left), which is in $0.5^\circ \times 0.5^\circ$ resolution. After remapping

5 (by `SCRIP`) to a T42 grid (Fig. 3, upper right) the peak values are much smaller, as to be expected for the coarser grid. Due to the rotation of the COSMO model grid even by mapping to a grid of 0.36° resolution, the peak values are decreased, while regridding to the much smaller 0.0625° rotated COSMO grid keeps the extreme values. Due to the rotation of the grid, the borders of the original grid boxes look somewhat blurred.

10 3.1.3 Special case: tracer initialisation

In order to reduce the memory consumption, we currently deviate from our strategy of one single point of import for the initialisation of tracers. It is advantageous to initialise the tracers in the BMIL of the tracer submodel. Here, the full tracer structure is straightforwardly accessible. Additionally, tracer initialisation is only required at model start and

15 therefore the time event control of `IMPORT_GRID` is not required here.

As illustrated in Fig. 4, even though we do not use the BMIL routines of `IMPORT_GRID`, we do use the `IMPORT_GRID` SMCL routines for importing the files and the `GRID_TRAFO` routines for regridding them. In this way we minimise the additional overhead of an additional import implementation.

20 3.1.4 Parallelisation

In principle, the processing of the data (import and remapping) can proceed in parallel. Depending on the calling model different methods are applicable. In case of a stand-alone tools parallelisation is possible but is not necessarily required. For 3-D models parallel domain decomposition can be used, i.e., each parallel task processes the data

25 required for its respective part of the model domain. For `IMPORT_GRID` this is the case for the COSMO model. In models with a more complex domain decomposition (e.g.,

ECHAM5) this is not straightforwardly applicable. Therefore, no parallelisation is applied unless the number of variables contained in one file is large enough. In this case parallelisation takes place over the number of variables. This is the case for the tracer initialisation in EMAC. If possible, parallelisation over the domain is preferred over parallelisation over variable number. Therefore, the tracer initialisation in COSMO/MESSy is parallelised over the domain.

3.2 IMPORT_TS

The IMPORT submodel IMPORT_TS reads standardised abstract time series data from ASCII or netCDF files. Time series data generally consist of an equidistant time axis and a parameter axis. The time axis covers data defined annually, monthly, daily, hourly, every minute or every second. The parameter axis can be freely chosen. It may consist of a number of vertical levels or be just a collection of different data. For example, the parameter axis of a radio sonde measurement could be

```
lat[deg] lon[deg] height[hPa] O3[ppb] temperature[K] .
```

At the beginning of a simulation, the file is read (i.e., all time steps and parameters). During the simulation the data is processed according to the namelist entries: for simulation dates, which do not exactly match the times provided by the input data, the available data is interpolated to the current date by using the previous or the next point in time, or by interpolating linearly between the two nearest points in time. For more details see Sect. 3.2.2 and the user manual in the Supplement of this article.

3.2.1 Layout of the input data file

If the data is available as an ASCII file (see example in Fig. 5), the file consists of an eight line-header and a data table. The first three header lines provide the required information about the source of the data. The fourth line explains the syntax of the following lines. The fifth line defines the temporal layout of the data file. It contains four integers:

8619

1. the flag for the time interval used,
2. the start year,
3. the end year of the data and
4. the number of parameters (columns of the table except time information).

In the example in Fig. 5 the data is provided annually. The first data point is for the year 2010 and the last for the year 2014. The sixth line should comprise a description of the parameter axis. In the example in Fig. 5 this is “levels in hPa”. The actual heights are given in the seventh line. The final header line acts as a separator between header and data. The table, which follows, contains in each line the data for one specific point in time. Depending on the chosen time interval, the first one to six columns contain the date specification (year, month, day, hour, minute, seconds). For yearly data this results in only one column, while for hourly data four columns are required, and so forth. Subsequently, the parameter axis columns (4 in the example in Fig. 5) follow.

In a netCDF file, the information about the data origin is stored in attributes. Additionally, the composition of the parameter axis should be contained in an attribute describing the parameter axis. For a netCDF file, the interval of the time axis is detected by the analysis of the time unit and the time coordinate variable. The length of the parameter axis is determined automatically from its dimension. Afterwards, the data set (time \times parameter dimension) is read at once.

3.2.2 The IMPORT_TS namelist

IMPORT_TS is driven by the &CTRL_TS namelist. See Fig. 6 for an example. Each TS entry describes one time series data set. The meaning of the components is:

- The first string defines the name of the time series data set and thus the name of the CHANNEL object containing the finally processed data. By means of this name the data can be accessed in other parts of the model.

8620

- The second string comprises the name, including the full path, of the data file. Only for netCDF files, the string contains the name of the variable to be read. Its name has to be given at the beginning of the string and is separated from the filename by an @-sign. In the example in Fig. 6 `TS(1)` defines the time series data named 'exnc'. The variable in the netCDF file is named "EXNC", while the data file is found under `/DATA/exnc/EXNC_1950_2012.nc`.
- The next two float entries determine the valid range of the data. In case of `TS(1)` in Fig. 6 this is between `-99.9` and `99.9`. The default valid range is between `-HUGE(0.)` and `HUGE(0.)`⁶.
- The next two integer variables set the valid time range for the time series data, i.e., if data is provided in cases where the simulation date lies outside of the time span covered by the data file. If set to "0" the model execution is stopped, where as "1" allows for the continuation of the simulation. In the second case, the data of the nearest point in time present in the file is used. As the desired policy may differ for dates before and after the covered time span, the first integer determines the method used for dates prior to the time span comprised in the file, and the second integer the method used after the provided time span. In the example (Fig. 6) the simulation would be stopped, if a date outside the time frame covered by the exnc file (`TS(1)`) is reached. For `TS(2)`, the simulation will be continued after 1990, as the second integer flag is set to 1. In this case, `IMPORT_TS` would provide the data for 1990 for all dates later than 1990.
- The third integer defines the mapping method for time steps in between the points in time defined by the time series data.
 - 1 : The previous point in time is used.
 - 0 : A linear interpolation between the two nearest points in time is performed.

⁶Fortran intrinsic

1 : The next point in time is used.

In Fig. 6 the data for "exnc" is linearly interpolated, while for `TS(2)` the previous point in time is used.

- The following six integers allow for the selection of a specific date or a specific time span of the data file. The order of entries is `year, month, day, hour, minute, second`. If all six variables are defined one specific date is used independent of the simulation date. If, for example, only the year has been set for a monthly data set, `IMPORT_TS` cycles over the 12 months of this specific year. Note: the other entries are always deduced from the current date. Thus a simulation using a monthly data set and cycling through one specific year (e.g., 1989 as for `TS(2)`) starting in June would at model start correctly use the data for June. Likewise, it is possible to use, e.g., only 12:00 UTC data of an hourly data set. By default, i.e., all six variables are not set, the data is selected according to the actual simulation date.
- The last float variable defines an offset. The unit of this offset is *days*. With this entry the whole time series can be shifted by a fixed time interval. Thus, for a daily data set defined at 00:00 UTC, an offset of 0.5 would trigger the usage of new data at 12:00 UTC instead of 00:00 UTC.

3.3 Stand-alone tools of `IMPORT_GRID` and `IMPORT_TS`

- Stand-alone tools of `IMPORT_GRID` and `IMPORT_TS` are part of the electronic Supplement⁷. The stand-alone tools can be used for off-line regridding. Furthermore, they are helpful tools, to test if a new file and/or the namelist settings are correct. The `IMPORT` User Manual (also part of the Supplement) contains a detailed description how to compile and run the stand-alone tools of `IMPORT_GRID` and `IMPORT_TS`, respectively.

⁷import_grid.zip and import_ts.zip

4 The history of data import in MESSy

Historically, each submodel in MESSy performed its own data import. At the beginning, when MESSy was only connected to ECHAM5 (Jöckel et al., 2006), import of 2-D or 3-D gridded data was based on NCREGRID (Jöckel, 2006). In particular, the submodels OFFLEM, ONLEM and DRYDEP (Kerkweg et al., 2006b, a) for the calculation of offline emissions, online emission and dry deposition of gases and aerosol particles, respectively, used to perform the largest part of the data import for an atmospheric chemistry simulation. Over time, OFFLEM was more and more (mis-)used as an interface for data import also for other submodels. This was pushed by the introduction of the CHANNEL interface (Jöckel et al., 2010), as the coupling of data objects was standardised within MESSy and the meta-information exchange via data attributes improved. Thus the usage of OFFLEM as import interface avoided the necessity to call the data import routines from every single submodel.

The implementation of MESSy into the regional weather prediction and climate model COSMO (Doms and Schättler, 1999; Rockel et al., 2008; Kerkweg and Jöckel, 2012a) and the development of the 1-way on-line coupling of COSMO/MESSy instances to EMAC, i.e., of the MECO(n) system⁸ (Kerkweg and Jöckel, 2012b), led inevitably to a clean split between a generic submodel for data import (IMPORT) and the process submodels: OFFLEM, ONLEM and DRYDEP have been replaced by the corresponding submodels OFFLEMIS, ONLEMIS and DDEP, respectively, which now calculate the respective processes exactly in the same way as their predecessors, but do not import data anymore. This was required, as the 1-way on-line coupling provided a new way of data provision. In a first development stage of MECO(n) all emission data was imported from the global model. This construct was due to the fact, that data import in MESSy so far was performed by NCREGRID, only. But NCREGRID is not applicable for transformation between non-orthogonal grids, thus, in this first stage no direct data import to the rotated regional COSMO model grid was possible. Obviously,

⁸MECO(n) = MESSy-fied ECHAM and COSMO models nested n-times

8623

this stimulated the implementation of routines able to transform gridded data between geographically-rectangular, rotated and limited-area grids. Thus, SCRIP (Jones, 1999) was implemented as alternative to NCREGRID, not only for data import and regridding between non-orthogonal grids, but also as grid transformation interface for intra-model data remapping, which was also required for the implementation of a 2-way on-line coupling of COSMO instances and EMAC within the framework of the MiKlip⁹ project FLAGSHIP.

5 Summary

This article gives a short overview of the generic MESSy submodels GRID and IMPORT. GRID provides a standard interface for transformations between different grids. Currently, two regridding software packages are available: the, in EMAC well established, NCREGRID algorithm (Jöckel, 2006) is limited to remapping between orthogonal grids. Therefore, SCRIP (Jones, 1999) was implemented as part of GRID, allowing also for transformations between non-orthogonal (i.e., curvi-linear or even unstructured) grids in MESSy. The interface routines provided bei GRID_TRAFO simplify the use of GRID for data remapping internally within a MESSy-fied model: e.g., from the EMAC to the COSMO model grid and vice versa. Furthermore, these interface subroutines are used to transfer data from input files to the respective (namelist given) target grid within IMPORT_GRID.

The generic MESSy submodel IMPORT establishes a single point for data import into a MESSy model. Currently, IMPORT consists of two submodels:

- IMPORT_GRID to read and remap gridded data from netCDF-files and
- IMPORT_TS to read in and process abstract time series data.

⁹<http://www.fona-miklip.de>

If import of an additional data representation or the implementation of a new data format is required, IMPORT can be easily expanded by further subsubmodels.

Code availability

The code described here is part of the Modular Earth Submodel System (MESSy), which is continuously further developed and applied by a consortium of institutions. The usage of MESSy and access to the source code is licenced to all affiliates of institutions which are members of the MESSy Consortium. Institutions can be a member of the MESSy Consortium by signing the MESSy Memorandum of Understanding. More information can be found on the MESSy Consortium Website (www.messy-interface.org). While the files of the basemodel interface layer are only available (and useful) for members of the MESSy Consortium, the stand-alone tools of IMPORT_GRID (including the SMCL of GRID) and IMPORT_TS are available in the electronical Supplement of this article.

The Supplement related to this article is available online at

doi:10.5194/gmdd-8-8607-2015-supplement.

Acknowledgements. The work was financed by the German Ministry of Education and Research (BMBF) in the framework of the MiKlip (Mittelfristige Klimaprognose/Decadal Prediction) subproject FLAGSHIP (Feedback of a Limited-Area model to the Global-Scale implemented for Hindcasts and Projections, funding ID 01LP1127A). We are grateful to Mariano Mertens (DLR) for testing and improving IMPORT_GRID for application in COSMO/MESSy. We thank Bastian Kern (DLR) and Andrea Pozzer (MPIC) for fruitful discussions concerning the application of SCRIP for remapping between the EMAC and the MPIOM grid. The authors acknowledge use of the Ferret program for the graphics in this paper. Ferret is a product of NOAA's Pacific Marine Environmental Laboratory (information is available at <http://www.ferret.noaa.gov>).

8625

References

- Doms, G. and Schättler, U.: The nonhydrostatic limited-area model LM of DWD. Part 1: Scientific documentation, Deutscher Wetterdienst, Offenbach, Germany, available at: www.cosmo-model.org/content/model/documentation/core/default.htm (last access: 8 October 2015), 1999. 8623
- Jöckel, P.: Technical note: Recursive rediscretisation of geo-scientific data in the Modular Earth Submodel System (MESSy), *Atmos. Chem. Phys.*, 6, 3557–3562, doi:10.5194/acp-6-3557-2006, 2006. 8608, 8609, 8610, 8611, 8612, 8615, 8623, 8624
- Jöckel, P., Tost, H., Pozzer, A., Brühl, C., Buchholz, J., Ganzeveld, L., Hoor, P., Kerkweg, A., Lawrence, M. G., Sander, R., Steil, B., Stiller, G., Tanarhte, M., Taraborrelli, D., van Aardenne, J., and Lelieveld, J.: The atmospheric chemistry general circulation model ECHAM5/MESSy1: consistent simulation of ozone from the surface to the mesosphere, *Atmos. Chem. Phys.*, 6, 5067–5104, doi:10.5194/acp-6-5067-2006, 2006. 8623
- Jöckel, P., Kerkweg, A., Pozzer, A., Sander, R., Tost, H., Riede, H., Baumgaertner, A., Gromov, S., and Kern, B.: Development cycle 2 of the Modular Earth Submodel System (MESSy2), *Geosci. Model Dev.*, 3, 717–752, doi:10.5194/gmd-3-717-2010, 2010. 8609, 8614, 8615, 8616, 8623
- Jones, P.: First- and second-order conservative remapping schemes for grids in spherical coordinates, *Mon. Weather Rev.*, 127, 2204–2210, 1999. 8608, 8610, 8613, 8624
- Jones, P. W.: A User's Guide for SCRIP: A Spherical Coordinate Remapping and Interpolation Package, Version 1.4, Theoretical Division Los Alamos National Laboratory, The User's Guide is part of the downloadable SCRIP code distribution, available at: <http://oceans11.lanl.gov/trac/SCRIP> (last access: 8 October 2015), 1998. 8613
- Kerkweg, A. and Jöckel, P.: The 1-way on-line coupled atmospheric chemistry model system MECO(n) – Part 1: Description of the limited-area atmospheric chemistry model COSMO/MESSy, *Geosci. Model Dev.*, 5, 87–110, doi:10.5194/gmd-5-87-2012, 2012a. 8623
- Kerkweg, A. and Jöckel, P.: The 1-way on-line coupled atmospheric chemistry model system MECO(n) – Part 2: On-line coupling with the Multi-Model-Driver (MMD), *Geosci. Model Dev.*, 5, 111–128, doi:10.5194/gmd-5-111-2012, 2012b. 8611, 8623
- Kerkweg, A., Buchholz, J., Ganzeveld, L., Pozzer, A., Tost, H., and Jöckel, P.: Technical Note: An implementation of the dry removal processes DRY DEPosition and SEDimenta-

8626

- tion in the Modular Earth Submodel System (MESSy), Atmos. Chem. Phys., 6, 4617–4632, doi:10.5194/acp-6-4617-2006, 2006a. 8623
- Kerkweg, A., Sander, R., Tost, H., and Jöckel, P.: Technical note: Implementation of prescribed (OFFLEM), calculated (ONLEM), and pseudo-emissions (TNUDGE) of chemical species in the Modular Earth Submodel System (MESSy), Atmos. Chem. Phys., 6, 3603–3609, doi:10.5194/acp-6-3603-2006, 2006b. 8615, 8623
- Pozzer, A., Jöckel, P., Kern, B., and Haak, H.: The Atmosphere-Ocean General Circulation Model EMAC-MPIOM, Geosci. Model Dev., 4, 771–784, doi:10.5194/gmd-4-771-2011, 2011. 8611
- 10 Rockel, B., Will, A., and Hense, A.: The regional climate model COSMO-CLM (CCLM), Meteorol. Z., 17, 347–348, 2008. 8623
- Roeckner, E., Brokopf, R., Esch, M., Giorgetta, M., Hagemann, S., Kornblueh, L., Manzini, E., Schlese, U., and Schulzweida, U.: Sensitivity of simulated climate to horizontal and vertical resolution in the ECHAM5 atmosphere model, J. Climate, 19, 3771–3791, 2006. 8609

8627

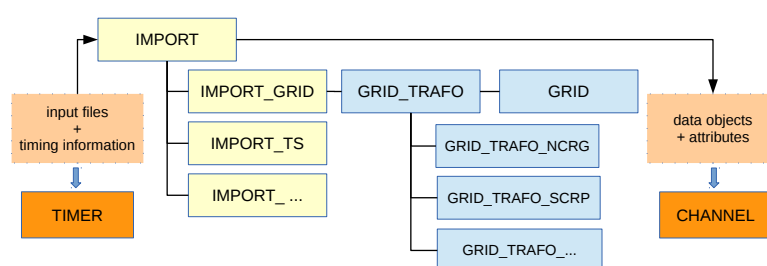


Figure 1. Structure of the generic MESSy submodels GRID (blue) and IMPORT (yellow). The orange boxes indicate the connections between IMPORT and the generic MESSy submodels TIMER and CHANNEL. Each of the boxes stand for one or more subsubmodels. IMPORT comprises the subsubmodels IMPORT_GRID and IMPORT_TS. Additional import subsubmodels (IMPORT_...) can be easily added in the future. IMPORT_GRID utilises GRID_TRAFO. This subsubmodel of GRID depends on the grid definition and handling routines of GRID and provides access to different remapping algorithms (GRID_TRAFO_NRG, GRID_TRAFO_SCRP). In future additional algorithms can be easily added (GRID_TRAFO_...).

8628

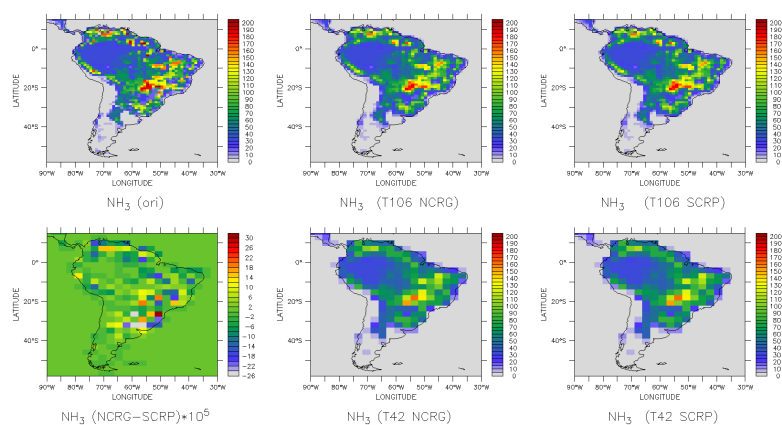


Figure 2. NH_3 emission flux in $10^{12} \text{ molec m}^{-2} \text{ s}^{-1}$. The original field in 1° resolution is shown in the upper left panel. The middle column displays the results after remapping by NREGRID to a T106 (upper) and a T42 (lower) grid, respectively, while the right column visualises the result using the SCRIP conservative remapping method. The lower left panel displays the difference of T42 fields resulting from NREGRID and SCRIP. Note, that the differences are five orders of magnitude smaller than the data.

8629

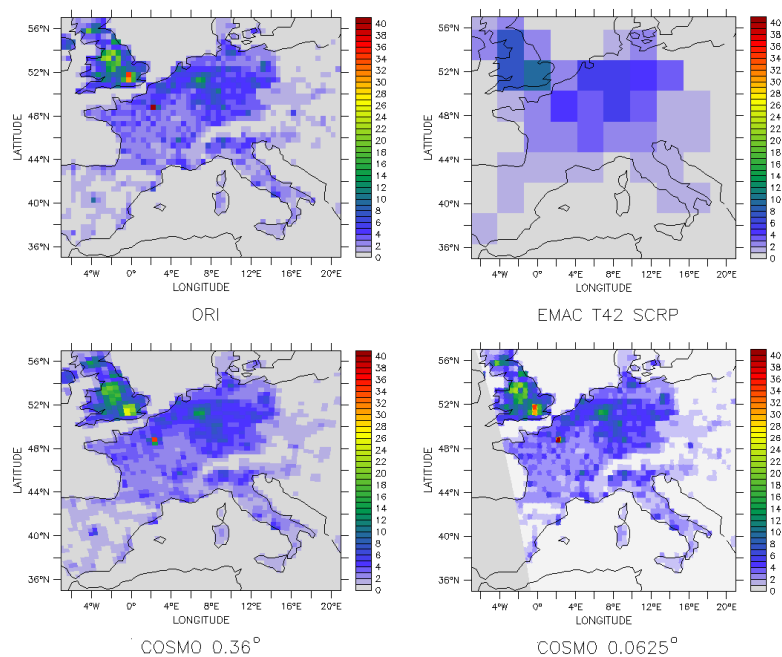


Figure 3. Road traffic CO emission flux in $10^{12} \text{ molec m}^{-2} \text{ s}^{-1}$. The upper left plot shows the original field in 0.5° resolution. The upper right displays the field mapped on a T42 grid, while the lower right and the lower left visualise the result of a remapping to rotated regional grids with 0.36° and 0.0625° resolution, respectively.

8630

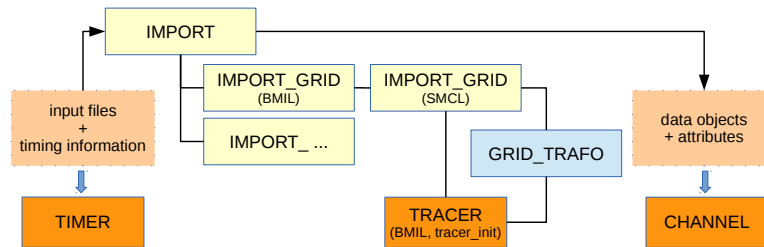


Figure 4. Structure of the generic MESSy submodels IMPORT_GRID and its connections to other generic submodels based on Fig. 1. The figure shows a zoom on the IMPORT_GRID submodel illustrating the linkage between IMPORT_GRID, GRID and TRACER.

8631

```

### EXAMPLE DATA SET CREATED BY A. KERKWEIG
### showing mean number of trajectories
### -9999.9 is flag for undefined value
### flag(1=annually,2=monthly,3=daily,4=hourly), 1st year, last year, columns
1 2010 2014 4
### Parameter Axis: levels [hPa]
1000.0 850.0 550.0 200.0
### data (YYYY 4 parameters)
2010 -9999.9 10.0 100.2 33.0
2011 200.0 113.2 53.7 99.1
2012 17.2 -9999.9 13.4 520.9
2013 965.9 0.0 34.2 13.8
2014 78.9 23.6 343.5 -9999.9

```

Figure 5. Example for an ASCII data file for IMPORT_TS.

8632

```

&CTRL_TS
! ### SYNTAX:
!   - name of time series
!   - [var@] name (incl. path) of data file
!       .nc -> netCDF, e.g., "var@my_path_to_my_file/my_file.nc"
!       -> ASCII, e.g., "my_path_to_my_file/my_file.txt"
!   - valid range ( default: -HUGE(0._dp), HUGE(0._dp) )
!   - out of time interval policy: 0: stop; 1: continue with nearest ...
!       ... (before time interval, after time interval)
!   - interpolation method: -1: previous; 0: linear interpolation; 1: next
!   - yr,mo,dy,hr,mi,se : pick out always this date/time
!       (example: 2000, , , , , will cycle through the year 2000 etc.)
!   - offset (in days)
!
! ### EXAMPLE netCDF ###
TS(1) = 'exnc', 'EXNC@/DATA/exnc/EXNC_1950_2012.nc', -99.90, 99.90, 0, 0, 0, , , , , 0.0,
!
! ### EXAMPLE ASCII ###
TS(2) = 'exascii', '/DATA/example/misc/ex_1985-1990.txt', , , 0, 1, -1, 1989, , , , , 0.0,
!
/

```

Figure 6. Example for the CTRL_TS namelist of IMPORT_TS.