Geoscientific
Model Development

Model description paper

The MESSy DWARF (based on MESSy v2.55.2)

Astrid Kerkweg[1,2], **Timo Kirfel**[1,2], **Duong H. Do**[1], **Sabine Griessbach**[2,3], **Patrick Jöckel**[4], **and Domenico Taraborrelli**[1,2]

[1]Institute of Climate and Energy Systems – Troposphere (ICE-3), Forschungszentrum Jülich, Jülich, Germany
[2]Centre for Advanced Simulation and Analytics (CASA), Forschungszentrum Jülich, Jülich, Germany
[3]Jülich Supercomputing Centre, Forschungszentrum Jülich, Jülich, Germany
[4]Deutsches Zentrum für Luft- und Raumfahrt (DLR), Institut für Physik der Atmosphäre, Oberpfaffenhofen, Germany

Correspondence: Astrid Kerkweg (a.kerkweg@fz-juelich.de)

Abstract. The adaptation of Earth system model (ESM) codes to modern computing architectures is challenging as ESMs consist of a multitude of different components. Historically grown and developed by scientists rather than software engineers, the codes of the individual components are often interwoven, making the optimisation of the ESMs rather challenging, if not impossible. Thus, in the last few years, the codes have become increasingly modularised, and, with that, different components are now disentangled from each other. This helps porting the code section by section to modern computing architectures, e.g. to GPUs.

For more than 20 years, the modularisation has been the fundamental concept of the Modular Earth Submodel System (MESSy). It is an integrated framework providing data structures and methods to build comprehensive ESMs from individual components. Each component, e.g. a cloud microphysical scheme, dry deposition of tracer gases, or diagnostic tools, as output along satellite orbits, is coded as an individual so-called submodel. Each submodel is connected via the MESSy infrastructure with all other components, together forming a comprehensive model system. MESSy was mainly developed for research in atmospheric chemistry, and, so far, it has always been connected to a dynamical (climate or weather forecast) model, i.e. what we call a base model. The base model is a development outside the MESSy framework. Running a full dynamical model for technical tests when porting only one submodel is a tedious task and unnecessarily resource-consuming. Therefore, we developed the so-called MESSy DWARF, a simplified base model based on the MESSy infrastructure. We implemented the definition of a very simple grid, a parallelisation scheme, and a time control to replace a fully fledged base model.

The MESSy DWARF serves as a valuable tool for technical applications, such as porting individual component implementations to GPUs and performance tests or as an easy test environment for process implementations. Due to the MESSy structure, the applied components can be used in a dynamical model without any changes because the interface is exactly the same. Furthermore, the MESSy DWARF is suited for scientific purposes running simplified models (with only a selection of components), e.g. a chemical box model for the analysis of chamber experiments or a trajectory box model imitating an air parcel rising slowly into the stratosphere. Column and plume models could also easily be built based on the DWARF.

In this article, we introduce the technical setup of the MESSy DWARF and show four example applications: (1) a simple application using a component calculating orbital parameters, (2) a chemical kinetics model including photolysis frequency calculation, (3) an application of a chemical box model, and (4) some details on a GPU performance test of the chemical kinetics model.

1 Introduction

This article describes the design concept of the MESSy DWARF. The DWARF establishes a standard tool to easily create simplified setups for technical tests. This is of major concern for the process of porting such a large code base as MESSy stepwise to GPU and for performance analysis of individual submodels. Moreover, the possibility of easily developing simplified MESSy models in a unified way, e.g. box or column models, to focus on local processes without the

need to consider horizontal transport was still missing within the MESSy framework.

The basic concept of the MESSy DWARF is to use MESSy data types and methods (for further details, see Sects. 2 and 3) to provide the technical framework required to run one specific process implementation (a submodel in MESSy terminology) to be able to concentrate in an efficient way on the optimisation of this specific process implementation or to "build" simplified models by combining a number of process implementations. To reach this goal, a model grid, the (distributed-memory) parallel domain decomposition, and a time control, which are usually provided by the dynamical models, need to be provided by MESSy. Before the DWARF development, MESSy already provided all data types and methods required to

- initialise the model variables by reading input data and transforming (remapping) it to the actual model grid;

- produce output of not only instantaneous values but also of namelist-driven standard diagnostic output as minima, maxima, averages, etc.;

- control the time-stepping and checkpointing;

- perform a tendency analysis for prognostic variables.

But for real applications, MESSy was always connected to a dynamical model. The idea to fully utilise the capacity of the MESSy framework to build simplified models, especially for testing and also for simpler scientific applications, was triggered by the publication of Müller et al. (2019). Earth system models (ESMs), especially those including explicit atmospheric chemistry process descriptions, are very expensive in terms of required computing time and memory. As these models have been developed over a long time, they are not straightforwardly efficiently applicable or even portable to new computing architectures. This problem is faced by most models of this kind around the world. Therefore, a strategy on how "to evolve weather and climate prediction models to next-generation computing technologies" (Müller et al., 2019) was developed within the ESCAPE (Energy-efficient Scalable Algorithms for Weather Prediction at Exascale) funded by HORIZON2020 (https://research-and-innovation.ec.europa.eu/funding/funding-opportunities/funding-programmes-and-open-calls/horizon-2020, last access: 11 February 2025). The first point of this strategy is to "(i) identify domain-specific key algorithmic motifs in weather prediction and climate models (which we term Weather & Climate Dwarfs)" (Müller et al., 2019, "Introduction" section). "The dwarfs thus represent domain-specific mini-applications (Messer et al., 2018) which include direct input from the domain scientist together with documentation" (Müller et al., 2019, Sect. 2.1). With the publication of Müller et al. (2019), the idea of restructuring comprehensive, established source codes to separate them into smaller pieces related to specifically simulated processes, or, in short, the

idea of formally modularising the source codes, became widely acknowledged in the community after already having been proposed in 2005 (Jöckel et al., 2005).

Combining the basic design concepts of the Modular Earth Submodel System (MESSy, Jöckel et al., 2005), i.e. that each component is already coded as an individual submodel, with the "dwarf strategy" developed and published by the ESCAPE project (Müller et al., 2019), resulted in the development of the MESSy DWARF.

While the ESCAPE Weather & Climate Dwarfs are seven specific dwarfs for seven process implementations, the MESSy DWARF adds a new base model to the MESSy framework, enabling running each MESSy submodel (currently, more than 90 are available, and the number is growing) in simplified setups. Each individual ESCAPE dwarf solves the problems by setting up its own minimal driver (grid, parallelisation etc.), while the MESSy DWARF is a generic base model with which all process descriptions, available as MESSy submodels, can be applied. The ESCAPE dwarfs have the advantage that the overhead is minimal, thus providing the perfect setup to optimise this specific process.

In contrast, the MESSy DWARF requires some additional overhead to the individually tackled submodel always using the above listed MESSy functionalities. While the larger overhead could be considered a downside for performance optimisation, it is an upside that the MESSy DWARF establishes a standard model, which can be used to drive all submodels, avoiding the need to develop individual drivers for each specific submodel. Furthermore, since the submodel interfaces are the same for all base models according to the MESSy concept, a submodel developed or optimised in a DWARF configuration can be directly used in a fully fledged base model without any changes. Additionally, it is possible to run and test combinations of submodels with the DWARF. This is not only helpful for technical optimisations (see Sect. 4.4), but also to open up the DWARF for scientific use, e.g. for simplified models such as chemical box models (see Sect. 4.3).

Starting with a short introduction into the basic MESSy concept (Sect. 2), the following section provides an overview of the technical realisation of the MESSy DWARF (Sect. 3). Section 4 contains several examples of DWARF applications to illustrate its usability. Finally, the article is concluded with a short summary and a list of some future development plans (Sect. 5).

2 The MESSy infrastructure and submodel concept

The Modular Earth Submodel System (MESSy, Jöckel et al., 2005) is an integrated software framework which was initially built to facilitate the implementation of atmospheric chemistry processes into existing models of atmospheric dynamics (Jöckel et al., 2010; Kerkweg and Jöckel, 2012). Its

basic idea is that each process and diagnostics can be coded as an individual submodel. Such a submodel consists of two parts: the submodel core layer (SMCL) contains the implementation of the process coded in its basic entity (e.g. kinetics is represented by a box and convection by a column process), while the submodel's interface layer (SMIL) connects to the MESSy infrastructure components on the (base-model-dependent) base-model interface layer (BMIL), which in turn is connected to the base-model layer (BML), i.e. the base model (see Fig. 1). This four-layer software architecture allows for the implementation of the SMCL to be completely independent of the base model, for instance, its local variable dimensions, order of ranks, and parallel decomposition.

MESSy distinguishes between regular submodels and infrastructure submodels. While the regular submodels are realisations of individual processes (e.g. the kinetic solver for the gas-phase chemistry, dry deposition and convection) or diagnostics, the MESSy infrastructure submodels establish the overall machinery required to orchestrate the individual submodels (data exchange, control flow, checkpointing, data import and export, etc.). Thus, the infrastructure submodels establish the framework to execute the regular submodels. The MESSy infrastructure submodels comprise – among others – submodels for

- memory management and output control (submodel CHANNEL, Appendix A[1], Jöckel et al., 2010);

- import of data (submodel IMPORT, Kerkweg and Jöckel, 2015);

- time and event control (submodel TIMER, Jöckel et al., 2010);

- data exchange between the base model and MESSy submodels (submodel DATA, Jöckel et al., 2005);

- flow control, i.e. activation of submodels (submodel SWITCH), and calling of the submodels during simulation (CONTROL, Jöckel et al., 2005);

- tracer data and tracer metadata handling (submodel TRACER, Jöckel et al., 2008);

- tendency diagnostic (submodel TENDENCY, Eichinger and Jöckel, 2014).

Table 1 lists the MESSy submodels mentioned in this publication, including a very short description and respective references.

If MESSy is connected to a dynamical model, the dynamical model defines the grid, provides the parallel domain decomposition information, including methods for data transpositions (e.g. based on the Message Passing Interface, MPI)

---

[1]Appendix A provides a very short introduction to the CHANNEL submodel. For readers not familiar with MESSy, it introduces the most important terms used in the remainder of this article.

and the set of prognostic variables and external data, such as the land–sea mask and surface elevation. This information is picked up by the MESSy infrastructure submodels and translated into MESSy data types for harmonised usage by all MESSy submodels.

3 The technical realisation of the MESSy DWARF

So far, MESSy has always been connected to dynamical models, e.g. the global climate model ECHAM5 (Roeckner et al., 2006; Jöckel et al., 2010) or the regional weather and climate model COSMO (Rockel et al., 2008; Kerkweg and Jöckel, 2012). However, for simplified scientific applications or for technical tests (such as source code optimisation), a dynamical model is not always required or even counterproductive due to unnecessary overhead, e.g. for performance analysis of a single MESSy submodel.

As for the MESSy DWARF, the base model is basically "empty"; all functionalities and data usually provided by the base model and made accessible via the MESSy infrastructure to the submodels need to be replaced by either additional infrastructure functionalities (e.g. the definition of a grid) or import of external data. The chemical kinetics submodel MECCA (Model Efficiently Calculating the Chemistry of the Atmosphere, Sander et al., 2019) is used as an example to demonstrate the dependence of a process submodel on entities usually provided by the base model and to illustrate the concept of the DWARF.

To meaningfully set up the DWARF, with MECCA being the only regular MESSy submodel used, three main aspects need to be considered:

1. *The provision of input data required by MECCA*. These are the following:

    i. *The temperature and the specific humidity*. Temperature and specific humidity are prognostic variables. Dynamical base models provide the set of prognostic variables as determined by their dynamical core. In MESSy, these are made accessible to the MESSy submodels via the infrastructure submodel TENDENCY (see Sect. 3.1.1). As the DWARF does not contain a dynamical core, the regular MESSy submodel DWARFDCD (DWARF's Dynamical Core Dummy) simply provides a set of prognostic variables (see Sect. 3.1.1).

    ii. *The photolysis frequencies*. The photolysis frequencies are usually calculated by the MESSy submodel JVAL (Sander et al., 2014). Given a setup where MECCA is the only used regular submodel, J values need to be provided as external data. External data are imported via the MESSy infrastructure submodel IMPORT (see Sect. 3.1.2).

    iii. *The pressure at box mid-points*. The pressure is provided by the base model itself and referenced by

Table 1. List of MESSy submodels including a short description and citation (if possible).

| Submodel name	Description	Citation
	Infrastructure submodels	
CHANNEL	CHANNEL handles the memory and metadata management and data export.	Appendix A, Jöckel et al. (2010)
CONTROL	CONTROL controls the workflow of a MESSy simulation. In each entry point, it calls the individual submodels (interlinked with SWITCH).	Jöckel et al. (2005)
DATA	DATA steers the data exchange between the base model and MESSy submodels in both directions.	Jöckel et al. (2005)
DECOMP	DECOMP provides the parallel domain decomposition of the model grid.	This article
GRID	GRID provides the definition of the model grid and of grid-related variables, such as longitude, latitude, box height, and grid volume.	Kerkweg et al. (2018), this article
IMPORT	IMPORT performs the data import into the model. It contains three sub-submodels which themselves deal with different types of data.	Kerkweg and Jöckel (2015)
– IMPORT_GRID	IMPORT_GRID imports time series of gridded data in netCDF format to the actual model grid.	
– IMPORT_LT	IMPORT_LT performs the import of lookup tables. In MESSy they are used, for example, to import optical properties required for the submodel AEROPT (Dietmüller et al., 2016), which calculates the aerosol optical properties.	
– IMPORT_TS	IMPORT_TS imports standard time series data.	
MPI	MPI establishes parallel communication (e.g. broadcasts and data gathering and scattering) via the Message Passing Interface (MPI).	
SWITCH	SWITCH activates individual submodels.	Jöckel et al. (2005)
TENDENCY	TENDENCY provides interfaces for the access and modification of prognostic variables. Furthermore, it enables the namelist-driven possibility for tendency diagnostics for individual prognostic variables for single processes or collections of processes.	Eichinger and Jöckel (2014)
TIMER	TIMER establishes the time settings of the simulation and provides an event control; i.e. among others it sets the date components, schedules restarts, and provides the interfaces for scheduling events in other submodels.	Jöckel et al. (2010)
TRACER	TRACER manages tracers and their metadata. This includes, among others, the definition of tracers; their attribution to specific processes (e.g. if a tracer should be advected); and the assignment of chemical and physical properties as molar mass, Henry coefficients, etc.	Jöckel et al. (2008)
	Regular submodels	
DWARFDCD	DWARF-specific submodel which defines a set of prognostic variables for DWARF and enables initialisation and nudging of the prognostic variables defined in this submodel.	This article
JVAL	JVAL performs the calculation of photolysis frequencies.	Sander et al. (2014)
MECCA	MECCA is a kinetic solver for gas- and liquid-phase reactions.	Sander et al. (2019)
ORBIT	ORBIT provides the calculation of orbital parameters.	Dietmüller et al. (2016)
PTRACINI	It stands for Passive TRACer INItialisation.	Hofmann et al. (2016), Appendix C in this article
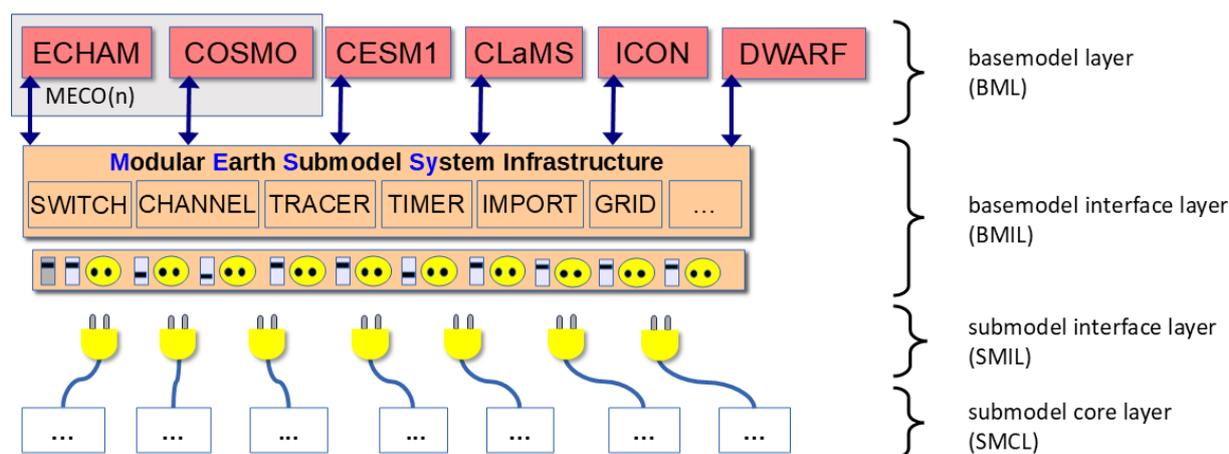
Figure 1. Sketch of the four-layer code structure of MESSy. The image illustrates how the different base models and the base-model-independent core layer of the individual submodels are interconnected via the two interface layers. The base-model interface layer (BMIL) organises the correct workflow and data exchange between the regular submodels, the cores of which are connected by the submodel interface layer (SMIL) to the BMIL.

the MESSy infrastructure submodel DATA, or, if the base model does not provide a pressure field, it is calculated in DATA. Thus, in the case of the MESSy DWARF as well, it is calculated or referenced by DATA (see Sect. 3.1.3). In the case of a reference, it will relate to external data provided by IMPORT_GRID (see Sect. 3.1.2).

iv. *The tracers (chemical active trace species).* The tracers are created by MESSy submodels, i.e. in the example, by MECCA. The tracer metadata and memory management is provided by MESSy infrastructure submodel TRACER and the data are accessed and changed via the infrastructure submodel TENDENCY (same as the other prognostic variables). Thus, tracers are handled in exactly the same way for the DWARF as for legacy base models.

v. *The longitude and latitude fields for debug output.* Last but not least, for debug output, the longitude and latitude fields are required by MECCA. These are always provided by the MESSy infrastructure submodel GRID_DEF (see in Sect. 3.2) independently of the base model.

2. *The handling of the output data provided by MECCA.* MECCA solves the kinetic equations of the gas-phase chemistry and calculates the change in reactive chemical species over a time increment; thus, the output of MECCA consists of the tendencies of the reactive trace gases. These tendencies, i.e. the rate of change for the current time step inferred by kinetics, are fed back via the TENDENCY infrastructure model.

3. *The grid on which the simulation should be performed.* A full dynamical model naturally provides a grid on

which the primitive equations and the model physics are solved. For the DWARF, a simple grid is defined by the infrastructure submodel GRID_DEF (see in Sect. 3.2). Additionally, the parallel domain decomposition of the grid and the parallel communication patterns need to be established (Sect. 3.3) if working in (distributed-memory) parallel environments.

The following subsections provide more details on how the features listed above are implemented in the MESSy DWARF.

3.1 Data supply

In the usual MESSy model setup using a legacy model as the base model, the MESSy submodels have access to two types of external data:

– data provided by the base model and

– data imported by the MESSy submodel IMPORT.

The base-model data themselves separate into (i) initial and boundary data (including nudging data) and (ii) base model variables.

i. Initial data are used to initialise the model and therefore are only required at the very first model time step. In contrast to this, boundary data need to be updated regularly, for example, after discrete time intervals during the model simulation.

ii. Base-model variables are calculated by the base model. The most important variables are the prognostic variables, which are the variables used to solve the primitive equations. The set of prognostic variables can therefore differ between base models. Non-prognostic variables are often called diagnostic variables.

```
!------------------------------------------------------------------------------
! -*- f90 -*-
&CTRL
nudgedt_t  = 3600.0,
/


&CPL
inp_tm1    = 'import_grid', 'inp3d_tm1',
inp_xlm1   = 'import_grid', 'inp3d_xlm1',
inp_xim1   = 'import_grid', 'inp3d_xim1',
! provide rhum instead of q
inp_rhum   = 'import_grid', 'inp3d_rhum',
inp_press  = 'import_grid', 'inp3d_press' ! required if inp_rhum is set
/

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

Figure 2. Example namelist file, `dwarfdcd.nml`, for the MESSy submodel DWARFDCD. According to the &CPL namelist, all prognostic variables provided by DWARFDCD except q are initialised from external data read by IMPORT_GRID. q is initialised from an imported relative humidity field. For this calculation, additionally, the pressure field is required, which is also provided by IMPORT_GRID. The only time-varying, i.e. nudged, prognostic variable is the temperature, as a nudging coefficient is defined only for this in the &CTRL namelist.

If MESSy is run in connection with a legacy base model, the base-model-specific variables are made available to the MESSy submodels by the translation of those (in the BMIL) into so-called MESSy channel objects. The technical implementation of this translation in the BMIL depends on the base model. In EMAC, for instance, the translation is based on pointer association[2] of channel objects to the internal data structures of the ECHAM5 base model. In COSMO/MESSy, the COSMO source code has been modified to allocate memory of the COSMO variables directly in the form of channel objects via the corresponding CHANNEL methods. Prognostic variables, specifically, are translated (i.e. made accessible) via the MESSy submodel TENDENCY (Eichinger and Jöckel, 2014). Diagnostic variables required by several MESSy submodels but not present in the respective base model are calculated in the (base-model-specific part of the) MESSy submodel DATA. One example might be the 3-dimensional geopotential height field, which is available in ECHAM5 but not in COSMO and therefore is declared and calculated in the COSMO-specific part of DATA.

The DWARF, however, is a generic (basically empty) base model without the functionality of a legacy base model. Thus, first, a set of prognostic variables needs to be defined. This is implemented within the submodel DWARFDCD (see Sect. 3.1.1). Second, all other variables in the DWARF become initial or boundary data and are imported by the MESSy submodel IMPORT.

Which set of prognostic variables is defined by the new MESSy submodel DWARFDCD and how they are modified is described next (Sect. 3.1.1). The functionality of the IM-

PORT infrastructure submodel is explained in Sect. 3.1.2. The last subsection (Sect. 3.1.3) describes which variables are calculated additionally by the MESSy submodel DATA.

3.1.1 Definition and modification of prognostic variable

The MESSy infrastructure submodel TENDENCY (Eichinger and Jöckel, 2014) allows for detailed analyses of individual tendencies[3] of prognostic variables. Each process using a prognostic variable as input, or changing it, must access and/or modify it via methods (i.e. calling of subroutines) of the infrastructure submodel TENDENCY. In this way, it can be ensured that the operator-splitting concept is always followed (meaning implemented in a numerically correct way) and that all tendencies can be budgeted by TENDENCY if desired.

This implies that the MESSy submodels are expecting specific prognostic variables to be accessible and modifiable via TENDENCY. Therefore, for the MESSy DWARF, the new submodel DWARFDCD defines a set of the prognostic variables. Currently, the following prognostic variables are provided by DWARFDCD:

- the air temperature, t, in kelvin;

- the horizontal wind velocities, u and v, in $\mathrm{m\,s^{-1}}$;

- the water vapour, q, in $\mathrm{kg\,kg^{-1}}$;

---

[2]This definition is provided by the Fortran language standard.

[3]Here, tendency denotes the time increment of a variable X: $X_{\mathrm{new}} = X_{\mathrm{old}} + \Delta X \times \mathrm{d}t$, with X_{new} and X_{old} indicating two time levels of the variable, $\mathrm{d}t$ being the time step length, and ΔX being the tendency of the prognostic variable.

– the liquid water content, x_l, in $kg\,kg^{-1}$;

– the ice water content, x_i, in $kg\,kg^{-1}$.

Due to the setup of the DWARF (i.e. without a dynamical core), the prognostic variables are only changed if a MESSy submodel adds a tendency to the respective variable. However, it might be desirable to initialise and change the content of the prognostic variables. Consequently, DWARFDCD provides an initialisation and a nudging option. Using the `&CTRL` namelist of the DWARFDCD namelist file (see Fig. 2), a relaxation time can be specified for each of the prognostic variables independently. The variable `nudgedt_Y`, with $Y = t, u, v, q, x_l, x_i$, respectively, provides this relaxation time (in seconds) with which the respective variable is forced to follow the imported data that are accessed via the corresponding `&CPL` namelist entries (`inp_Y`). The tendency (`tend_Y`) added to the prognostic variable Y is then calculated as follows:

$$\texttt{tend\_Y} = (\texttt{inp\_Y} - \texttt{start\_Y})/\texttt{nudgedt\_Y}, \quad (1)$$

with `start_Y` being the current value of variable Y.

If only the initialisation, without nudging, of a prognostic variable is required, the setting of the `inp_` parameters in the `&CPL` namelist is sufficient. A nudging coefficient must not be defined as the definition of the nudging coefficients triggers the nudging.

Note, a prognostic variable stays zero if no input data are provided and no process adds a tendency to it. This is technically possible as not all simplified DWARF setups require all prognostic variables.

As the DWARF will be used for measurement campaign analyses, a special case has been introduced for the initialisation and nudging of the water vapour mixing ratio. This is rendered useful as measurements often provide only the relative humidity. Therefore, the option was implemented to use the relative humidity (in percentages) as input for the forcing of the water vapour, q (in $kg\,kg^{-1}$), instead of prescribing the water vapour mixing ratio directly. This option is activated by providing `inp_rhum` instead of `inp_q`. Additionally, the conversion of relative humidity to specific humidity requires the pressure as input; thus, `inp_press` needs to be provided in the `&CPL` namelist.

3.1.2 Initial and boundary data

One important part of each model simulation is the provision of initial and boundary data. The MESSy infrastructure submodel IMPORT provides the standard interface for data import at the beginning of and during the model simulation. See the IMPORT user manual (available in the supplement of Kerkweg and Jöckel, 2015, or in the MESSy code distribution) for technical details and the usage of the IMPORT submodel, including a detailed description of the meaning of namelist parameters. Two sub-submodels of IMPORT (com-

pare Table 1 and Kerkweg and Jöckel, 2015) are used for replacing base-model functionalities:

– *IMPORT_GRID*. The MESSy infrastructure submodel IMPORT_GRID imports 2- or 3-dimensional gridded data at the beginning of and during a simulation. Which data are imported and when is determined by namelists. The import process includes (i) reading in the requested data, (ii) distributing it according to the parallel domain decomposition, and (iii) remapping it onto the actual model grid used. The order of these steps depends on the base model used. For the DWARF and its limited-area grid (see Sect. 3.2), each parallel MPI task reads and interpolates the data for its part of the grid. The DWARF uses the SCRIP (A Spherical Coordinate Remapping and Interpolation Package) algorithm in IMPORT_GRID (Jones, 1999) for the remapping.

Gridded data can be read once at the beginning of the simulation or in discrete time intervals during the simulation if a time series of gridded data is available. The triggers for reading and the sequences of time steps to be selected from the files are defined by a Fortran namelist as the user interface.

– *IMPORT_TS*. IMPORT_TS provides an interface for reading standardised time series data, i.e. data available for a specific period in time and with a specific number of parameters, from files in ASCII or netCDF format.

In full 3-dimensional simulations, this is, for example, used to read the vertical equatorial zonal wind profile for quasi-biannual oscillation (QBO) nudging (Jöckel et al., 2016). In the case of the DWARF, IMPORT_TS is especially helpful for box-model simulations, i.e. for reading in measurement data to constrain the DWARF to the experiment conditions. Section 4.3 provides an example for this.

3.1.3 DATA provision

The MESSy infrastructure submodel DATA serves as a translator, providing the base-model data to the MESSy submodels in a harmonised way. DATA performs different operations to simplify the access to the base-model data for the submodels:

1. Variables available from different base models describing the same content might be named differently. In this case, DATA defines a channel object reference (see Appendix A) to harmonise the channel and object name with which the individual submodels access this variable.

2. Variables which are not directly available from the base model but are derivable from other variables are calculated by DATA. For these, a channel object is defined (i.e. the respective memory is allocated), and the variable is calculated during the time loop.

```
!----------------------------------------
! -*- f90 -*-

&CPL
inp_press  = 'import_grid', 'inp3d_press'
inp_pressi = 'import_grid', 'inp3d_pressi'
/

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

Figure 3. Example namelist file, `data.nml`, for the MESSy submodel DATA.

The first category is not available in the MESSy DWARF as, in the DWARF setup, there is no data-providing base model; i.e. the DWARF is completely driven by imported data. The second category is available for the DWARF.

Variables, which are usually provided by the dynamical model, are required by some process submodels. If these can be calculated from the basic model setup and the prognostic variables, they are calculated within DATA for the DWARF. These variables are

- the Coriolis parameter (s^{-1}),

- the vorticity (s^{-1}),

- the density of dry air $(kg\,m^{-3})$.

Even more importantly, the 3-dimensional pressure fields defined at the box centres or the vertical interfaces are required by many calculations in the MESSy submodels. These can either be prescribed by imported data (in this case, the imported data need to be coupled to DATA via the `&CPL` namelist of DATA; see Fig. 3) or are allocated and calculated in DATA itself following the US Standard Atmosphere, 1976 (U.S. Government Printing Office, 1976), using the height fields as determined in GRID (see Sect. 3.2).

3.2 GRID_DEFinition

For the MESSy DWARF the grid is defined in the MESSy infrastructure submodel GRID_DEF. For the start, we decided to use a very simple grid as, for technical tests, only the number of grid boxes or the overall size of the grid matters. The basic grid layout is similar to the one used in the regional weather prediction and climate model COSMO (Doms and Baldauf, 2021; Rockel et al., 2008) but much more simplified as the grid defined for the DWARF in GRID_DEF does not allow for any rotation of the geographical grid and does not include halo longitudes and latitudes.

The horizontal grid is defined as a rectangle spanning a limited area (Fig. 4a), which is defined by the lower-left corner (the geographical coordinates of the grid-box centre point of the lower-left grid corner have to be provided in the GRID_DEF namelist as `startlon_tot`, in $°E$, and

`startlat_tot`, in $°N$), the number of grid boxes in longitude and latitude directions (`mgprow` and `mgpcol`), and the grid spacing (`dlon` / `dlat` in $°E/°N$, respectively). Figure 5 displays an example of the namelist file of the GRID submodel `grid.nml`. The above listed namelist parameters are part of the `&CTRL_GRID_DEF` namelist.

The vertical grid (Fig. 4b) is dimensioned by the number of vertical boxes (nlev). Their vertical extent is defined via the namelist parameter `vc_heighti` (formula symbol vc), which provides the heights in metres above topography (H_{topo} in metres) of the interface levels of the vertical grid boxes. The 3-dimensional fields containing the altitude above ground ($H^{I_{gnd}}$) and the altitude above mean sea level ($H^{I_{msl}}$) at the vertical interface (I) of the grid are calculated from this input as follows:

$$H^{I_{gnd}}(i, j, k) = vc(k),$$
$$H^{I_{msl}}(i, j, k) = vc(k) + H_{topo}(i, j), \tag{2}$$

with i, j, and k being the longitudinal, latitudinal, and vertical index, respectively. The height at the box mid-levels is calculated by

$$H_{gnd}(i, j, k) = 0.5 \times (H^I_{gnd}(i, j, k+1) + H^I_{gnd}(i, j, k)),$$
$$H_{msl}(i, j, k) = 0.5 \times (H^I_{msl}(i, j, k+1) + H^I_{msl}(i, j, k)). \tag{3}$$

The second namelist of the GRID namelist file (Fig. 5), i.e. `&CPL_GRID_DEF`, defines the coupling, i.e. the input required from other MESSy submodels. In the example case, the topography is read by the MESSy submodel IMPORT_GRID, and the coupled object is named `inp_topoheight`. Additionally, the calculation of the air mass in the grid box and the dry air mass in the grid box require the pressure. In Fig. 5, the pressure is not pre-described but is calculated by the MESSy submodel DATA according to the US Standard Atmosphere (1976).

Last but not least, for a very reduced setup, when even the water vapour content is not known, the calculation of the grid mass can be skipped by setting the logical switch `lignore_mass` to true.

3.3 DECOMPosition and parallel communication (MPI)

The MESSy infrastructure submodel DECOMP organises the parallel decomposition of the MESSy DWARF. The parallelisation is a classical horizontal domain decomposition along the longitudes and latitudes. The example namelist in Fig. 6 shows a decomposition along the longitudinal range into four segments and along the latitudinal range into two segments, requiring, overall, $4 \times 2 = 8$ MPI tasks. The number of grid boxes is divided by the number of segments in each direction. If it cannot be divided without a remainder, the number of grid boxes is increased by 1 for as many tasks as correspond to the remainder. In our example, the latitude
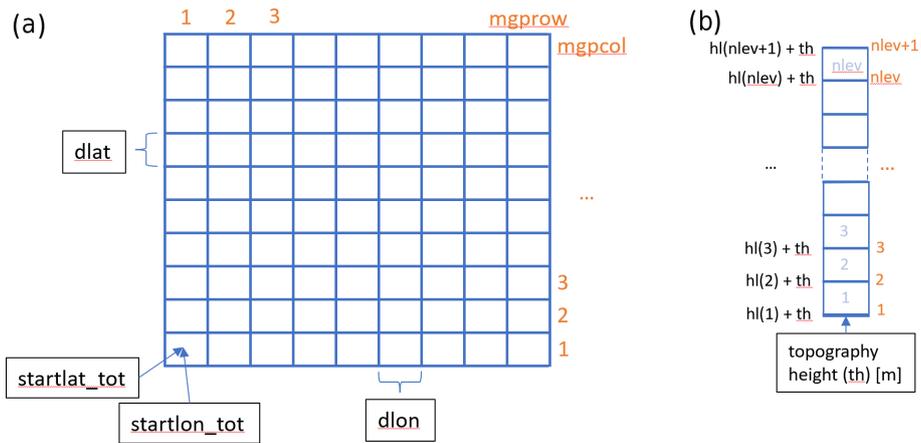
Figure 4. Illustration of the horizontal **(a)** and vertical **(b)** grid definition.

```
!-----------------------------------------------------------------------------
! -*- f90 -*-
&CTRL_GRID_DEF
!
mgpcol = 93
mgprow = 383
nlev   = 40
!
startlon_tot = 15.0
startlat_tot = 30.0
!
dlon = 0.5
dlat = 0.5
!
! height over ground for 40 levels:
vc_heighti = "22700.0, 20800.0, 19100.0, 17550.0, 16150.0, 14900.0, 13800.0, 12785.0,
              11875.0, 11020.0, 10205.0,  9440.0,  8710.0,  8015.0,  7355.0,  6725.0,
               6130.0,  5565.0,  5035.0,  4530.0,  4060.0,  3615.0,  3200.0,  2815.0,
               2455.0,  2125.0,  1820.0,  1545.0,  1295.0,  1070.0,   870.0,   695.0,
                542.0,   412.0,   303.0,   214.0,   143.0,    89.0,    49.0,    20.0,
                  0.0"
/
!-----------------------------------------------------------------------------
&CPL_GRID_DEF
inp_topoheight = 'import_grid', 'inp2d_topoheight',
inp_press_3d   = 'dwarf', 'press',
!lignore_mass = T,
/
!-----------------------------------------------------------------------------
```

Figure 5. Example namelist file grid.nml for the MESSy sub-submodel GRID_DEF.

range covers 93 grid boxes, which are distributed among two segments. In this case, the tasks of the first segment get 46 grid boxes in the latitudinal direction, while the tasks in the second segment get 47 grid boxes. The same happens for the longitudinal range: $393/4 = 98.25$. Thus, the tasks in the first three segments get 98 grid boxes, while the tasks in the fourth segment get 99 grid boxes. Thus, task 0 gets 98×46 grid boxes, while task 7 gets 99×47 grid boxes.

Internally, DECOMP saves the start indices of the respective local domains and determines the start longitude and latitude (startlon/startlat) of each local domain.

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

! -*- f90 -*-

&CTRL

npx = 4

npy = 2

/

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

Figure 6. Example namelist file, `decomp.nml`, of the MESSy submodel DECOMP.

The parallel communication is established by the MESSy submodel MPI. It uses the Message Passing Interface (MPI) to perform the communication between the different parallel tasks. The individual broadcast, gather, and scatter routines are based on the respective implementation of the COSMO model.

3.4 Limitations of the DWARF approach

As the DWARF provides a very simplistic replacement for a fully fledged dynamical model, there are limitations to the applicability of the DWARF:

– The MESSy DWARF in its basic configuration (i.e. if only the infrastructure submodels and DWARFDCD are active) does not provide any dynamical, physical, or chemical processes; thus, if no regular MESSy submodel is switched on to provide any tendencies, prognostic variables (including tracers) will be constant in time. If they are not initialised, the prognostic variables (including tracers) will be zero. Furthermore, as long as no regular MESSy submodel contributes a transport process, the DWARF just consists of a number of independent (unconnected) boxes, which are organised on a 3-dimensional geographical grid.

– The regular MESSy submodel DWARFDCD is required to create the prognostic variables, which are accessed by the regular MESSy submodels via TENDENCY. Each of the prognostic variables needs to be initialised if it should have a meaningful value. Nudging needs to be switched on if a prognostic variable should change over time. All of these are namelist settings, and it is the obligation of the user to ensure a meaningful setup. Otherwise, prognostic variables that are zero could lead to errors.

– The currently applied simple grid is defined without halo cells; i.e. calculations, which require data of the horizontally neighbouring cells, will fail at the border of the local grid cell. As a consequence, these fields show incorrect values at the edge cells of the local domain, and the results are dependent on the chosen parallel domain decomposition. However, almost all MESSy submodels operate only on one horizontal grid cell or within a vertical column of grid cells; thus, for engineering tests, this limitation can be accepted. If the knowledge of neighbouring cells is necessary for a scientific application, the MPI communication needs to be implemented to exchange the values of the halo cells.

– The user has to take care that all the data required as input to a submodel are available from either imported data or other MESSy submodels. Missing data, however, in contrast to the first two points, will lead to an error during runtime telling the user that data are missing.

4 Example applications

This article describes the design concept of the MESSy DWARF. The major driving force for developing the DWARF was, on the one hand, the need to establish a standard tool to easily create simplified setups for technical tests. This is of major concern for the process of porting such a large code base as MESSy stepwise to GPU and for performance analysis of individual submodels. On the other hand, the possibility of easily developing simplified MESSy models in a unified way, e.g. box or column models, to focus on local processes without the need to consider horizontal transport was still missing within the MESSy framework.

Table 2. Process submodels used in example simulations.

Acronym	Process submodels used
Orbit1/2	ORBIT
Kin	ORBIT, JVAL, MECCA
Box1	JVAL, MECCA
Box2	JVAL, MECCA, PTRACINI
GPU	ORBIT, JVAL, MECCA

In the following, four examples for the applicability of the MESSy DWARF are presented. At the beginning, a very simple model calculating orbital variations (Sect. 4.1) is shown. A simplified 3-dimensional chemical model (Sect. 4.2) and a chemical box model (Sect. 4.3) are presented next. The last example demonstrates how a simplified 3-dimensional chemical model can be used for performance analysis in the framework of porting the chemical kinetics model to GPU (Sect. 4.4). Table 2 lists the regular submodels used for the respective examples, while Table 3 provides an overview of the respective model configurations.

Table 3. Basic setup of example simulations.

Acronym	Lower-left corner	mgprow	mgpcol	nlev	dlon [° E]	dlat [° N]	Start time (YYYY-MM-DD-HH)	End time	Time step [s]	Output frequency [min]
Orbit1	70.0° S, 150.0° W	123	71	40	2.0	2.0	1998-03-19-23	1998-03-22-01	300	60
Orbit2	70.0° S, 150.0° W	123	71	40	2.0	2.0	1997-12-31-23	1998-01-02-01	300	60
Kin	30.0° N, 15.0° W	63	43	40	0.5	0.5	1998-06-01-00	1998-06-02-01	120	10
Box 1/2	50.9° N, 6.4° E	1	1	1	0.0001	0.0001	2021-01-10-10	2021-01-10-16	120	10
GPU	30.0° N, 15.0° W	63	43	40	0.5	0.5	1998-06-01-00	1998-06-01-01	120	10

4.1 Orbital variations

As the first, very simple example, the submodel ORBIT (Dietmüller et al., 2016) calculating orbital parameters is used. The only input ORBIT requires are the time and the geographical location. These are provided in the same way for all base models, including the MESSy DWARF, by the infrastructure submodel TIMER and GRID_DEF, respectively.

For this example, a simulation has been run for 20–21 March 1998 (Orbit1 in Tables 2 and 3) and 1 January 1998 (Orbit2). Figure 7 shows some of the results.

The diurnal cycle of the cosine of the solar zenith angle exhibits the largest amplitude at the Equator, and the amplitude decreases with the distance to the Equator (Fig. 7b, d). On the equinox on 20–21 March (Fig. 7b), the cosine of the solar zenith angle for 60 and 40° S equals exactly those at 60 and 40° N, respectively. However, this is only true for the equinox, while differences in the cosine of the solar zenith angle are already visible before and after the equinox. In addition, in January (Fig. 7d), the shift in the solar zenith angle according to the season for the regions nearer to the poles is apparent.

While the right panels give an impression of the modulation of the solar zenith angle from a local point of view, the left panels of Fig. 7 give a spatial impression. They present area plots of the cosine of the solar zenith angle in the model domain at 12:00 UTC. As expected, the maximum for 20 March is exactly at the Equator, while the maximum for January is shifted towards the South Pole.

4.2 The chemical kinetics model

The second example (Kin in Tables 2 and 3) is a simple 3-dimensional setup for atmospheric gas-phase chemical kinetics. The only submodels used are (see Table 2)

- the chemical kinetics submodel MECCA (Sander et al., 2005, 2019), solving the ordinary differential equation system for gas-phase chemistry, where the chemical mechanism is the same as the basic mechanism used by Jöckel et al. (2016);

- the submodel JVAL (Sander et al., 2014), calculating the photolysis frequencies required within MECCA;

- the submodel ORBIT (Dietmüller et al., 2016), providing the orbital parameters required by JVAL.

As no transport processes are included in this model setup, in fact, independent simulations in $\mathrm{mgprow} \times \mathrm{mgpcol} \times \mathrm{nlev} = 63 \times 43 \times 40 = 108\,360$ grid boxes (compare Table 3) are performed. The boxes differ in their horizontal and vertical position according to the defined grid (from 30 to 51.5° N and 15° W to 16.5° E). Differences in the simulated chemistry between the individual grid boxes are due to either the initialisation of the chemical species or the prescribed external data.

The inputs required by ORBIT and MECCA are discussed in Sect. 4.1 and the introduction of Sect. 3, respectively. The input fields required by JVAL are the following:

1. Firstly, the temperature, liquid water content, and ice water content are required. These are accessed via TENDENCY and are, in the MESSy DWARF, created by the submodel DWARFDCD. In DWARFDCD, they have been initialised by data imported with IMPORT_GRID. No nudging is applied. Thus, these three variables are constant over time.

2. Another required field is the ozone tracer. It is defined and modified by the submodel MECCA. Thus, it is provided in the same way in DWARF as in fully fledged base models.

3. The third requirement are the cosine of the zenith angle and the distance between the Sun and the Earth. These are provided by the MESSy submodel ORBIT.

4. Next, the ozone column above the model top and solar cycle data are required. These are made available from external data via IMPORT_GRID for all base models.

5. The pressure at box mid-points and vertical interfaces are another required input field. These are, in this DWARF setup, calculated in the infrastructure submodel DATA from the height grid defined in the infrastructure submodel GRID_DEF.
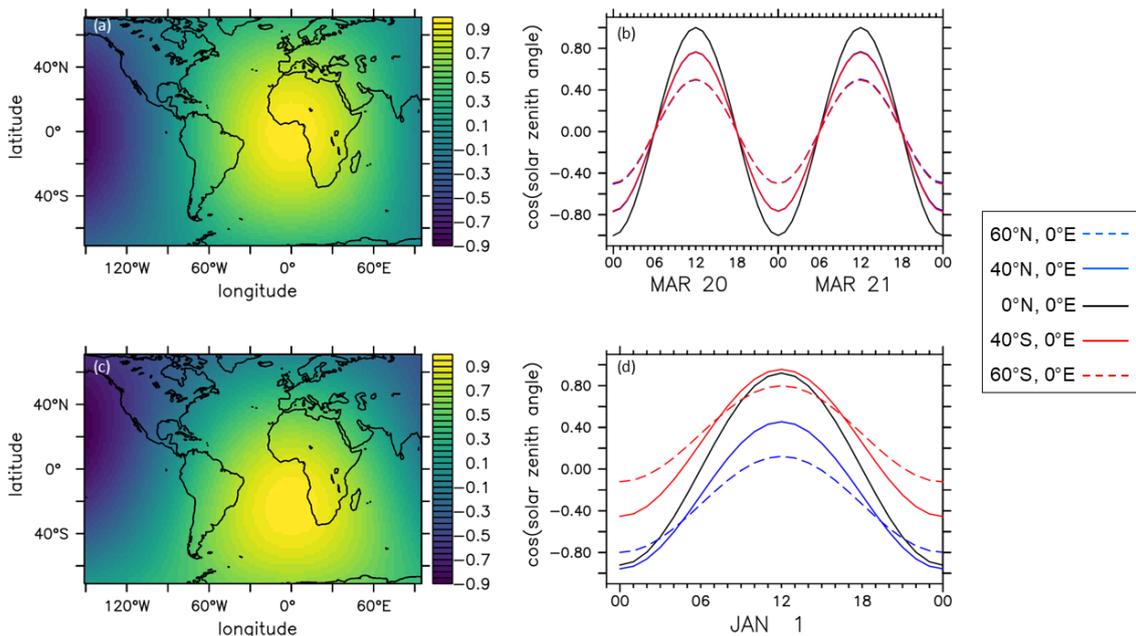
Figure 7. Cosine of the solar zenith angle calculated by the submodel ORBIT. Panels **(a)** and **(c)** show area plots over the entire model domain for 20 March 1998, 12:00 UTC **(a)**, and for 1 January 1998, 12:00 UTC **(c)**. Panels **(b)** and **(d)** show the diurnal cycles of the cosine of the solar zenith angle: 60° S, 40° S, 0° N, 40° N, and 60° N and 0° E for 20–21 March 1998 **(b)** and 1 January 1998 **(d)**, respectively.
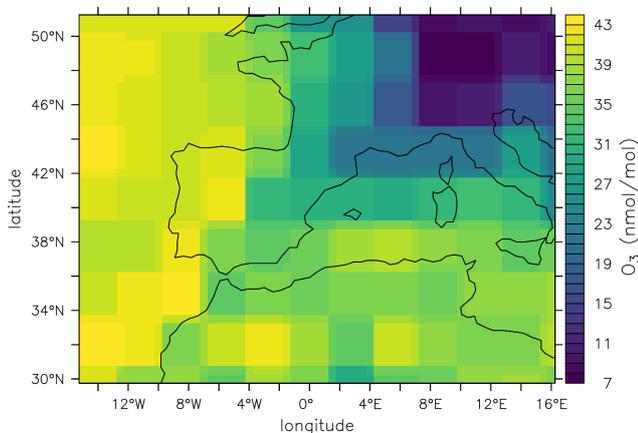

Figure 8. Ozone mixing ratio (nmol mol^{-1}) in the lowest model layer 10 min after model start (i.e. 1 January 1998, 00:10:00 UTC).

6. Finally, this includes relative humidity, land–sea fraction, cloud cover, and albedo. All four variables are usually provided by a dynamical base model; in this example, these are provided as external data, i.e. read by IMPORT_GRID.

For the initialisation of the tracers, a restart file from a former T42 EMAC simulation is used. Figure 8 displays the initial surface mixing ratio of ozone. The chessboard pattern results from the remapping of the coarse T42 (approx. $2.8° \times 2.8°$) resolution to the $0.5° \times 0.5°$ grid of the DWARF.

Table 4. Prescribed cosine of the solar zenith angle of the kinetic box-model study (Box1), the hours at which the sun is "switched on" and "switched off", and the respective line colour in Fig. 10.

Cosine zenith angle	Start time (UTC)	End time (UTC)	Line colour
0.9	11	14	black
0.7	11	14	red
0.5	11	14	blue
0.7	11	12	light blue

As the reaction rates in MECCA depend on temperature, pressure, and humidity, the initialisation of these fields influences the simulated kinetics directly. Note that, in this setup, these temperature, pressure, and humidity fields are only initialised. For the sake of simplicity, they are kept constant in time. Furthermore, the calculation of the photolysis frequencies (submodel JVAL) influences the chemistry and depends itself on solar activity, orbital parameters, pressure, ozone, cloud cover, relative humidity, albedo, and surface type (land or sea).

To illustrate the spatial variation in these parameters, Fig. 9 displays vertical profiles at three different locations: 49° N, 9° E (blue); 35° N, 9° E (red); and 35° N, −8° E (black). These locations have been chosen with regard to the surface ozone (compare Fig. 8), for low (blue), medium (red), and high (black) surface ozone, respectively.
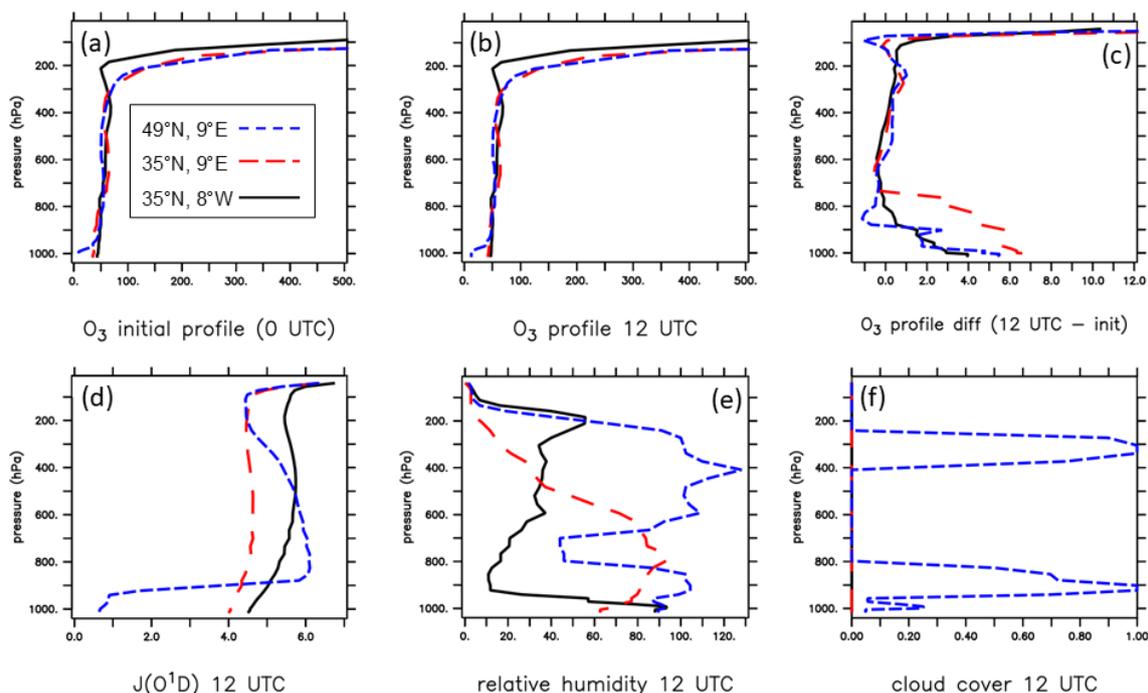
Figure 9. Ozone profiles at three geo-locations are presented: 49° N, 9° E (blue); 35° N, 9° E, (red); and 35° N, 8° W (black). The upper row displays ozone profiles (nmol mol^{-1}) at model start (1 January 1998, 00:00 UTC) **(a)** and at 12:00 UTC **(b)** and the difference between these profiles **(c)**. The lower row depicts profiles at 12:00 UTC of the photolysis frequency of O^{1}D (in 10^5 s^{-1}) **(d)**, the relative humidity (in percentages) **(e)**, and the cloud cover (as a fraction between 0 and 1) **(f)**.

The upper row of Fig. 9 depicts the ozone profiles shortly after model start (left, annotated as initial) and at 12:00 UTC (middle) and the difference between these profiles (right). Ozone is the lowest in the lowest model layer, while it is some orders of magnitude higher in the stratosphere (above approx. 250 hPa). However, the difference between the vertical profiles (right panel) between the initial condition and the 12:00 UTC vertical profile exhibits different time evolutions of the profiles in each box. The bottom row of Fig. 9 displays the respective vertical profiles of the photolysis frequency of O^{1}D (O$_3$ + $h\nu$ → O^{1}D + O$_2$, left), the relative humidity (mid), and the cloud cover (right). While the black and the red profiles are cloud-free, the blue profile exhibits two distinct cloud layers in which the 100 % relative humidity level is exceeded and supersaturation persists. Below the lower-level cloud, the photolysis frequency is much lower than for the two profiles without clouds. Above the cloud, the photolysis frequency is slightly enhanced (compared to the other two profiles), which is caused by the reflection of sunlight on top of the underlying cloud. While the red and the blue profiles are above land, the black profile is above the ocean. However, when comparing the red with the black profile, the red profile exhibits a much higher relative humidity, except for the lowest model layer. In the pressure range approximately between 980 and 500 hPa, the red profile contains much more water vapour. This is reflected in the photolysis frequencies, which are much lower for the red profile.

As atmospheric chemistry is a non-linear process, none of the shown aspects alone can explain the vertical profile of a specific chemical species as here, for example, ozone. However, using a chemical box model provides an efficient tool to test dependencies on individual parameters. How such a box model can be built with the DWARF and how it can be used is explained in the next subsection.

4.3 The chemical box model

In contrast to the model setup described in the subsection above, this subsection deals with a single box model, i.e. `mgprow`, `mgpcol`, and `nlev` are all 1, the bottom and the top height of this box are defined as `vc_heighti = 15.0, 5.0`, i.e. the centre of the box is 10 m above the surface. The geographical location of the box is Jülich, Germany (50.9° N, 6.4° E). In the first example, only the submodels MECCA and JVAL are used. All required parameters are prescribed as time series data via IMPORT_TS. The initialisation of the chemical species, if required, is taken from the same restart file as in the previous example. As photolysis frequencies depend on the light intensity, the MESSy submodel JVAL requires information about the ozone column above the simulated chemical box. This is no problem for a 3-dimensional dynamical model (additional information about the upper-atmosphere ozone content is always required as input to JVAL); however, if only one single box should be
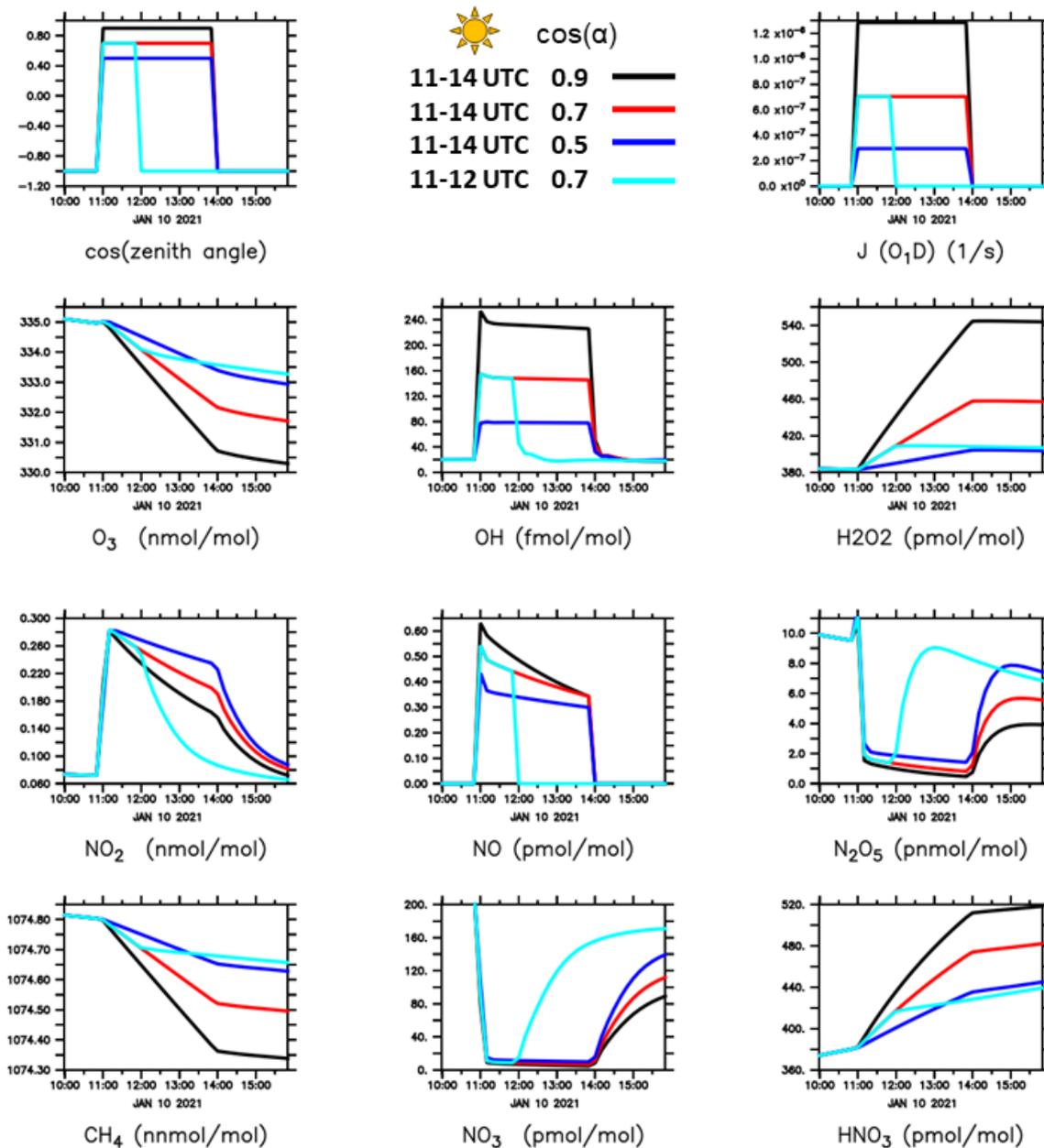
Figure 10. Time evolution of different parameters and chemical species in a chemical box-model simulation. Here, the cosine of the solar zenith angle (upper-left panel) is the input parameter varied (see also Table 4).

simulated, this profile information is missing. To enable the usage of JVAL also for single box-model experiments, JVAL has been expanded by the option to use an artificial profile as vertical column information. Appendix B provides further information about this expansion of JVAL.

The following two examples are shown to demonstrate how sensitivity studies could be easily performed with such a box model.

4.3.1 Sensitivity to solar radiation

Four simulations (Box1 in Tables 2 and 3) have been performed, varying only one input parameter: the cosine of the solar zenith angle. Tables 3 and 4 provide an overview of the different simulations. The simulations were started at midnight in darkness. The sun was "switched on" at 11:00 UTC and "switched off" again at 14:00 UTC in three of the simulations, while in the simulation depicted in light blue, darkness prevails already from 12:00 UTC on.

Table 5. Emission times of OH and NO in the three simulations of the idealised kinetic box-model study (Box2) and the respective line colour in Fig. 11.

OH emission time (UTC)	NO emission time (UTC)	Line colour
14	12	black
12	14	red
12	12	blue

Exemplary for all, the photolysis frequency of the reaction J (O^{1}D): $O_3 + h\nu \rightarrow O(^1D) + O_2$ is shown in Fig. 10. Naturally, the photolysis frequency depends on the solar zenith angle: the higher the sun above the horizon, the larger the photolysis frequency. This artificially triggered onset and offset of photochemistry is mirrored in the time evolution of all nine chemical species depicted in the remaining nine panels. The higher the photolysis frequencies are, the faster

- ozone decreases (mainly due to $O_3 + h\nu \rightarrow O(^1D) + O_2$),

- OH increases (mainly driven by $O(^1D) + H_2O \rightarrow 2OH$),

- H_2O_2 increases (mainly due to $2HO_2 \rightarrow H_2O_2 + O_2$),

- methane decreases (mainly driven by $CH_4 + OH \rightarrow CH_3 + H_2O$), and

- HNO_3 increases (mainly due to $NO_2 + OH \rightarrow HNO_3$).

The other species show a much more complicated time evolution, visualising the non-linear interactions in atmospheric chemistry.

4.3.2 Idealised experiment

The second example (Box2) presents a very idealised setup. In a simulation running from 10:00 to 18:00 UTC, all chemical species except for ozone are initialised with a value of zero. Ozone is initialised with a mixing ratio of 330 nmol mol^{-1}. Photochemistry is switched off as the cosine of the solar zenith angle is set to -1. The simulation setups trigger OH and NO emissions at 12:00 UTC and/or 14:00 UTC. Table 5 provides an overview of the setups. The results (Fig. 11) show a clear dependence of the production and destruction of the individual species on the emission times of NO and OH:

- Ozone is more efficiently destroyed by NO. If OH and NO are emitted at the same time, competing reactions lead to overall less ozone destruction ($=$ larger ozone mixing ratio at the end of the simulation).

- If only OH is emitted first, HO_2 forms most efficiently because nitrogen oxides do not compete with the HO_2 building reactions $O_3 + OH \rightarrow HO_2 + O_2$. Consequently, H_2O_2 is also built most efficiently if OH is emitted first.

- Nitrogen oxides can only be built if NO is emitted. NO_2 is produced most efficiently, along with NO_3 and consequently N_2O_5 buildup, if NO is emitted into the system.

- HNO_3 can only be built ($NO_2 + OH \rightarrow HNO_3$) if hydrogen and nitrogen are both added to the system. It forms most efficiently if OH and NO are emitted simultaneously. By far less HNO_3 forms if OH is added first as, in this case, most hydrogen ends up in H_2O_2, which cannot be converted to HNO_3 without photochemistry.

4.4 Performance tests: GPU port of chemical kinetics

This last example illustrates the usefulness of the DWARF for performance analysis. Analysing a specific submodel is much easier if the model configuration can be significantly simplified compared to complex base models. A smaller code base allows for a simpler and faster usage of performance analysis tools as the time required to create the runtime profiles is significantly reduced.

Here, the same simple 3-dimensional setup, as described in Sect. 4.2, was used to analyse and compare the runtime of the MECCA submodel operated either entirely on CPUs or executing the chemical kinetic calculations on GPUs. The MEDINA tool (Alvanos and Christoudias, 2017; Christoudias et al., 2021) was used to create the CUDA code of the kinetics integration.

Additionally, MECCA ran with two different reaction mechanisms: firstly, the standard mechanism (MIM, Mainz Isoprene Mechanism), taking into account 159 species and 360 reactions. The more complex mechanism based on MOM (Mainz Organic Mechanism, Sander et al., 2019) has 729 species with 2193 reactions. Obviously, due to the larger matrix to be solved, it is expected that the more complex mechanism shows an even greater speedup on GPU. Profiling was done with the NVIDIA Nsight Systems tool (https://developer.nvidia.com/nsight-systems, last access: 25 March 2024) on the JUWELS BOOSTER high-performance computing (HPC) system (Jülich Supercomputing Centre, 2021) by adding special markers to the DWARF code to enable the tracing of the code parts which are of interest here. On this machine, one computing node equipped with two AMD EPYC 7402 processors and four NVIDIA A100 40GB was used. For the CPU run, the GPUs were disabled. The runtime[4] of the integration phase for these variants are displayed in Table 6. Additionally, Fig. 12 visualises the tracing displayed by the NVIDIA Nsight Systems tool for two subsequent time-loop passes during the integration.

---

[4]The first time step was excluded to avoid measuring the one-time GPU initialisation.
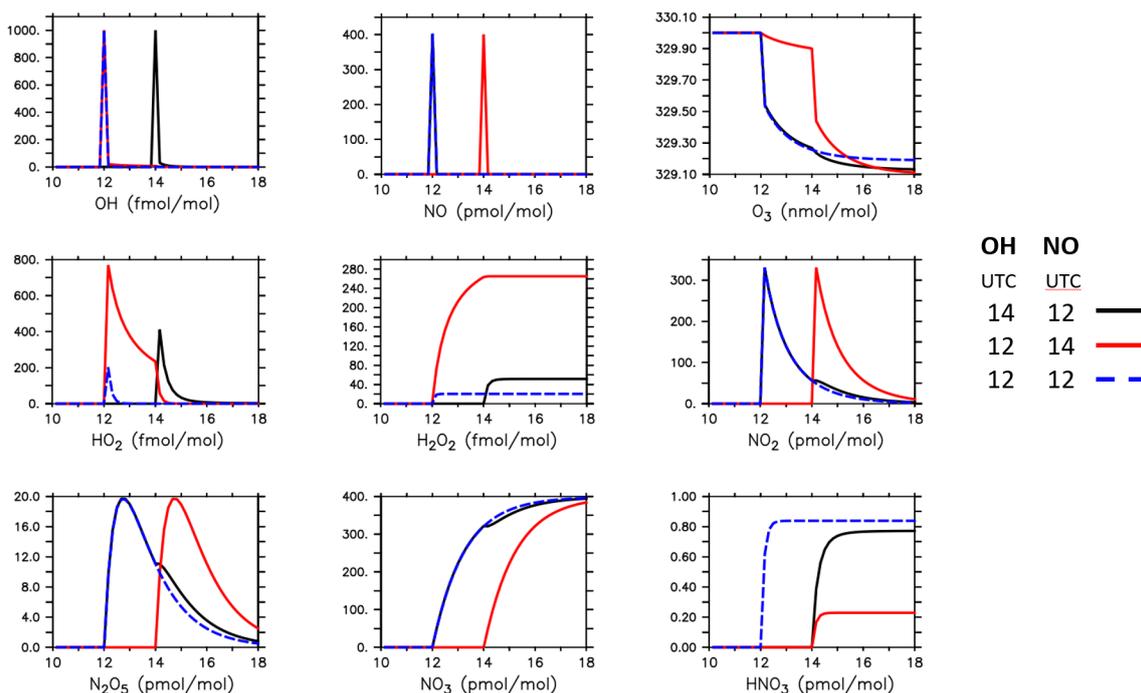
Figure 11. Time evolution of chemical species in a chemical box-model simulation. Only ozone (O_3) was initialised. OH and NO were emitted at 12:00 and 14:00 UTC in different order (see the main text and Table 5 for the respective line colours).

Table 6. Runtime comparison between CPU and GPU for two chemical mechanisms (MIM and MOM) on one JUWELS BOOSTER node.

	Overall runtime [s]		Speedup	Kinetics runtime [s]		Speedup
	CPU	GPU		CPU	GPU	
MIM	23.6	10	2.36	16.5	3.21	5.14
MOM	244.1	58	4.21	213.03	27.71	7.69

Here, `mecca_physc` contains the call to the kinetic solver. Obviously, running the kinetic solver on GPUs speeds up not only the integration of the kinetics per se (Table 6, right), but also the overall runtime (Table 6, left) is reduced by a substantial amount, i.e. a factor 2.36 and 4.21 for MIM and MOM, respectively. The more complex chemistry shows a much higher speedup as the larger vector of chemical species results in more calculations, which can utilise the capability of the GPUs better.

5 Summary and outlook

The new MESSy generic base model DWARF is presented here. It was developed to provide the basis for a unified testing of individual MESSy submodels for model performance analysis and improvement and, last but not least, as a basis for setting up simplified models (e.g. box and column models) in a standardised way using all functionalities provided by the MESSy framework.

The examples presented in Sect. 4 of this article illustrate that the DWARF works, as intended, as a simple process submodel test tool (Sect. 4.1), as a 3-dimensional atmospheric chemistry multi-box model (Sect. 4.2), as a 0-dimensional box model for atmospheric chemistry sensitivity simulations (Sect. 4.3), and for performance tests when porting an individual submodel to GPUs (Sect. 4.4).

While these are basically technical further developments of the infrastructure of the DWARF, the DWARF has already been used for a larger technical development project in the framework of the natESM project (https://www.nat-esm.de, last access: 12 February 2025). The aim of the project was to develop a concept for a MESSy infrastructure expansion to trigger copies between CPU and GPU memory most efficiently in MESSy setups, where only parts of the MESSy submodels are ported to GPUs. To enable an easy start for the programmer from the natESM project, we developed a DWARF setup employing only some artificial tracers and with five specialised submodels using the basic data handling and communication procedures of the MESSy infrastructure
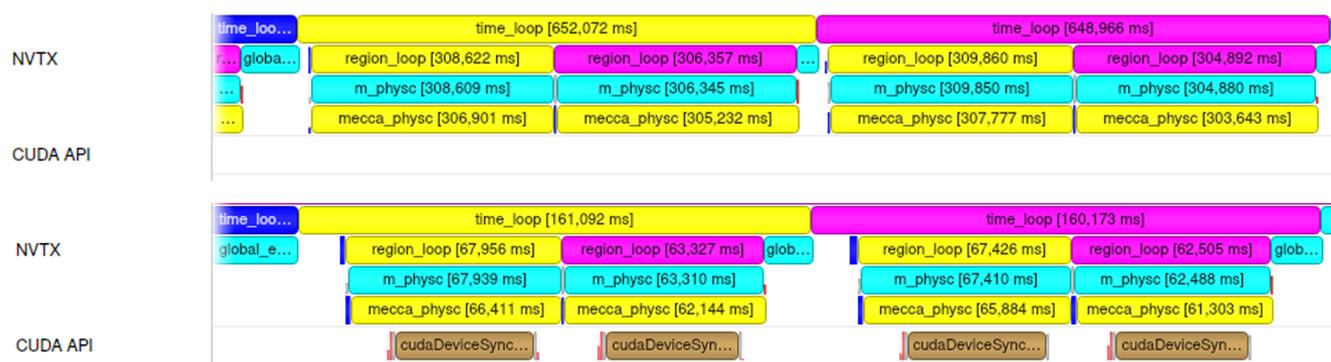
Figure 12. Example of an NVIDIA Tools Extension Library (NVTX) tracing with the NVIDIA Nsight Systems tool for the DWARF integration of the MIM chemical mechanism. Visualised are two subsequent time steps, including the call of the kinetics integration (upper panel: CPU only, lower panel:kinetics integration on GPUs).

(i.e. from the TRACER, CHANNEL, and TENDENCY infrastructure submodels). With this, the programmer could start relatively quickly with the actual code developments, while when using a comprehensive MESSy setup that includes a legacy dynamic model and atmospheric chemistry, it would have taken much longer for the programmer to understand the concepts and methods of the MESSy infrastructure submodels[5].

The MESSy DWARF is also suitable for scientific application. Within a PhD project, the chemistry box-model DWARF setup as presented in Sect. 4.3 is further developed to be applicable for the analysis of experiments as, for example, performed within the atmospheric simulation chambers SAPHIR (Novelli et al., 2018; Franco et al., 2021), SAPHIR-STAR (Baker et al., 2024), and BATCH (Löher et al., 2024). This way, the improvements in the multiphase chemical kinetic model of MECCA, e.g. recent ones, such as those in Soni et al. (2023) and Wieser et al. (2024), can be readily applied and tested in the MESSy framework for the global troposphere (Rosanka et al., 2024).

The implementation of the DWARF is at some points limited and has the potential to be improved further. For example, in the future, the rather inflexible decomposition and corresponding MPI communication shall be performed via YAXT (Yet Another eXchange Tool, https://dkrz-sw.gitlab-pages.dkrz.de/yaxt, last access: 12 February 2025) to make it more flexible. Among other advantages, this helps to get at least the MESSy DWARF code independent of licence-restricted software parts, which currently prevent MESSy from becoming open-source. In another ongoing project, the simple grid currently defined for DWARF will be optionally replaceable by grids defined by the t8code library (https://dlr-amr.github.io/t8code, last access: 12 February 2025), which features adaptive mesh refinement.

---

[5]The development of the MESSy infrastructure expansion for GPUs will be published elsewhere once it has been finalised.

Appendix A: The CHANNEL submodel

The MESSy infrastructure submodel CHANNEL provides data types and methods for the memory management and data export within MESSy. It is described in detail by Jöckel et al. (2010) and the corresponding supplement. Here, we briefly explain the terms used in the present paper. For further details, please refer to Jöckel et al. (2010).

CHANNEL organises the memory in so-called *channels*, which consist of the so-called *channel objects*. A channel object is an individual data object comprising a pointer to the actual memory storing the contents of a variable and additional metadata, such as a name and further attributes describing the object in more detail (e.g. units and descriptive long names). A channel object moreover contains information about the geometry of the object, i.e. its spatial dimensions and information about its parallel decomposition. To access a channel object, two strings need to be specified, i.e. (i) the name of the channel and (ii) the name of the channel object itself.

Appendix B: External vertical profile information in JVAL

As discussed in Sect. 4.3, JVAL requires the vertical profile of some variables. In a MESSy configuration coupled to a 3-dimensional dynamical model, the column information is present, and additional information about the ozone profile even above the top of the dynamical model is imported via IMPORT_GRID (see Sander et al., 2014, for further details). If JVAL is used in a single box-model simulation, no such column information is available. To close this gap, JVAL was extended to use externally provided vertical profile information in box-model applications. This mode is triggered by setting the `&CTRL` namelist parameter `l_artif_vprof` to `.TRUE.`. In this case, the additional namelist `&CPL_PROF` is read:

```
&CPL_PROF
prof_nlev  = 19,
inp_prof_temp  ='import_ts','tempprof',
inp_prof_press ='import_ts','pressprof',
inp_prof_rhum  ='import_ts','rhumprof',
inp_prof_clp   ='import_ts','clpprof',
inp_prof_aclc  ='import_ts','aclcprof',
inp_prof_o3    ='import_ts','O3_H',
inp_prof_v3    ='import_ts','V3_H',
inp_prof_slf   ='import_grid','inp2d_slf',
/
```

`prof_nlev` defines the number of vertical levels. In the example, 19 levels are provided. The required information are temperature (K), pressure (Pa), relative humidity (%), cloudiness (–), cloud cover (–), ozone mixing ratio ($\text{mol}\,\text{mol}^{-1}$) in the respective layer, and ozone column above the respective level (molecules cm^{-2}). These data are read via IMPORT_TS. Furthermore, the sea–land fraction is required as input. In the example, this is provided via a remapping of a 2-dimensional field by IMPORT_GRID.

Appendix C: The MESSy submodel PTRACINI

The submodel PTRACINI (Prognostic TRACer INItialisation) was developed and used for stratosphere–troposphere transport diagnostics by Hofmann et al. (2016). In the meantime, the submodel was expanded. Most importantly, while Hofmann et al. (2016) prescribed the initial value to be $10^{-7}\,\text{mol}\,\text{mol}^{-1}$, the initial value can now be provided by a namelist as either a constant or a channel object. The following overview of the submodel PTRACINI focuses on those functionalities required for the DWARF box-model applications as presented in Sect. 4.3.

PTRACINI allows for the emission of tracers (defined elsewhere) dependent on prescribed criteria. The emission proceeds

– either once, at one single point in time, indicated by the parameter `TRINI(.)%INI1STEP`, or

– in regular intervals if the interval is provided via an event, e.g. `TRINI(.)%EMIS_IOEVENT = 1,'hours','first',0`.

The time of the single point emission is defined via the parameter `TRINI(.)%EVENT_START`. See the TIMER manual available in Jöckel et al. (2010) for further information about events.

For the definitions of the criteria, the following operators are available:

`> , >= , < , and <=`.

Figure C1 displays an example namelist file, `ptracini.nml`, showing how the criteria can be defined:

1. In the first example (`TRINI(1)`), the tracer NO is initialised to the mixing ratio of $4 \times 10^{-10}\,\text{mol}\,\text{mol}^{-1}$ at one single time step, i.e. on 10 January 2021, 12:00 UTC.

2. In the second example (`TRINI(2)`), the tracer OH is initialised and repeatedly reset to a mixing ratio of $10^{-10}\,\text{mol}\,\text{mol}^{-1}$. The initialisation takes place on 10 January 2021, 14:00 UTC, and is reset to this value every hour.

3. The third example (`TRINI(3)`) is similar to the second example. The difference here is the definition of the event. By defining the event to `1,'steps'` the mixing ratio is reset every time step back to $10^{-10}\,\text{mol}\,\text{mol}^{-1}$.

4. The last example shows how criteria can be used. In this case, the tracer 4 pvu is initialised if the pressure is between 900 and 150 hPa. Additionally, the potential vorticity (provided by the submodel TROPOP) needs to be larger than 4 PVU, and the specific humidity has to be smaller than $0.001\,\text{kg}\,\text{kg}^{-1}$.

```
!-------------------------------------------------------------------
! -*- f90 -*-
&CPL
inp_press = 'dwarf','press'

! SYNTAX for SET:
!INI1STEP:      - for onetime initialisation (if T, EMIS_IOEVENT is ignored,
!                  if F, EVENT_START is ignored )
!EVENT_START:   - set for onetime initialisation, is ignored for INI1STEP=F
!               - exact date to start the event (has to be a multiple of the
!                 timestep, because exact time is prompted)
!EMIS_IOEVENT:  - set for continous initialisation, is ignored for INI1STEP=T
!               - use step-intervall (otherwise oszillation for leapfrog scheme)
!IFNOT:         - if T, field is initialised when crits are NOT fullfilled
!
!TRACER: 'name_tr1', 'subname_tr1', 'name_tr2', 'subname_tr2', ...
!CRIT:    'channel', 'object', 'crit', const(_dp!) , 'compfld_cha','compfld_obj'

TRINI(1)%INI1STEP = T
TRINI(1)%EVENT_START = 2021,01,10,12,00,0
TRINI(1)%TRACER = 'NO','','','','','','','','','','',
TRINI(1)%CRIT(1)='','','IF',4.e-10,'',''
!
TRINI(2)%INI1STEP = F
TRINI(2)%EVENT_START = 2021,01,10,14,00,0
TRINI(2)%EMIS_IOEVENT  = 1,'hours','first',0
TRINI(2)%TRACER = 'OH','','','','','','','','','','',
TRINI(2)%CRIT(1)='','','IF',1.e-12,'',''
!
TRINI(3)%INI1STEP = F
TRINI(3)%EVENT_START = 2021,01,10,12,00,0
TRINI(3)%EMIS_IOEVENT  = 1,'steps','first',0
TRINI(3)%TRACER = 'HNO3','','','','','','','','','','',
TRINI(3)%CRIT(1)='','','IF',1.e-12,'',''
!
TRINI(4)%INI1STEP = T
TRINI(4)%EVENT_START = 2021,01,10,12,06,0
TRINI(4)%IFNOT = F
TRINI(4)%TRACER = '4pvu','','','','','','','','','','',
TRINI(4)%CRIT(1)='${MINSTANCE[$i]}','press','<=',90000,'',''
TRINI(4)%CRIT(2)='${MINSTANCE[$i]}','press','>=',15000,'',''
TRINI(4)%CRIT(3)='tropop','PV','>=',4,'',''
TRINI(4)%CRIT(4)='tracer_gp','qv','<=',0.001,'',''
/
!-------------------------------------------------------------------
```

Figure C1. Example namelist file, `ptracini.nml`, of the MESSy submodel PTRACINI.

References

Alvanos, M. and Christoudias, T.: GPU-accelerated atmospheric chemical kinetics in the ECHAM/MESSy (EMAC) Earth system model (version 2.52), Geosci. Model Dev., 10, 3679–3693, https://doi.org/10.5194/gmd-10-3679-2017, 2017.

Baker, Y., Kang, S., Wang, H., Wu, R., Xu, J., Zanders, A., He, Q., Hohaus, T., Ziehm, T., Geretti, V., Bannan, T. J., O'Meara, S. P., Voliotis, A., Hallquist, M., McFiggans, G., Zorn, S. R., Wahner, A., and Mentel, T. F.: Impact of HO_2/RO_2 ratio on highly oxygenated α-pinene photooxidation products and secondary organic aerosol formation potential, Atmos. Chem. Phys., 24, 4789–4807, https://doi.org/10.5194/acp-24-4789-2024, 2024.

Christoudias, T., Kirfel, T., Kerkweg, A., Taraborrelli, D., Moulard, G.-E., Raffin, E., Azizi, V., van den Oord, G., and van Werkhoven, B.: GPU Optimizations for Atmospheric Chemical Kinetics, in: The International Conference on High Performance Computing in Asia-Pacific Region, HPC Asia 2021, Association for Computing Machinery, New York, NY, USA, 136–138, ISBN 978-1-4503-8842-9, https://doi.org/10.1145/3432261.3439863, 2021.

Dietmüller, S., Jöckel, P., Tost, H., Kunze, M., Gellhorn, C., Brinkop, S., Frömming, C., Ponater, M., Steil, B., Lauer, A., and Hendricks, J.: A new radiation infrastructure for the Modular Earth Submodel System (MESSy, based on version 2.51), Geosci. Model Dev., 9, 2209–2222, https://doi.org/10.5194/gmd-9-2209-2016, 2016.

Doms, G. and Baldauf, M.: A Description of the Nonhydrostatic Regional COSMO-Model, Part-I: Dynamics and Numerics, Tech. Rep., Deutscher Wetterdienst, Consortium for Small-Scale Modelling, https://doi.org/10.5676/DWD_pub/nwv/cosmo-doc_6.00_I, 2021.

Eichinger, R. and Jöckel, P.: The generic MESSy submodel TENDENCY (v1.0) for process-based analyses in Earth system models, Geosci. Model Dev., 7, 1573–1582, https://doi.org/10.5194/gmd-7-1573-2014, 2014.

Franco, B., Blumenstock, T., Cho, C., Clarisse, L., Clerbaux, C., Coheur, P.-F., De Mazière, M., De Smedt, I., Dorn, H.-P., Emmerichs, T., Fuchs, H., Gkatzelis, G., Griffith, D. W. T., Gromov, S., Hannigan, J. W., Hase, F., Hohaus, T., Jones, N., Kerkweg, A., Kiendler-Scharr, A., Lutsch, E., Mahieu, E., Novelli, A., Ortega, I., Paton-Walsh, C., Pommier, M., Pozzer, A., Reimer, D., Rosanka, S., Sander, R., Schneider, M., Strong, K., Tillmann, R., Van Roozendael, M., Vereecken, L., Vigouroux, C., Wahner, A., and Taraborrelli, D.: Ubiquitous atmospheric production of organic acids mediated by cloud droplets, Nature, 593, 233–237, https://doi.org/10.1038/s41586-021-03462-x, 2021.

Hofmann, C., Kerkweg, A., Hoor, P., and Jöckel, P.: Stratosphere-troposphere exchange in the vicinity of a tropopause fold, Atmos. Chem. Phys. Discuss. [preprint], https://doi.org/10.5194/acp-2015-949, 2016.

Jöckel, P., Sander, R., Kerkweg, A., Tost, H., and Lelieveld, J.: Technical Note: The Modular Earth Submodel System (MESSy) – a new approach towards Earth System Modeling, Atmos. Chem. Phys., 5, 433–444, https://doi.org/10.5194/acp-5-433-2005, 2005.

Jöckel, P., Kerkweg, A., Buchholz-Dietsch, J., Tost, H., Sander, R., and Pozzer, A.: Technical Note: Coupling of chemical processes with the Modular Earth Submodel System (MESSy) submodel TRACER, Atmos. Chem. Phys., 8, 1677–1687, https://doi.org/10.5194/acp-8-1677-2008, 2008.

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, https://doi.org/10.5194/gmd-3-717-2010, 2010.

Jöckel, P., Tost, H., Pozzer, A., Kunze, M., Kirner, O., Brenninkmeijer, C. A. M., Brinkop, S., Cai, D. S., Dyroff, C., Eckstein, J., Frank, F., Garny, H., Gottschaldt, K.-D., Graf, P., Grewe, V., Kerkweg, A., Kern, B., Matthes, S., Mertens, M., Meul, S., Neumaier, M., Nützel, M., Oberländer-Hayn, S., Ruhnke, R., Runde, T., Sander, R., Scharffe, D., and Zahn, A.: Earth System Chemistry integrated Modelling (ESCiMo) with the Modular Earth Submodel System (MESSy) version 2.51, Geosci. Model Dev., 9, 1153–1200, https://doi.org/10.5194/gmd-9-1153-2016, 2016.

Jones, P.: First- and Second-Order Conservative Remapping Schemes for Grids in Spherical Coordinates, Mon. Weather Rev., 127, 2204–2210, 1999.

Jülich Supercomputing Centre: JUWELS Cluster and Booster: Exascale Pathfinder with Modular Supercomputing Architecture at Juelich Supercomputing Centre, Journal of Large-Scale Research Facilities, 7, A183, https://doi.org/10.17815/jlsrf-7-183, 2021.

Kerkweg, A.: Data and plot scripts used in "The MESSy DWARF (based on MESSy v2.55.2)" Kerkweg et al., GMD 2024, Zenodo [data set], https://doi.org/10.5281/zenodo.14355764, 2024.

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, https://doi.org/10.5194/gmd-5-87-2012, 2012.

Kerkweg, A. and Jöckel, P.: The infrastructure MESSy submodels GRID (v1.0) and IMPORT (v1.0), Geosci. Model Dev. Discuss., 8, 8607–8633, https://doi.org/10.5194/gmdd-8-8607-2015, 2015.

Kerkweg, A., Hofmann, C., Jöckel, P., Mertens, M., and Pante, G.: The on-line coupled atmospheric chemistry model system MECO(n) – Part 5: Expanding the Multi-Model-Driver (MMD v2.0) for 2-way data exchange including data interpolation via GRID (v1.0), Geosci. Model Dev., 11, 1059–1076, https://doi.org/10.5194/gmd-11-1059-2018, 2018.

Löher, F., Borrás, E., Muñoz, A., and Nölscher, A. C.: Characterization of a new Teflon chamber and on-line analysis of isomeric multifunctional photooxidation products, Atmos. Meas. Tech., 17, 4553–4579, https://doi.org/10.5194/amt-17-4553-2024, 2024.

Messer, O. B., D'Azevedo, E., Hill, J., Joubert, W., Berrill, M., and Zimmer, C.: MiniApps derived from production HPC applications using multiple programing models, Int. J. High Perform. C., 32, 582–593, https://doi.org/10.1177/1094342016668241, 2018.

Müller, A., Deconinck, W., Kühnlein, C., Mengaldo, G., Lange, M., Wedi, N., Bauer, P., Smolarkiewicz, P. K., Diamantakis, M., Lock, S.-J., Hamrud, M., Saarinen, S., Mozdzynski, G., Thiemert, D., Glinton, M., Bénard, P., Voitus, F., Colavolpe, C., Marguinaud, P., Zheng, Y., Van Bever, J., Degrauwe, D., Smet, G., Termonia, P., Nielsen, K. P., Sass, B. H., Poulsen, J. W., Berg, P., Osuna, C., Fuhrer, O., Clement, V., Baldauf, M., Gillard, M., Szmelter, J., O'Brien, E., McKinstry, A., Robinson, O., Shukla, P., Lysaght, M., Kulczewski, M., Ciznicki, M., Piątek, W., Ciesielski, S., Błażewicz, M., Kurowski, K., Procyk, M., Spychala, P., Bosak, B., Piotrowski, Z. P., Wyszogrodzki, A., Raffin, E., Mazauric, C., Guibert, D., Douriez, L., Vigouroux, X., Gray, A., Messmer, P., Macfaden, A. J., and New, N.: The ESCAPE project: Energy-efficient Scalable Algorithms for Weather Prediction at Exascale, Geosci. Model Dev., 12, 4425–4441, https://doi.org/10.5194/gmd-12-4425-2019, 2019.

Novelli, A., Kaminski, M., Rolletter, M., Acir, I.-H., Bohn, B., Dorn, H.-P., Li, X., Lutz, A., Nehr, S., Rohrer, F., Tillmann, R., Wegener, R., Holland, F., Hofzumahaus, A., Kiendler-Scharr, A., Wahner, A., and Fuchs, H.: Evaluation of OH and HO_2 concentrations and their budgets during photooxidation of 2-methyl-3-butene-2-ol (MBO) in the atmospheric simulation chamber SAPHIR, Atmos. Chem. Phys., 18, 11409–11422, https://doi.org/10.5194/acp-18-11409-2018, 2018.

Rockel, B., Will, A., and Hense, A.: The Regional Climate Model COSMO-CLM (CCLM), Meteorologische Zeitschrift, 17, 347–348, 2008.

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.

Rosanka, S., Tost, H., Sander, R., Jöckel, P., Kerkweg, A., and Taraborrelli, D.: How non-equilibrium aerosol chemistry impacts particle acidity: the GMXe AERosol CHEMistry (GMXe–AERCHEM, v1.0) sub-submodel of MESSy, Geosci. Model Dev., 17, 2597–2615, https://doi.org/10.5194/gmd-17-2597-2024, 2024.

Sander, R., Kerkweg, A., Jöckel, P., and Lelieveld, J.: Technical note: The new comprehensive atmospheric chem-

istry module MECCA, Atmos. Chem. Phys., 5, 445–450, https://doi.org/10.5194/acp-5-445-2005, 2005.

Sander, R., Jöckel, P., Kirner, O., Kunert, A. T., Landgraf, J., and Pozzer, A.: The photolysis module JVAL-14, compatible with the MESSy standard, and the JVal PreProcessor (JVPP), Geosci. Model Dev., 7, 2653–2662, https://doi.org/10.5194/gmd-7-2653-2014, 2014.

Sander, R., Baumgaertner, A., Cabrera-Perez, D., Frank, F., Gromov, S., Grooß, J.-U., Harder, H., Huijnen, V., Jöckel, P., Karydis, V. A., Niemeyer, K. E., Pozzer, A., Riede, H., Schultz, M. G., Taraborrelli, D., and Tauer, S.: The community atmospheric chemistry box model CAABA/MECCA-4.0, Geosci. Model Dev., 12, 1365–1385, https://doi.org/10.5194/gmd-12-1365-2019, 2019.

Soni, M., Sander, R., Sahu, L. K., Taraborrelli, D., Liu, P., Patel, A., Girach, I. A., Pozzer, A., Gunthe, S. S., and Ojha, N.: Comprehensive multiphase chlorine chemistry in the box model CAABA/MECCA: implications for atmospheric oxidative capacity, Atmos. Chem. Phys., 23, 15165–15180, https://doi.org/10.5194/acp-23-15165-2023, 2023.

The MESSy Consortium: The Modular Earth Submodel System (2.55.2-dwarf), Zenodo [code], https://doi.org/10.5281/zenodo.12089833, 2023a.

The MESSy Consortium: The Modular Earth Submodel System, Zenodo [code], https://doi.org/10.5281/zenodo.8360186, 2023b.

U.S. Government Printing Office, Washington, D.: U.S. Standard Atmosphere, https://www.ngdc.noaa.gov/stp/space-weather/online-publications/miscellaneous/us-standard-atmosphere-1976/us-standard-atmosphere_st76-1562_noaa.pdf (last access: 12 February 2025), 1976.

Wieser, F., Sander, R., Cho, C., Fuchs, H., Hohaus, T., Novelli, A., Tillmann, R., and Taraborrelli, D.: Development of a multiphase chemical mechanism to improve secondary organic aerosol formation in CAABA/MECCA (version 4.7.0), Geosci. Model Dev., 17, 4311–4330, https://doi.org/10.5194/gmd-17-4311-2024, 2024.