gmd-2024-117 – Reply to referee #1

Dear referee #1,

Thank you very much for your supportive in-depth review, which has certainly helped to improved our manuscript considerably. Please find our replies to your comments below. Your original comments are repeated in italics, our replies – for easier reading – are depicted in blue, normal font, and text passages which we included in the manuscript are in bold.

The authors of the article "The MESSy DWARF" describe The Modular Earth Submodel System (MESSy), a framework to integrate components to form an Earth system model. The topic of the article is an extension of the MESSy framework that simplifies the set-up and running of submodel(s) without the need for a full dynamical core to drive the simulation. Before this work, MESSy would always require a weather or climate forecast model as a "basemodel".

The underpinning concept of the presented approach is to substitute the basemodel and its role of providing the evolution of prognostic variables by a dummy component (the new submodel DWARFDCD). Prognostic variables are initialised to zero or, optionally, from input files and can be relaxed over time at a given rate. This approach provides other submodules with the relevant variables without the overhead of a basemodel, and its development is described as being inspired by the "dwarf strategy" developed and published in the ESCAPE project.

The authors highlight some differences to this dwarf strategy, namely the additional overhead incurred from always setting up a standard model with the entire coupling framework and infrastructure, which is notably more involved than the minimal drivers developed in ESCAPE, where each is bespoke to the corresponding component. This difference is being presented as an advantage because it allows to drive different components with the same setup. I do not fully subscribe to this point of view as the amount of required infrastructure is rather substantial, which can be a hindrance for technical exploration in experimental software stacks and the balance does not necessarily seem right for small submodels such as an individual microphysics parameterisation (one of the dwarves in ESCAPE). However, for larger submodels, e.g., related to atmospheric chemistry, which is a typical application scenario for MESSy, this seems less relevant, and overall I do agree that the gain in flexibility offers substantial advantages. In addition, one advantage that has not been listed explicitly, is the fact that the same submodel that has been developed and used with the DWARF can also be used without further changes coupled to a basemodel.

In fact, we deliberately avoided the word 'advantage' and instead used the words 'downside' and 'upside', which in our understanding are less judgemental, to indicate that the size of the benefit or harm depends on the application. We adjusted the phrasing a bit. Thanks for the hint, that we did not mention the stability of the interfaces. We changed the text to

While the larger overhead is definitely 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 basemodels according to the MESSy concept, a submodel developed or optimised in a DWARF configuration can be used directly in a fully-fledged basemodel without any changes.

To me, the most substantial issue of the article is the structure and presentation of the concept and implementation. This starts already in the introduction, which does not sufficiently motivate the reasons why this development has been undertaken in the MESSy framework, and instead circles back-and-forth to the ESCAPE ideas and the similarities and differences. Pointing these out is valid but shouldn't constitute the basis of the work. Some of what I would have liked to read in this section comes in the first paragraph of Sec. 4 instead.

Thanks for providing the hint, how to re-set the focus of the introduction. The revised introduction starts now with sentences similar to the first paragraph in Sect. 4, describes the basics of the DWARF concept and only after that relates to the idea of the ESCAPE project and the differences between the weather & climate dwarf concept and the MESSy DWARF concept. As we had to move major text blocks around, we do not cite the whole new introduction here. Please have a look at the revised article.

The introduction to the MESSy infrastructure and submodel concept in Sect. 2 is reasonably short and intuitive, with helpful pointers to additional resources for more details.

As none of the referees expressed a need for changes in Sect. 2 we kept it, as it was.

However, the description of the implementation in Sec. 3 dives immediately into technical details for individual infrastructure components of the MESSy framework. If this was instead motivated by the conceptual ideas of the dummy driver and the data requirements of a scientific submodel, it would benefit a wider audience beyond the users of the MESSy framework. This is notable, e.g., because more than once the relevant subsections start with statements similar to "usually this or that is defined/provided by the basemodel". In my opinion, if may suggest just one possibility: start with the description of the requirements of a hypothetical (or actual, as a case study) submodel component that is planned to be executed in a DWARF setting. Then explain how these inputs (e.g., prognostic variables), outputs (e.g., tendencies or diagnostic variables), and fundamental data structures (e.g., grid, domain decomposition), would be usually provided in the MESSy framework, thus highlighting the importance of the basemodel for many of these. And ultimately, detail how the DWARF setup and DWARFDCD replace the basemodel for providing these, amended with technical details about their implementation in the basemodel interface layer.

Thanks for this great idea. However, as the other two referees suggested that Section 3 is basically fine as it is, we decided to go with a compromise. The introduction of Sect. 3 now provides the information, what is required when using MECCA as the only regular submodel in a DWARF setup and how the input and output to / from MECCA is handled.

So far, MESSy was always 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 basemodel is basically "empty", all functionalities and data usually provided by the basemodel and made accessible via the MESSy infrastructure to the submodels, needs to be replaced, either by additional infrastructure functionalities (e.g. the definition of a grid), or by 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 basemodel and to illustrate the conception 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:**

   i) **the temperature and the specific humidity:**
   Temperature and specific humidity are prognostic variables. Dynamical 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 set-up where MECCA is the only used regular submodel, J-values need to be provided as external data. External data is read in via the MESSy infrastructure submodel IMPORT (see Sect. 3.1.2).

iii) the pressure at box mid-points:
The pressure is provided by the basemodel itself and referenced by the MESSy infrastructure submodel DATA or, if the basemodel does not provide a pressure field, it is calculated in DATA. Thus, also in the case of the MESSy DWARF, it is calculated or referenced by DATA (see Sect. 3.1.3). In 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 meta-data 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 DWARF as for legacy basemodels.

v) and 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 provided by the MESSy infrastructure submodel GRID_DEF (see in Sect. 3.2) independent of the basemodel.

2. the handling of the output data provided by MECCA:
MECCA solves the kinetic equations of the gas-phase chemistry and calculates the change of 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, is 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.

Importantly, also include a description of limitations. Stubs of all this can be found in introductory paragraphs to Sec. 3 and some of it's subsections, but is immediately lost in-between the technical details. My hope would be, embedding this into a better carved out storyline would better convey the clearly well though-out concepts and flexibility of the solution.

A new subsection 3.4 now lists specifically the limitations of DWARF setups:

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 depending 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 is available either from imported data or from other MESSy submodels. Missing data, however, in contrast to the first two points, will lead to an error during run-time telling the user, that data is missing.**

This is also reflected in the description of the examples in Sec. 4: While these are well-suited to illustrate the capability of the approach and the presented results appear convincing, I am missing technical details that would provide a connection to the DWARF implementation description in the previous section. For example, what specific variables and data structures were required by the submodel and how did the DWARF provide them?

We added this information:

- Sect. 4.1: **As a first, very simple example, the submodel ORBIT (Dietmüller et al., 2016) calculating orbital parameters is used. The only input ORBIT requires are the current time and the geographical location. These are provided in the same way for all basemodels including the MESSy DWARF by the infrastructure submodels TIMER and GRID_DEF, respectively.**

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

- Sect. 4.2: **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**

  1. **temperature, liquid water content, and ice water content:**
     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. **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 basemodels.
  3. **the cosine of the zenith angle and the distance between sun and earth:**
     These are provided by the MESSy submodel ORBIT.
  4. **the ozone column above the model top and the solar cycle data:**
     These are made available from external data via IMPORT_GRID for all basemodels.
  5. **pressure at box mid-points and vertical interfaces:**
     These are, in this DWARF setup, calculated in the infrastructure submodel DATA from the height grid defined in the infrastructure submodel GRID_DEF.
  6. **relative humidity, the land-sea fraction, the cloud-cover, and the albedo:**
     All four variables are usually provided by a dynamical basemodel, in this example, these are provided as external data, i.e., read by IMPORT_GRID.

- Sect. 4.3 and Sect. 4.4. do not require additional information, as this is contained in the previous examples or the information was already included.

Finally, a few minor remarks to specific figures/parts of the manuscript:

- *Figure 2 would benefit from some additional context in the caption, e.g., what nudging is active, what inputs are provided explicitly and which are omitted, etc. Moreover, I find the presence of commented namelist entries confusing (e.g., nudgedt_u).*

  Thanks for pointing that out. We cleaned up the figure and expanded the caption to:
  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 only for this a nudging coefficient is defined in the &CTRL namelist.

- *The labeling inside Figure 4 suggests that mgprow/mgpcol denote the number of grid points per direction, while the text in Sec. 3.2 refers to them as the number of grid boxes (which would imply the number of grid points is $mgprow + 1/mgpcol + 1$).*

  We corrected the figure.

- *The description of the decomposition in Sec. 3.3 is misleading: For example, "the latitude range covers 93 grid boxes, which are distributed among 2 tasks" is factually wrong, because these are distributed among 8 tasks. The latitude range is rather decomposed into two subranges, each of which is assigned to one of the two sets of compute tasks in latitudinal direction. The corresponding longitudinal range in each of these sets is then further subdivided and assigned to one of the four compute tasks assigned to the latitudinal subrange.*

  Thanks for spotting this misleading formulation. We used the word "segments" before, therefore we reformulated this paragraph:

  In our example, the latitude range covers 93 grid boxes, which are distributed among 2 segments. In this case, the tasks of the first segment get 46 grid boxes in 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 the task 0 gets $98x46$ grid boxes, while task 7 gets $99x47$ grid boxes.

- *The technical description of the performance benefits of the GPU port is incomplete. Performance numbers are presented without stating the used hardware (the mentioning of JUWELS-BOOSTER for the GPU numbers suggests that A100 GPUs were used but it is entirely unclear how many and to what CPUs this is being compared).*

  We added the missing information:
  On this machine one compute node equipped with 2 AMD EPYC 7402 processors and 4 NVIDIA A100 40GB was used. For the CPU run the GPUs were disabled.

- *The language is in some places rather informal, for example the use of "Anyhow" in the Code and data availability statement.*

  We removed the "anyhow" in the code availability section. However, detecting these "informal" language in additional parts is not so easy for non-native speakers. We hope that the "anyhow done" copy-editing will help in that respect.

Overall, the presented work is substantial, the concepts appear sound and the benefits and applicability promising. But the presentation in the article would benefit from a clearer structure and storyline.

Thanks again for this overall very positive review.

References

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), Geoscientific Model Development, 9, 2209–2222, https://doi.org/10.5194/gmd-9-2209-2016, 2016.

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), Geoscientific Model Development, 3, 717–752, https://doi.org/10.5194/gmd-3-717-2010, 2010.

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, Geoscientific Model Development, 5, 87–110, https://doi.org/10.5194/gmd-5-87-2012, 2012.

Rockel, B., Will, A., and Hense, A.: The Regional Climate Model COSMO-CLM (CCLM), Meteorologische Zeitschift, 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, Journal of Climate, 19, 3771–3791, 2006.

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), Geoscientific Model Development, 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, Geoscientific Model Development, 12, 1365–1385, https://doi.org/10.5194/gmd-12-1365-2019, 2019.