# COSMO-Model 6.0 - The Last Official Version

Ulrich Schättler

*Deutscher Wetterdienst*

## 1 Introduction

Already during the COSMO General Meeting in Eretria in 2014, one year after COSMO version 5.0 had been released, DWD announced that a limited-area mode of ICON will be developed to achieve a uniform modelling system for operational forecasting in the mid-term future. Three years later there was already clear evidence, that ICON-LAM will deliver significantly better forecasts than COSMO. This started the final phase of developments for the COSMO-Model and we put up a list of contributions for the last official version 6.0, which should be released at the end of 2019.

In this article we will summarize the activities from the last years and try to explain, why there was a delay of about two years until version 6.0 could really be published.

When starting to plan this last version in 2018, the implementation of a unified COSMO-ICON physics package just had finished with results that were not really satisfying. A deeper investigation of the differences between ICON and COSMO, which we will summarize in Section 2, strengthened the decision to only invest in the ICON-LAM from now on.

But of course the ongoing work for COSMO should be included in the last official version. The developments covered the COSMO-EULAG dynamical core and the outcomes from the priority project *Testing and Tuning the Revised Cloud-Radiation Coupling*. Also some tasks which were only about to start were added to the list of accepted contributions for 6.0, including the TERRA-URB scheme (from priority task AEVUS) and the new multi-layer snow-scheme SNOWPOLINO (from priority task SAINT). Short descriptions of these works are given in Section 3.

Besides the meteorological aspects, the results of the technical priority project *POMPA*, which ported the major parts of COSMO to GPUs, were also incorporated, which is described in Section 4.

And last, but not least, version 6.0 again is a *unified system*, which means that contributions from the CLM Community also have been taken over to the official COSMO version. The add-ons from the CLM group are listed in Section 5.

More information on all the implementation and development work can be found in the *(http://www.cosmo-model.org/content/model/releases/histories/default.htm)* Release Notes for the different versions and the pages for the priority projects and priority tasks.

## 2 The COSMO-ICON physics package

Between 2016 and 2018 a major restructuring of the COSMO-Model physics package has been implemented. Goal was to use the same code for the physical parameterizations like ICON, at least as far as possible. For such a unification of the physics, several issues had to be addressed:

- Memory layout, data structures
  The data structure, that has been implemented in all parameterizations, is the one from ICON, a 2-dimensional one `(number of grid points, vertical dimension)`, whereas the COSMO-Model uses a 3-dimensional memory layout and data structure, where horizontal fields are stored with 2 dimensions: `(ie,je, vertical dimension)`. The new 2-dimensional structure is referred to as the blocked data

structure. Because the rest of the COSMO-Model still is running in the traditional `ijk` data structure, the data have to be copied between these two structures, to call the physical parameterizations. Fig. 1 illustrates a horizontal field in the traditional COSMO `(ie,je)` data structure on the left side, where all grid points are stored in a 2-dimensional array. The blocked structure is depicted on the right side with a block length of `nproma` = 8 grid points.

- Interfaces
  All global fields (multidimensional arrays), which are used in a parameterization, are now passed by argument lists. All other (scalar) variables still can be accessed via `USE` statements.
- Naming conventions (of modules, routines, variables)
  In COSMO and in ICON there are different naming conventions for modules and routines, but also for variables. Where possible, the names have been unified.

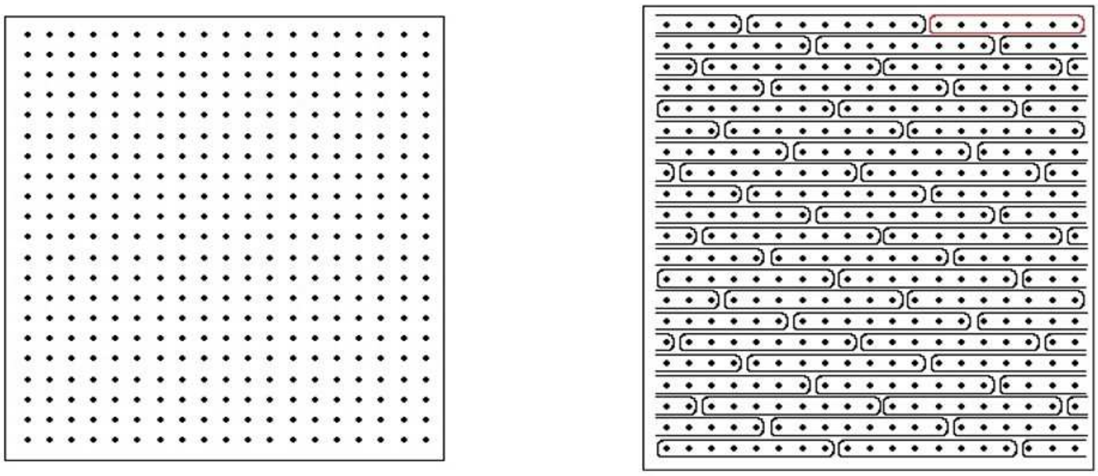

Figure 1: Horizontal field in `(ie,je)` (left) and blocked data structure `(nproma,nblock)` (right)

For the COSMO-ICON physics we do not store the full 3-dimensional fields. The long-term memory, to save values from one time step to another, still is in the traditional `(ie,je,ke)` structure. To run the physical parameterization for one step, we only copy the values for just one block to the new data structure and copy back the results to the `(ie,je,ke)` fields. This means, that more copy work has to be done, but less memory is consumed.

Figure 2 shows the parameterization packages used in COSMO and ICON. The ones with a blue background are not fully unified, but the scheme used is the same. All parameterizations with a green background really share the same source code with the same file names in COSMO and in ICON.

*Results of using the COSMO-ICON physics in the COSMO-Model*

Especially for the turbulence scheme and TERRA there have been significant modifications for the ICON implementations, that have now also been taken to COSMO. We refer to the COSMO Release Notes for version 5.04a (turbulence scheme) and 5.04e (TERRA) for further informations to these changes.

Alas, when running COSMO-DE experiments with the new versions, the results got worse compared to the former implementations. A major problem was the drying out of the soil during the simulated periods. And although COSMO was using nearly the same packages as ICON, there still were some significant differences. Among them are the treatment of external parameters, where COSMO reads *end products*, while ICON reads *raw data* and prepares them for using a tile-approach, which is not possible in COSMO. Also some parameterization components from ICON, which make direct use of land-use classes for example, cannot be transferred to COSMO.

| Scheme | COSMO | ICON |
|---|---|---|
| Microphysics | prognostic water vapour, cloud water, ice, rain, snow, graupel (Doms, 2004; Seifert, 2010) | |
| Radiation | Ritter-Geleyn δ two-stream | RRTM |
| Subgrid scale orography | Lott and Miller (1997) | |
| Turbulence | prognostic TKE scheme (Raschendorfer) | |
| Surface Schemes | TERRA (Heise and Schrodin, 2002)<br>FLake (Mironov)<br>SeaIce (from IFS) | |
| Convection | Tiedtke or shallow | |
| | Tiedtke-Bechtold (in COSMO only optional) | |

Figure 2: Physics packages unified between COSMO and ICON

Another major difference is that ICON runs a soil-moisture analysis, which was never implemented for COSMO-DE.

Because of these results, DWD decided to run the physics schemes taken from ICON in the old ”COSMO-style”, which was possible by choosing a special namelist configuration. But performing and evaluating the COSMO-DE experiments took some time, which resulted in a late delivery of COSMO-Model version 5.05. And now it was clear that there was at least no easy way to further improve COSMO forecasts.

## 3 Meteorological Developments

*Implementation of COSMO-EULAG dynamical core*

The first implementations of the COSMO dynamical cores (Leapfrog and Runge-Kutta) did have no explicit conservation properties, which is a crucial issue in fluid dynamics. To implement that, a priority project CDC (Conservative Dynamical Core; 2008-2012) was started. A subtask of the project was to investigate the anelastic EULAG dynamical core regarding its use in COSMO. Two further priority projects CELO (COSMO-EULAG operationalization; 2012-2017) and EX-CELO (Extension of COSMO-EULAG operationalization; 2017-2019) were conducted to integrate the EULAG dynamical core later on into the COSMO framework and also address data assimilation issues.

During these projects, the original EULAG dynamical core changed from an anelastic to a compressible formulation, which caused some delays of the developments. In the end, the implementation in COSMO version 5.07a in May 2020 came too late for a wide-spread operational use. Moreover, improvements had also been implemented in the existing Runge-Kutta dynamical core to tackle conservation issues and the problems of steep orography.

*Revised Cloud Radiation Coupling*

Radiation is the main source of earth’s energy and is strongly coupled to other elements of NWP models especially the heating and cooling rates. Following a Priority Task ”(RC)$^2$” (Revised Cloud Radiation Coupling), the Priority Project ”T$^2$(RC)$^2$” (Testing and Tuning the Revised Cloud Radiation Coupling) aimed at improving the cloud-radiation coupling. For more information see the *(http://www.cosmo-model.org/content/tasks/pastProjects/t2rc2/default.htm)* project description on the COSMO Web Page.

This project started in 2015 and was finished in 2020 shortly after implementing the code extensions in the official COSMO version 5.06b in October 2019.

*TERRA-URB*

The urban-canopy land-surface scheme TERRA_URB has originally been implemented in the COSMO CLM, version cclm-sp_2.4_terra_urb_2.2.1, and later also in the version cclm_urb_clm9_2.3 by Hendrik Wouters. These versions still used the classical `ijk` data structure. In 2018, TERRA_URB 2.3 from this latter version has been extracted and implemented in a test version based on COSMO-Model 5.05 using the blocked data structure for the physics. Quite some technical adaptations had to be implemented because of that. The test version has been used in the priority tasks *(http://www.cosmo-model.org/content/tasks/pastTasks/pt_aevus.priv.pdf)* AEVUS and *(http://www.cosmo-model.org/content/tasks/pastTasks/pt_aevus2.priv.pdf)* AEVUS2 (Analysis and Evaluation of TERRA_URB Scheme).

In 2021 this test version based on 5.05 has been finally implemented in the official COSMO version. The simple tile approach that is used in TERRA_URB for soil and surface variables needed considerable modifications in the framework of the model, especially for the I/O. While there was only a *quick and dirty* implementation in the test version, a *clean* implementation was done for COSMO version 5.09, which took some time.

*A new Multi-Layer Snow Model for COSMO*

The single layer snow model in TERRA has some limitations and draw-backs for numerical weather prediction, as e.g. the missing of a melting and freezing cycle. An already existing multi-layer snow model in COSMO gives better results but is not ready for operational use and is no longer developed and maintained due to lack of resources.

Here MeteoSwiss and the Swiss Institute for Snow and Avalanche Research in Davos (SLF) stepped in and started the priority task *(http://www.cosmo-model.org/content/tasks/priorityTasks/pt_saint.priv.pdf)* SAINT in 2017 to implement an improved snow model in COSMO. The task was extended a few times and was succeeded by the Working Group 3b task SNOWPOLINO in 2021.

The new snow model is essentially based on the model SNOWPACK from SLF and provides a good basis for future developments also in ICON. A first intensively tested version has been implemented in COSMO 6.0. But development is going on and the ICON version of SNOWPOLINO might already see further differences.

## 4 Porting the COSMO-Model to GPUs

In 2010 Switzerland started a major initiative to develop applications to run at scale and make efficient use of *next generation of supercomputers*. This initiative HP2C (Swiss Platform for High-Performance and High-Productivity Computing) has been succeeded later on by PASC, the Swiss Platform for Advanced Scientific Computing. Within these initiatives also projects were started to make the COSMO-Model aware of these emerging massively parallel computers. The COSMO priority project *(http://www.cosmo-model.org/content/tasks/pastProjects/pompa/default.htm)* POMPA (Performance on Massively Parallel Architectures; 2010-2015) supported these swiss initiatives.

The target architecture for POMPA were general purpose graphic cards (GP-GPUs), which need a different way of programming compared to CPUs. For the dynamical core, a rewrite using a domain specific language (DSL) has been chosen. The first prototype of the DSL called STELLA (stencil library) has been replaced some time ago by GRIDTOOLS. All other components of COSMO have been ported by adding OpenACC compiler directives to the Fortran code, which can be processed by a few compilers.

Another outcome of POMPA was a running single-precision version. To run the COSMO-Model in single precision has been aimed at from the very beginning of the development, but has never been tested. Therefore it was a major achievement to make this work.

COSMO version 5.07 was the first in which all components were ported to GPUs, which are used at MeteoSwiss and the Aeronautica Militare in Rome, who are running GPU systems.

## 5 Unification with CLM Versions

A goal for every major version of COSMO always was the re-unification with the code used in the CLM Community, the COSMO-CLM. This also was planned again for version 6.0. This time we got contributions from the COSMO-CLM, but also from the C2SM (Center for Climate Systems Modelling) group at the ETH in Zürich. Most of these developments have been implemented in version 5.08, unless stated otherwise.

*Contributions from CLM*

The CLM Community added several diagnostic output variables, for example for wind sector classes and for sunshine duration. Also new tuning variables have been introduced for several parameterization schemes.

In the module `src_setup_vartab.f90` only official NetCDF standard and long names are now used. Some of these names have been explicitely registered by the NetCDF community. If no official names exist, only a `'-'` appears (version 5.09b).

*Contributions from C2SM*

A new additional hydrology scheme (groundwater and runoff) has been developed by Linda Schlemmer during her stay at the ETH. This has been implemented in COSMO version 5.08 For this scheme and also for the radiation some additional diagnostic output variables have been added.

Colleagues from the ETH also worked on NetCDF I/O. In the last years they were using a COSMO version based on the work done at MeteoSwiss to run the model on GPUs. And because they are using a rather powerful GPU machine, they can run very big COSMO domains, which showed some limitations in NetCDF I/O. To overcome at least some of these limitations, a prefetching of NetCDF boundary data has been implemented (version 5.11) and some optimizations for asynchronous output have been done (version 5.09). And because classic NetCDF restricts the size of variables the possibility to work with NetCDF4 is available (version 5.09). Also, an online compression of NetCDF data using zlib can now be done (version 5.10).

Together with colleagues from the CLM group also the possibility to write restart files in NetCDF has been implemented (version 5.06b).

*Contributions from MESSy*

The interfaces to couple MESSy packages with COSMO and INT2LM have been updated. The MESSy group regularly is testing COSMO and INT2LM with the NAG Compiler (Numerical Algorithms Group), which does a very thorough testing of the Fortran Standard. Because of that we got several updates from the MESSy group to improve our codes.

One major contribution was the implementation of some MPI 3.0 interfaces, where the names and / or the functionality had changed. For COSMO the module `environment.f90` was affected, where MPI data types are used. The new interfaces are implemented with a pragma MPI3, and the old interfaces are retained for installations, which do not yet provide MPI 3.0 standard.

## 6 Summary

For more than 20 years the COSMO-Model served as an operational forecast model not only at DWD, but at about 30 national weather centers. Also it was used as a regional climate model by the CLM Community and as a research tool at numerous institutions worldwide.

We would like to thank all the users who contributed to the development and the improvement of the code and the documentation of the COSMO-Model in these two decades. It sure was an outstanding experience working together with all of you!

On 10th of February, 2021, DWD replaced its COSMO-D2 with the application ICON-D2, based on the limited-area mode of ICON. With that, no more COSMO application is now running at DWD. Nevertheless, all other partners still are using the COSMO-Model. While the consortium partners already started to migrate their work also to ICON-LAM, all other partners as the licensees or the CLM community may stay with COSMO for a bit longer. Therefore, support and maintenance for running the COSMO-Model still is ongoing, but there will be no more further developments. Maintenance now is limited to bug-fixes only.

With that we invite all of you to use the ICON-LAM in the future and start to migrate your work now.

Farewell to COSMO

Welcome to and Good Luck with ICON-LAM