Consortium for Small-Scale Modelling

A Description of the

Nonhydrostatic Regional COSMO-Model

Part V

Initial and Boundary Data for the COSMO-Model

U. Schättler and U. Blahak

INT2LM 2.08 February 2021

www.cosmo-model.org

Publisher
Deutscher Wetterdienst
Business Area “Research and Development”
Frankfurter Straße 135
63067 Offenbach
www.dwd.de

Editors
Ulrich Schättler, DWD
Ulrich.Schaettler@dwd.de

Contents

Section 1

Overview on the Model System

1.1 General Remarks

The *COSMO-Model* is a nonhydrostatic limited-area atmospheric prediction model. It has been designed for both operational numerical weather prediction (NWP) and various scientific applications on the meso-β and meso-γ scale. The COSMO-Model is based on the primitive thermo-hydrodynamical equations describing compressible flow in a moist atmosphere. The model equations are formulated in rotated geographical coordinates and a generalized terrain following height coordinate. A variety of physical processes are taken into account by parameterization schemes.

Besides the forecast model itself, a number of additional components such as data assimilation, interpolation of boundary conditions from a driving host model, and postprocessing utilities are required to run the model in NWP-mode, climate mode or for case studies. The purpose of the *Description of the Nonhydrostatic Regional COSMO-Model* is to provide a comprehensive documentation of all components of the system and to inform the user about code access and how to install, compile, configure and run the model.

The basic version of the COSMO-Model (formerly known as *Lokal Modell (LM)*) has been developed at the *Deutscher Wetterdienst* (DWD). The COSMO-Model and the triangular mesh global gridpoint model ICON form – together with the corresponding data assimilation schemes – the NWP-system at DWD. The subsequent developments related to the COSMO-Model have been organized within COSMO, the *Consortium for Small-Scale Modeling*. COSMO aims at the improvement, maintenance and operational application of a nonhydrostatic limited-area modeling system, which is now consequently called the COSMO-Model. The meteorological services participating to COSMO at present are listed in Table 1.1.

For more information about COSMO, we refer to the web-site at `www.cosmo-model.org`.

The COSMO-Model is available free of charge for scientific and educational purposes, especially for cooperational projects with COSMO members. However, all users are required to sign an agreement with a COSMO national meteorological service and to respect certain conditions and restrictions on code usage. For questions concerning the request and the agreement, please contact the chairman of the COSMO Steering Committee. In the case of a planned operational or commercial use of the COSMO-Model package, special regulations

Table 1.1: COSMO: Participating Meteorological Services

DWD	Deutscher Wetterdienst, Offenbach, Germany
MeteoSwiss	Meteo-Schweiz, Zürich, Switzerland
ITAF-ReMet	Ufficio Generale Spazio Aero e Meteorologia, Roma, Italy
HNMS	Hellenic National Meteorological Service, Athens, Greece
IMGW	Institute of Meteorology and Water Management, Warsaw, Poland
ARPA-SIMC	Agenzia Regionale per la Protezione Ambientale dellÂť Emilia-Romagna Servizio Idro Meteo Clima Bologna, Italy
ARPA-Piemonte	Agenzia Regionale per la Protezione Ambientale, Piemonte, Italy
CIRA	Centro Italiano Ricerche Aerospaziali, Italy
ZGeoBW	Zentrum für Geoinformationswesen der Bundeswehr, Euskirchen, Germany
NMA	National Meteorological Administration, Bukarest, Romania
RosHydroMet	Hydrometeorological Centre of Russia, Moscow, Russia
IMS	Israel Meteorological Service, Bet-Dagan, Israel

will apply.

The further development of the modeling system within COSMO is organized in Working Groups which cover the main research and development activities: data assimilation, numerical aspects, upper air physical aspects, soil and surface physics aspects, interpretation and applications, verification and case studies, reference version and implementation and predictability and ensemble methods. In 2005, the COSMO Steering Committee decided to define *Priority Projects* with the goal to focus the scientific activities of the COSMO community on some few key issues and support the permanent improvement of the model. For contacting the Working Group Coordinators or members of the Working Groups or Priority Projects, please refer to the COSMO web-site.

The COSMO meteorological services are not equipped to provide extensive support to external users of the model. If technical problems occur with the installation of the model system or with basic questions how to run the model, questions could be directed via email to `cosmo-support@cosmo-model.org`. If further problems occur, please contact the members of an appropriate Working Group. We try to assist you as well as possible.

The authors of this document recognize that typographical and other errors as well as discrepancies in the code and deficiencies regarding the completeness may be present, and your assistance in correcting them is appreciated. All comments and suggestions for improvement or corrections of the documentation and the model code are welcome and may be directed to the authors.

1.2 Basic Model Design and Features

The nonhydrostatic fully compressible COSMO-Model has been developed to meet high-resolution regional forecast requirements of weather services and to provide a flexible tool for various scientific applications on a broad range of spatial scales. When starting with the development of the COSMO-Model, many NWP-models operated on hydrostatic scales of motion with grid spacings down to about 10 km and thus lacked the spatial resolution required to explicitly capture small-scale severe weather events. The COSMO-Model has been designed for meso-β and meso-γ scales where nonhydrostatic effects begin to play an essential role in the evolution of atmospheric flows.

By employing 1 to 3 km grid spacing for operational forecasts over a large domain, it is expected that deep moist convection and the associated feedback mechanisms to the larger scales of motion can be explicitly resolved. Meso-γ scale NWP-models thus have the principle potential to overcome the shortcomings resulting from the application of parameterized convection in current coarse-grid hydrostatic models. In addition, the impact of topography on the organization of penetrative convection by, e.g. channeling effects, is represented much more realistically in high resolution nonhydrostatic forecast models.

In the beginning, the operational application of the model within COSMO were mainly on the meso-β scale using a grid spacing of 7 km. The key issue was an accurate numerical prediction of near-surface weather conditions, focusing on clouds, fog, frontal precipitation, and orographically and thermally forced local wind systems. Since April 2007, a meso-γ scale version is running operationally at DWD by employing a grid spacing of 2.8 km. Applications with similar resolutions are now run by most COSMO partners. We expect that this will allow for a direct simulation of severe weather events triggered by deep moist convection, such as supercell thunderstorms, intense mesoscale convective complexes, prefrontal squall-line storms and heavy snowfall from wintertime mesocyclones.

The requirements for the data assimilation system for the operational COSMO-Model are mainly determined by the very high resolution of the model and by the task to employ it also for nowcasting purposes in the future. Hence, detailed high-resolution analyses have to be able to be produced frequently and quickly, and this requires a thorough use of asynoptic and high-frequency observations such as aircraft data and remote sensing data. Since both 3-dimensional and 4-dimensional variational methods tend to be less appropriate for this purpose, a scheme based on the observation nudging technique has been chosen for data assimilation from the beginning of the development. But in March 2017 the nudging scheme has been replaced by a new, more modern, ensemble-based method, called KENDA: Km-scale ENsemble Data Assimilation. Note, that KENDA, unlike the nudging scheme, is not available within the source code of the COSMO-Model.

Besides the operational application, the COSMO-Model provides a nonhydrostatic modeling framework for various scientific and technical purposes. Examples are applications of the model to large-eddy simulations, cloud resolving simulations, studies on orographic flow systems and storm dynamics, development and validation of large-scale parameterization schemes by fine-scale modeling, and tests of computational strategies and numerical techniques. For these types of studies, the model should be applicable to both real data cases and artificial cases using idealized test data. Moreover, the model has been adapted by other communities for applications in climate mode (CCLM) and / or running an online coupled module for aerosols and reactive trace gases (ART).

Such a wide range of applications imposes a number of requirements for the physical, numerical and technical design of the model. The main design requirements are:

(i) use of nonhydrostatic, compressible dynamical equations to avoid restrictions on the spatial scales and the domain size, and application of an efficient numerical method of solution;

(ii) provision of a comprehensive physics package to cover adequately the spatial scales of application, and provision of high-resolution data sets for all external parameters required by the parameterization schemes;

(iii) flexible choice of initial and boundary conditions to accommodate both real data cases and idealized initial states, and use of a mesh-refinement technique to focus on regions of interest and to handle multi-scale phenomena;

(iv) use of a high-resolution analysis method capable of assimilating high-frequency asynoptic data and remote sensing data;

(v) use of pure Fortran constructs to render the code portable among a variety of computer systems, and application of the standard MPI-software for message passing on distributed memory machines to accommodate broad classes of parallel computers.

The development of the COSMO-Model was organized along these basic guidelines. However, not all of the requirements are fully implemented, and development work and further improvement is an ongoing task. The main features and characteristics of the present release are summarized below.

COSMO-ICON Physics

In the last months, several physical packages have been unified with their counterpart in ICON, to reduce the maintenance work for having two different versions of one parameterization. We refer to this developments as the *COSMO-ICON Physics*.

A major technical change to implement this unification was, to use the ICON data structure for the variables in the physics. This structure does not reflect a horizontal field with two dimensions, but collects the grid points in a vector (or a block). This is in contrast to the COSMO-Model, which uses the `(i,j)`-structure for horizontal fields.

This *blocked data structure* is explained in more detail in Appendix A of Part II, the Physical Parameterizations.

Not all options for the parameterizations have been ported to the blocked data structure. More details are given below in the *Physical Parameterizations*.

Dynamics

- **Model Equations** – Nonhydrostatic, full compressible hydro-thermodynamical equations in advection form. Subtraction of a hydrostatic base state at rest.
- **Prognostic Variables** – Horizontal and vertical Cartesian wind components, pressure perturbation, temperature, specific humidity, cloud water content. Optionally: cloud ice content, turbulent kinetic energy, specific water content of rain, snow and graupel.
- **Diagnostic Variables** – Total air density, precipitation fluxes of rain and snow.
- **Coordinate System** – Generalized terrain-following height coordinate with rotated geographical coordinates and user defined grid stretching in the vertical. Options for (i) base-state pressure based height coordinate, (ii) Gal-Chen height coordinate and (iii) exponential height coordinate (SLEVE) according to Schär et al. (2002).

Numerics

- **Grid Structure** – Arakawa C-grid, Lorenz vertical grid staggering.
- **Spatial Discretization** – Second-order finite differences. For the two time-level scheme also 1st and 3rd to 6th order horizontal advection (default: 5th order). Option for explicit higher order vertical advection.
- **Time Integration** – Two time-level 2nd and 3rd order Runge-Kutta split-explicit scheme after Wicker and Skamarock (2002) and a TVD-variant (Total Variation Diminishing) of a 3rd order Runge-Kutta split-explicit scheme. Option for a second-order leapfrog HE-VI (horizontally explicit, vertically implicit) time-split integration scheme, including extensions proposed by Skamarock and Klemp (1992). Option for a three time-level 3-d semi-implicit scheme (Thomas et al. (2000)) based on the leapfrog scheme.
- **Numerical Smoothing** – 4th-order linear horizontal diffusion with option for a monotonic version including an orographic limiter. Rayleigh damping in upper layers. 2-d divergence damping and off-centering in the vertical in split time steps.

Initial and Boundary Conditions

- **Initial Conditions** – Interpolated initial data from various coarse-grid driving models (ICON (and former GME), ECMWF, COSMO-Model) or from the continuous data assimilation stream (see below). Option for user-specified idealized initial fields.
- **Lateral Boundary Conditions** – 1-way nesting by Davies-type lateral boundary formulation. Data from several coarse-grid models can be processed (ICON (and former GME), IFS, COSMO-Model). Option for periodic boundary conditions.
- **Top Boundary Conditions** – Options for rigid lid condition and Rayleigh damping layer.
- **Initialization** – Digital-filter initialization of unbalanced initial states (Lynch et al. (1997)) with options for adiabatic and diabatic initialization.

Physical Parameterizations

- **Subgrid-Scale Turbulence** – Prognostic turbulent kinetic energy closure at level 2.5 including effects from subgrid-scale condensation and from thermal circulations. Option for a diagnostic second order K-closure of hierarchy level 2 for vertical turbulent fluxes (not ported to the blocked data structure). Option for calculation of horizontal turbulent diffusion in terrain following coordinates (3D Turbulence; tested in artificial setups).
- **Surface Layer Parameterization** – A Surface layer scheme (based on turbulent kinetic energy) including a laminar-turbulent roughness layer. Option for a stability-dependent drag-law formulation of momentum, heat and moisture fluxes according to similarity theory This option has not been ported to the blocked data structure. (Louis (1979)).

- **Grid-Scale Clouds and Precipitation** – Cloud water condensation and evaporation by saturation adjustment. Precipitation formation by a bulk microphysics parameterization including water vapour, cloud water, cloud ice, rain and snow with 3D transport for the precipitating phases. Option for a new bulk scheme including graupel. Option for a simpler column equilibrium scheme.
- **Subgrid-Scale Clouds** – Subgrid-scale cloudiness is interpreted by an empirical function depending on relative humidity and height. A corresponding cloud water content is also interpreted. Option for a statistical subgrid-scale cloud diagnostic for turbulence. This option has not been ported to the blocked data structure.
- **Moist Convection** – Tiedtke (1989) mass-flux convection scheme with equilibrium closure based on moisture convergence. Option for the current IFS Tiedtke-Bechtold convection scheme.
- **Shallow Convection** – Reduced Tiedtke scheme for shallow convection only.
- **Radiation** – δ two-stream radiation scheme after Ritter and Geleyn (1992) short and longwave fluxes (employing eight spectral intervals); full cloud-radiation feedback.
- **Soil Model** – Multi-layer version of the former two-layer soil model after Jacobsen and Heise (1982) based on the direct numerical solution of the heat conduction equation. Snow and interception storage are included.
- **Fresh-Water Lake Parameterization** – Two-layer bulk model after Mironov (2008) to predict the vertical temperature structure and mixing conditions in fresh-water lakes of various depths.
- **Sea-Ice Scheme** – Parameterization of thermodynamic processes (without rheology) after Mironov and Ritter (2004). The scheme basically computes the energy balance at the iceâĂŹs surface, using one layer of sea ice.
- **Terrain and Surface Data** – All external parameters of the model are available at various resolutions for a pre-defined region covering Europe. For other regions or grid-spacings, the external parameter file can be generated by a preprocessor program using high-resolution global data sets.

Data Assimilation

- **Former Method** – Continuous four-dimensional data assimilation based on observation nudging (Schraff (1996), Schraff (1997)), with lateral spreading of upper-air observation increments along horizontal surfaces. Explicit balancing by a hydrostatic temperature correction for surface pressure updates, a geostrophic wind correction, and a hydrostatic upper-air pressure correction.
- **Actual Method** – Ensemble data assimilation based on the LETKF (Local Ensemble Transform Kalman Filter) (Schraff et al. (2016))
- **Assimilated Atmospheric Observations** – Radiosonde (wind, temperature, humidity), aircraft (wind, temperature), wind profiler (wind), and surface-level data (SYNOP, SHIP, BUOY: pressure, wind, humidity). Optionally RASS (temperature), radar VAD wind, and ground-based GPS (integrated water vapour) data. Surface-level temperature is used for the soil moisture analysis only.
- **Radar derived rain rates** – Assimilation of near surface rain rates based on latent heat nudging (Stephan et al. (2008)). It locally adjusts the three-dimensional thermodynamical field of the model in such a way that the modelled precipitation rates should resemble the observed ones.
- **Surface and Soil Fields** – Additional two-dimensional intermittent analysis:
  - **Soil Moisture Analysis** – Daily adjustment of soil moisture by a variational method (Hess (2001)) in order to improve 2-m temperature forecasts; use of a Kalman-Filter-like background weighting.

  - **Sea Surface Temperature Analysis** – Daily Cressman-type correction, and blending with global analysis. Use of external sea ice cover analysis.
  - **Snow Depth Analysis** – 6-hourly analysis by weighted averaging of snow depth observations, and use of snowfall data and predicted snow depth.

Code and Parallelization

- **Code Structure** – Modular code structure using standard Fortran constructs.
- **Parallelization** – The parallelization is done by horizontal domain decomposition using a soft-coded gridline halo (2 lines for Leapfrog, 3 for the Runge-Kutta scheme). The *Message Passing Interface* software (MPI) is used for message passing on distributed memory machines.
- **Compilation of the Code** – For all programs a Makefile is provided for the compilation which is invoked by the Unix *make* command. Two files are belonging to the Makefile: `ObjFiles` is a list of files that have to be compiled and `ObjDependencies` contains all file dependencies. In addition it reads the file `Fopts`, which has to be adapted by the user to specify the compiler, compiler options and necessary libraries to link.
- **Portability** – The model can be easily ported to various platforms; current applications are on conventional scalar machines (UNIX workstations, LINUX and Windows-NT PCs), on vector computers (NEC SX series) and MPP machines (CRAY, IBM, SGI and others).
- **Model Geometry** – 3-d, 2-d and 1-d model configurations. Metrical terms can be adjusted to represent tangential Cartesian geometry with constant or zero Coriolis parameter.

1.3 Single Precision Version

From the beginning of the development, the COSMO-Model had been designed to be able to run in both precisions: single and double precision. Therefore, the real variables are all defined using a `KIND`-parameter, named `wp` (means: working precision) in the module `kind_parameters.f90` (earlier, this `KIND`-parameter was named `ireals`). Other `KIND`-parameters are `sp` (for single precision) and `dp` (for double precision). Before compiling the model, the user has to decide whether `wp` will be set to `sp` or to `dp`. This can be done with the compiler pragma `-DSINGLEPRECISION`. If this pragma is set, single precision will be used, otherwise double precision.

But in the first years of the COSMO-Model, only the double precision version was developed and tested, nobody ever used or tried a single precision run.

But single precision programs run faster on computers, because of less memory traffic, therefore MeteoSwiss tested to run the COSMO-Model also in single precision. Which did not work in the first instance. Some effort had to be put in adapting the model to work for single precision.

The main changes are:

- Epsilons, which are used in comparisons or to make divisions safe, are adapted to work in both precisions. Variables `repsilon` and `rprecision` have been introduced in module `data_constants.f90`.
- New variables `imp_single` and `imp_double` are added to specify an appropriate MPI data type.

- To avoid automatic conversions by the compiler, all (!) real constants (as `2.0`, `0.5`, etc.) are now written with the kind parameter as suffix: `2.0_wp`, `0.5_wp`, etc.). Further developments should follow this rule!
- The pragma `SINGLEPRECISION` is now used to choose single precision for the COSMO-Model during compilation. If it is not set, double precision is used.

It turned out, that the radiation cannot be run in single precision (at least the routines `coe_th`, `inv_th`, `coe_so`, `inv_so`). Therefore it was decided to run the subroutine `fesft` and all routines called below in double precision. The necessary variables are defined with the `KIND`-parameter `dp`.

1.4 Organization of the Documentation

For the documentation of the model we follow closely the *European Standards for Writing and Documenting Exchangeable Fortran 90-Code.* These standards provide a framework for the use of Fortran-90 in European meteorological organizations and weather services and thereby facilitate the exchange of code between these centres. According to these standards, the model documentation is split into two categories: external documentation (outside the code) and internal documentation (inside the code). The model provides extensive documentation within the codes of the subroutines. This is in form of procedure headers, section comments and other comments. The external documentation is split into seven parts, which are listed in Table 1.2.

Table 1.2: COSMO Documentation: A Description of the Nonhydrostatic Regional COSMO-Model

Part I:	Dynamics and Numerics
Part II:	Physical Parameterization
Part III:	Data Assimilation
Part IV:	Special Components and Implementation Details
Part V:	Preprocessing: Initial and Boundary Data for the COSMO-Model
Part VI:	Model Output and Data Formats for I/O
Part VII:	User's Guide

Parts I - III form the scientific documentation, which provides information about the theoretical and numerical formulation of the model, the parameterization of physical processes and the four-dimensional data assimilation. The scientific documentation is independent of (i.e. does not refer to) the code itself. Part IV will describe the particular implementation of the methods and algorithms as presented in Parts I - III, including information on the basic code design and on the strategy for parallelization using the MPI library for message passing on distributed memory machines (not available yet). The generation of initial and boundary conditions from coarse grid driving models is described in Part V. This part is a description of the interpolation procedures and algorithms used (not yet complete) as well as a User's Guide for the interpolation program INT2LM. In Part VI we give a description of the data formats, which can be used in the COSMO-Model, and describe the output from the model and from data assimilation. Finally, the User's Guide of the COSMO-Model provides information on code access and how to install, compile, configure and run the model. The User's Guide contains also a detailed description of various control parameters in the model input file (in NAMELIST format) which allow for a flexible model set-up for various applications. All parts of the documentation are available at the COSMO web-site (http://www.cosmo-model.org/content/model/documentation/core/default.htm).

Section 2

Introduction

This part of the documentation for the COSMO-Model is the description of the interpolation program INT2LM, which performs the interpolation from coarse grid model data to initial and/or boundary data for the COSMO-Model. “LM” (LokalModell) was the former name of the COSMO-Model. The following coarse grid models are possible (at the moment):

- GME: the global DWD grid point model on a icosahedral grid.
- ICON: the new global DWD grid point model (also on an icosahedral grid), which has replaced GME in February 2015.
- IFS: the global ECMWF spectral model.
- COSMO-Model: the COSMO-Model can be nested into itself.

It is also possible to process the data from other climate models (like ECHAM), but other *pre-pre-processors* are needed then. These *pre-pre-processors* are available from the CLM-Community, which operates the “CLimate Mode of the COSMO-Model”.

Originally, INT2LM has been a joint development within COSMO and originates from the former GME2LM. The climate mode has been added by members of the CLM-Community. The development tasks were distributed as follows:

- DWD: parallel framework of the program; GME→COSMO, ICON→COSMO
- ARPA-SIM: IFS→COSMO
- MeteoSwiss, DWD: COSMO→COSMO
- CLM-Community: climate mode and processing data from other climate models

This documentation is not yet complete. The scientific documentation of the interpolation algorithms used has never been written. Nevertheless, it provides a User Guide of how to install the program and how to run it. Therefore it serves as a complete reference for all the NAMELIST groups and variables.

Section 3

The Interpolation Procedures

to be completed

Section 4

Installation of the INT2LM

This chapter explains the steps necessary to compile and run the interpolation program. Section 4.1 lists the external libraries that are necessary to run the program and what can be done, if these libraries are not available. The next sections give detailed information on how to prepare, compile, link and run the INT2LM.

4.1 External Libraries for the INT2LM

INT2LM uses external libraries for data I/O. Usage of most of these libraries can be controlled by conditional compilation. To handle this, the C preprocessor (cpp) must be called. Most Fortran compilers activate the C preprocessor for files ending with a capital `F` in the suffix: `.F` or `.F90`. INT2LM does not use capital letters in the suffix, therefore a special compiler option has to be set, to activate this preprocessor. Take a look to the manual of your compiler to find out about this option.

4.1.1 `libgrib1.a`:

The original implementation of INT2LM and the COSMO-Model used GRIB (Grided Binary), Version 1, as standard format. Coding and decoding of GRIB1 records can be done with the DWD GRIB1 library, the `libgrib1.a`. This library also contains C-routines to write data to and read it from disk. The Grib library is available from DWD and is provided together with the source code for the COSMO-Model. A short guide for the installation is included in the tar-file of the Grib library.

With GRIB1, DWD used a Grib file format, where all records are starting and ending with additional bytes, the so-called *controlwords*. To process these controlwords properly, you have to set the environment variable

```
export LIBDWD_FORCE_CONTROLWORDS=1
```

Usage of the DWD GRIB1 library can be controlled by conditional compilation and setting the macro `GRIBDWD`. If this macro is not set during compilation, the parts of the source code that do use `libgrib1.a` calls are not compiled and the library will not be linked to the binary.

4.1.2 libgrib_api.a, libgrib_api_f90.a or libeccodes.a, libeccodes_f90.a:

Since INT2LM Version 1.14, another grib library can be used to read grib data. This is the `grib_api` (Application Programmer's Interface) from ECMWF. With this library it is possible to read and write also GRIB2 data (i.e. GRIB, Version 2). Since some time, `grib_api` is inclued in the `eccodes` framework, which also contains a `bufr`-library. From Version 2.05 on INT2LM can also work with `eccodes` (COSMO-Model from Version 5.05 on). Source codes for both libraries together with installation instructions are available from the ECMWF web pages:
`http://www.ecmwf.int/en/computing/software`.

All following explanations are now for `eccodes`, but the same holds also for `grib_api`.

To use `eccodes` properly, two sets of definition files are necessary for running the programs: one contains all original definitions from ECMWF (`definitions`), the other set is compiled and maintained by DWD and must be used for the INT2LM and the COSMO-Model (`definitions.edzw`). Also, some special sample files are needed by both programs. Both sets of definition files and the necessary samples, which have to correspond to the `eccodes` version used, are distributed together with the INT2LM by DWD.

Please read more on `eccodes` and its usage in the Documentation, Part VI, *Model Output and Data Formats for I/O*.

Usage of the `eccodes` library can be controlled by conditional compilation and setting the macro `GRIBAPI`. If this macro is not set during compilation, the parts of the source code that do use `eccodes` calls are not compiled and the library will not be linked to the binary.

4.1.3 libnetcdf.a:

Since Version 1.7, input and output of data can also be done in the NetCDF format (Network Common Data Format). Using NetCDF requires an external library `libnetcdf.a`. The source code of this library can be downloaded from `http://www.unidata.ucar.edu` .

Usage of the NetCDF library can be controlled by conditional compilation and setting the macro `NETCDF`. If this macro is not set during compilation, the parts of the source code that do use NetCDF calls are not compiled and the library will not be linked to the binary.

NEW: Since INT2LM 2.1 the processing of ICON2LM also needs the NetCDF library to read the ICON grid files and external parameters. If you want to run ICON2LM, you have to compile with -DNETCDF.

4.1.4 libcsobank.a, libsupplement.a

Before COSMO-Model 4.25 and INT2LM 2.1

The COSMO-Model and INT2LM use a tool for parallel asynchronous GRIB I/O from or to files or a data base system (only for Grib). The routines for that tool are grouped together in a module `mpe_io.f90`. In the VCS of DWD, `mpe_io.f90` is provided as an external module, hence it is not in the source code of the model library. `mpe_io.f90` uses the two libraries `libcsobank.a` and `libsupplement.a`.

For users outside DWD, `mpe_io.f90` has been included in the source code of the COSMO-Model and also in the INT2LM. To satisfy the calls from `mpe_io` to the data base system, an additional file `dummy_db.f90` is provided.

- NEW:
  Since COSMO-Model Version 4.25 and INT2LM Version 2.1, `mpe_io.f90` has been replaced by a modified version `mpe_io2.f90`, which does not support the database access any more. Therefore the usage of these external libraries is now obsolete.

4.2 Preparing the Code

You have got a tar-file `int2lm_yymmdd_x.y`, where `yymmdd` describes the date in the form "Year-Month-Day" and `x.y` gives the version number. By de-taring, a directory is created with the following contents:

`DOCS`	Contains a short documentation of the changes in version `x`.
`edid`	Script to edit files in `src` and store them in `work`.
`Fopts`	Definition of the compiler options and also directories of libraries.
`LOCAL`	Contains several examples of `Fopts`-files for different computers.
`Makefile`	For compiling and linking the programs.
`RUNSCRIPTS`	Several examples of scripts (`runxxx2yyy`) to set the `NAMELIST` values for an interpolation from model / configuration xxx to yyy and start the program.
`src`	Subdirectory for the source code.
`obj`	Subdirectory where the object files are written.
`ObjDependencies`	Definition of the dependencies between the different source files.
`Objfiles`	Definition of the object files.
`work`	Subdirectory for intermediate files.

The directories `./obj` and `./work` are empty and can therefore get lost by the tar-process. If so, you have to create them again. In `edid` you have to adapt the pathnames if you want to work with it.

4.3 Compiling and Linking

You have to choose the options for compiling the code in the file `Fopts`. See the User Guide of your computer system for necessary and/or desired options. In particular you have to set the macros to select external libraries. Before linking check that all necessary external libraries (see 4.1) are available.

All other input variables for the program can be determined before running the program with the `NAMELIST`-input (see Chapter 7).

The INT2LM is parallelized for distributed memory parallel computers using the domain decomposition technique and explicit message passing with the Message Passing Interface (MPI). Thus it can run on parallel platforms but also on sequential platforms where MPI is not available. For this purpose an additional module `dummy_mpi.f90`, together with a file `mpif.h`, are provided in the directory `LOCAL`. Place these files in the `src`-directory before compiling and linking. Then you can only create `seq`uential binaries (see below).

You can invoke a `make`-run by typing `make` *entry*. For INT2LM only the two entries for a parallel or for a sequential binary are available:

`parexe` A binary for parallel execution is created.

`seqexe` A binary for sequential execution is created.

4.4 Running the Code

To run the code, an ASCII-file `INPUT` has to be provided that contains values for the `NAMELIST` variables. The form of this `INPUT` file is described in Chapters 6 and 7. This file is created by the provided run-scripts.

The run-scripts also contain the command to invoke the binary, which you have to adapt. While a sequential binary usually can be invoked just by typing `<name_of_binary>`, a parallel binary must be invoked by a special tool. Often this tool is `mpirun`, but also other tools are possible. See the manual for your system on how to invoke the binary created in the last step.

Section 5

Necessary Initial and Boundary Data for the COSMO-Model

This chapter lists all initial and boundary data which are necessary to run the COSMO-Model. Some of the data depend on special namelist settings in the COSMO-Model and the INT2LM, resp. This will be explained in detail.

5.1 Initial Data for the COSMO-Model

The data necessary to start the COSMO-Model can be divided into several groups:

5.1.1 External parameters directly given to the COSMO-Model

The COSMO-Model needs information about the lower boundary of the domain as for example the height of the orography. The fields in Tab. 5.1 are read by INT2LM from the external parameter file and are directly written to the COSMO initial file, usually without further processing. Only the height of the orography can optionally be modified by a filtering procedure. Some of these fields are only necessary, if certain components of the COSMO-Model are activated. Therefore, the usage of these fields can be controlled by Namelist switches, which are the same in INT2LM and the COSMO-Model. These are given in the third column of the table. Some remarks to the fields / components are given below.

All these fields have to be provided by an *external parameter* file (see also Section 6.2), which is produced for a certain region and resolution.

Note that older external parameter files might not contain all necessary fields. Please update your files in this case.

Name	Description	Namelist	Remarks
HSURF FIS SSO_STDH FR_LAND SOILTYP Z0	Height of surface topography [m] (alternatively) Geopotential of surface standard deviation of subgrid scale orography [m] Fraction of land in the grid cell [0-1] Soil type of the land (keys 0-9) Roughness length		1)
SSO_GAMMA SSO_THETA SSO_SIGMA	anisotropy of the orography [-] angle between the principal axis of orography and east [rad] mean slope of subgrid scale orography [-]	lsso	2)
SKYVIEW SLO_ASP SLO_ANG HORIZON	sky view [1] slope aspect [rad] slope angle [rad] horizon array (splitted in nhori sectors)	lradtopo	3)
FR_LAKE DEPTH_LK	lake fraction in a grid element [0,1] lake depth	llake	4)
RSMIN	minimum stomata resistance of plants	lstomata	5)
EMIS_RAD	thermal radiative surface emissivity	lemiss	6)
FOR_E FOR_D	ground fraction covered by evergreen forest ground fraction covered by deciduous forest	lforest	7)
SKC	skin conductivity	lskinc	8)
S_ORO	slope of orography	l_s_oro	9)
FR_PAVED AHF	Paved fraction of surface (possible alternative name: ISA) Yearly average of anthropogenic heat flux	lterra_urb	10)

Table 5.1: Constant External Parameters for the COSMO-Model

1: Mandatory fields

These constant external parameters are needed in any case to start a simulation with the COSMO-Model. Note that either HSURF or FIS is needed, but not necessarily both of them.

SSO_STDH is needed in any case now for computations in the turbulence scheme since Version 5.04e. Before it was only used for the subgrid scale orography scheme.

2: Subgrid scale orography

The sub-grid scale orography (SSO) scheme by Lott and Miller (1997) has been implemented in the COSMO-Model (from Version 4.5 on). The scheme deals explicitly with a low-level flow which is blocked when the sub-grid scale orography is sufficiently high. For this blocked flow separation occurs at the mountain flanks, resulting in a form drag. The upper part of the low-level flow is lead over the orography, while generating gravity waves.

Note: If also the switch `idiag_snowfrac > 0` is chosen, the field `SSO_SIGMA` is also used, even if `lsso = .FALSE.`.

3: Topographical corrections in the radiation scheme

Using these external parameters, topographical corrections for radiation calculations can be computed. The field `HORIZON` is treated as three-dimensional array with `nhori` levels. `nhori` is also read as a namelist parameter.

4: Treatment of fresh water lakes

With the FLake-Model, which was introduced in COSMO-Model Version 3.18, the surface temperature of lakes can be computed. To do that, more fields than these two external parameters are necessary, which e.g. are prognostic values of the temperatures in the lake. These fields should be provided by a data assimilation. But it is possible to initialize all prognostic lake variables for a cold start using the switch `llake_coldstart`. More details on how to use this switch are given in Section 9.1.

5: Minimum stomata resistance of plants

Up to Version 4.10 the multi-layer soil model of the COSMO-Model used a constant minimum value of stomatal resistance for plants. Now an external map can be read, that provides values for every grid point.

6: Thermal radiative surface emissivity

Up to Version 4.10 a constant value was used for the thermal radiative surface emissivity. Now an external map can be read, that provides values for every grid point.

7: Ground fraction covered by forests

The ground fraction covered by evergreen and deciduous forest, resp., can be used in the radiation scheme to determine the effect of snow covered forests on solar snow albedo.

8: Skin conductivity

For the skin temperature formulation added in COSMO-Model 5.06a, an additional external parameter field, the skin conductivity (`SKC`) is needed.

9: Slope of orography

For the modifications regarding the ground water runoff in the soil model TERRA (by Linda Schlemmer, to be implemented into COSMO-Model Version 5.06x), an additional external parameter field, the slope of orography (`S_ORO`) is needed.

10: Parameters for the urban-canopy land-surface scheme TERRA_URB

Note: Usage of TERRA_URB still is in an experimental phase. INT2LM can provide these two external parameters, but the package itself is not yet implemented in an official COSMO-Model version. Processing of the additional fields necessary for TERRA_URB is controlled by the namelist variable `lterra_urb=.TRUE.`.

5.1.2 External parameters computed by INT2LM

Another group of external parameters is not necessarily constant, they can vary from day to day. In NWP simulations, values for the day, when the forecast starts, are calculated by INT2LM and then hold constant for the duration of the simulation (few days). In climate applications, which are running for years, these fields are updated with the boundary data sets.

For some of these fields, given in Tab. 5.2 several ways for initialization exist, which can be chosen by namelist switches. To compute these fields, INT2LM has to read additional fields from the external parameter file. Which fields are needed in which case is described in the remarks below.

Name	Description	Namelist	Remarks
PLCOV	plant cover	itype_ndvi	1)
LAI	leaf area index	itype_ndvi	
Z0	roughness length	itype_ndvi=2	
ROOTDP	root depth	itype_rootdp	
VIO3	total vertically integrated ozone content		2)
HMO3	height of maximum ozone concentration		
AER_SO4	Tegen aerosol type sulfate drops	itype_aerosol=2/5	3)
AER_DUST	Tegen aerosol type mineral dust		
AER_ORG	Tegen aerosol type organic		
AER_BC	Tegen aerosol type black carbon		
AER_SS	Tegen aerosol type sea salt		
TAU_KINNE	Kinne aerosol optical thickness	itype_aerosol=3	3)
SSA_KINNE	Kinne aerosol single scattering albedo		
GKINNE	Kinne aerosol asymmetry parameter		
ALB_DRY	albedo of dry soil	itype_albedo=2	4)
ALB_SAT	albedo of saturated soil		

Name	Description	Namelist	Remarks
`ALB_DIF`	solar diffuse surface albedo	`itype_albedo=3`	4)

Table 5.2: Varying External Parameters for the COSMO-Model

1: Plant characteristics

There are several options to compute the plant characteristics. They are controlled by the namelist parameters `itype_ndvi` and `itype_rootdp`, resp.

`itype_ndvi` works on `PLCOV`, `LAI` and `Z0`. Possible values are:

0: Data sets for vegetation and for rest are read from the external parameter file for plant cover and the leaf area index The actual values for a special day are computed by producing a sinus-type annual cycle.
Necessary fields from the external parameter file:

`PLCOV_MX`	plant cover data set for vegetation time
`PLCOV_MN`	plant cover data set for time of rest
`LAI_MX`	leaf area index data set for vegetation time
`LAI_MN`	leaf area index data set for time of rest

1: Plant cover and leaf area index for the COSMO-Model and for a special day are produced by using only the data set for vegetation and an averaged normalized differential vegetation index (ndvi) ratio. This ndvi ratio is computed as a weighted mean between monthly mean values, which are taken from the external parameter data set for the COSMO-Model.
Necessary fields from the external parameter file:

`PLCOV_MX`	plant cover data set for vegetation time
`LAI_MX`	leaf area index data set for vegetation time
`NDVI_MRAT`	ratio of monthly mean normalized differential vegetation index to annual maximum for 12 months

2: plant cover, leaf area index and roughness length for the COSMO-Model and for a special day are produced by using 12 monthly climatological mean values for plant cover, leaf area index and roughness length. These values are read from the external parameter data set for the COSMO-Model.
Necessary fields from the external parameter file:

`PLCOV12`	12 monthly climatological mean values for plant cover
`LAI12`	12 monthly climatological mean values for leaf area index
`Z012`	12 monthly climatological mean values for roughtness length

For the root depth, just one dataset is given in the external parameter file. The options for computing actual values are given by `itype_rootdp`:

0: input from external parameter for the COSMO-Model is taken and modified with an annual cycle.

1: input from external parameter for the COSMO-Model is taken as is but with a minimal value of 0.12.

2: input from external parameter for the COSMO-Model is taken and modified with an annual cycle. In addition, the values are adapted to ECOCLIMAP niveau.

3: input from external parameter for the COSMO-Model is taken and modified with an annual cycle but without maximum cut off.

4: input from external parameter for the COSMO-Model is taken without any modifications.

The necessary field from the external parameter file is:

`ROOTDP`	root depth

2: Ozone contents

The Ritter-Geleyn radiation scheme implemented in the COSMO-Model needs information about the vertical integrated ozone content (`VIO3`) and the ozone maximum (`HMO3`).

INT2LM computes these fields for a special day in the year and provides them in the initial file for the COSMO-Model.

To be completed: Method??

3: Aerosol characteristics

The default treatment of aerosols in the COSMO-Model is by assuming constant values for aerosols in rural, urban or desert areas and over sea. In the past, several other aerosol climatologies have been added and can be used by the COSMO-Model, if INT2LM provides necessary external parameters.

What type of aerosols should be used for the COSMO-Model is controlled by the namelist parameter `itype_aerosol`:

1: Default: No additional fields for aerosol types are read from the external parameters. The COSMO-Model has to run with constant values for the aerosol distribution on rural areas, urban areas, desert areas or the sea.

2: Since INT2LM Version 1.09, the Tegen aerosol climatology with 12 monthly mean values for aerosol types of sulfate, mineral dust, organic, black carbon and sea salt (see Tegen et al. (1997)) are read by INT2LM and actual values for a special day are computed by a linear interpolation between the corresponding months. The actual values are given to the COSMO-Model.
Necessary fields from the external parameter file:

`AER_SO412`	Tegen (1997) aerosol type sulfate drops
`AER_DUST12`	Tegen (1997) aerosol type mineral dust
`AER_ORG12`	Tegen (1997) aerosol type organic
`AER_BC12`	Tegen (1997) aerosol type black carbon
`AER_SS12`	Tegen (1997) aerosol type sea salt

3: Since INT2LM Version 2.05, three additional fields can be read from the external parameter data set for COSMO describing the Kinne et al. (2013) aerosol climatology, but in version v2, not v1. These fields provide monthly mean values of total aerosol optical thickness, single scattering albedo and asymmetry parameter for the 8 wavelengths of the COSMO radiation scheme as well as for the 550 nm wavelength. INT2LM reads these 12 monthly mean fields and computes the time-interpolated fields for the actual day, which are given to the COSMO-Model.

Note: These parameters are only available in the NetCDF format of the external parameters and their usage still is in an experimental phase (developed in the COSMO Priority Project $T^2(RC)^2$)! They are not usable in an official COSMO-Model version.

Necessary fields from the external parameter file:

`AOT12`	Kinne aerosol optical thickness
`SSA12`	Kinne aerosol single scattering albedo
`ASY12`	Kinne aerosol asymmetry parameter

4: ECMWF provides operational aerosol forecasts from a relatively coarse (about 60 km resolution) deterministic IFS instance. The data are provided as (dry) mass mixing ratios for 11 categories. In IFS, the optical properties of the soluble species (such as sea salt), depend also on relative humidity, which enables to represent the direct radiative effects of haze in the COSMO-model. No external parameters need to be read for this option of aerosol treatment.

Since INT2LM Version 2.05 these fields can be read by INT2LM from the coarse grid data (corresponding initial or boundary data files) and can be interpolated to the COSMO grid.

Note: The implementation of these fields in the COSMO-Model still is in an experimental phase (developed in the COSMO Priority Project $T^2(RC)^2$)! They are not usable in an official COSMO-Model version.

5: In the scope of the PerduS project, daily mineral dust forecasts with ICON-ART are produced (Rieger et al. (2015); Rieger et al. (2017)). ICON-ART provides mass-mixing ratios of three log-normally distributed modes. No external parameters need to be read for this option of aerosol treatment.

Since INT2LM Version 2.05 these fields can be read by INT2LM from the coarse grid data (corresponding initial or boundary data files) and can be interpolated to the COSMO grid.

Note: The implementation of these fields in the COSMO-Model still is in an experimental phase (developed in the COSMO Priority Project $T^2(RC)^2$)! They are not usable in an official COSMO-Model version.

4: Treatment of surface albedo

The default treatment in the COSMO-Model is by assuming that the surface albedo is a function of the soil type. Corresponding settings are done in the COSMO-Model and no action is necessary in INT2LM.

In INT2LM 1.19, additional options have been introduced by the Climate Community and a namelist switch `itype_albedo` has been introduced to select a special treatment.

1: Default: Surface albedo is a function of soiltype.

2: Surface albedo prescribed by external fields for dry and saturated soil: Two additional fields are read from the external parameter file, which are passed to the COSMO-Model:

`ALB_DRY`	albedo of dry soil
`ALB_SAT`	albedo of saturated soil

3: Background albedo is prescribed by external fields (12 monthly mean values): Additional fields for monthly mean values are read from the external parameter file and a time-interpolated field for the solar diffuse surface albedo `ALB_DIF` is computed for a special day.

`ALB_DIF12`	monthly values of solar diffuse surface albedo

4: Vegetation albedo is modified by forest fraction: This is done in the COSMO-Model, no actions are necessary in INT2LM.

5.1.3 Variables for the surface, soil, lakes and sea-ice

Tab. 5.3 lists all the necessary initial data for the surface, the soil, lakes and sea-ice. Some of these data are optional, depending on the chosen configuration. The corresponding namelist switches are listed in the third column. Some remarks for special variables are given below.

Name	Description	Namelist	Remarks
T_S	temperature of the surface		1)
T_SNOW	temperature of snow surface		
W_SNOW	water content of snow		
W_I	water content of interception water		
QV_S	specific water vapor content at the surface		
T_SO	temperature of multi-layer soil levels	lmulti_layer	2)
W_SO	water content of multi-layer soil levels	= .TRUE.	
FRESH_SNOW	indicator for freshness of snow		
RHO_SNOW	prognostic snow density	lprog_rho_snow	
T_M	temperature between upper and medium layer	lmulti_layer	3)
T_CL	temperature between medium and lower layer	= .FALSE.	
WG_1	water content of upper soil layer		
WG_2	water content of medium soil layer		
WG_3 (*)	water content of lower soil layer	nl_soil_lm = 3	
W_CL	water content of climatological soil layer		
FR_LAKE	lake fraction in a grid box	llake	4)
DEPTH_LK	lake depth		
T_MNW_LK	mean temperature of the water column		
T_WML_LK	mixed-layer temperature		
T_BOT_LK	temperature at the water-bottom sediment interface		
C_T_LK	shape factor with respect to the temperature profile in lake thermocline		
H_ML_LK	thickness of the mixed-layer		
T_ICE	temperature of ice / lake surface	llake, lseaice	5)
H_ICE	lake / sea ice thickness		

Table 5.3: Initial Data for Surface, Soil, Lakes, Sea-Ice

1: Necessary surface variables

In an assimilation cycle, these variables are governed by the COSMO-Model, with regular updates by external analyses for `T_SNOW, W_SNOW and W_I` (at 00, 06, 12 and 18 UTC) and for `T_S` (at 00 UTC). The external analysis for `T_S` updates the values only over sea.

If no assimilation cycle is used, these fields are interpolated from the coarse grid model.

2: Necessary soil variables for the multi-layer version of TERRA

In the COSMO-Model, the usage of the soil model is controlled by the namelist variable `lmulti_layer=.TRUE.`.

Note: For the COSMO-Model, this namelist variable has been eliminated in version 5.05. Now only this multi-layer version can be used. For backward compatibility, INT2LM still can produce fields for the old 2 (or 3) layer scheme.

In INT2LM, the corresponding variable is `lmulti_layer_lm=.TRUE.`, which indicates, that the output of INT2LM has to be for the multi-layer soil model. There is also the namelist variable `lmulti_layer_in=.TRUE.`, which indicates, that the input (coarse grid) model also used a multi-layer soil model. This can only be true for the DWD models ICON, GME and COSMO-Model. No other model uses a comparable multi-layer soil model.

For the DWD global models ICON and GME, also the snow density can be read and interpolated. This is activated by setting `lprog_rho_snow=.TRUE.`.

When interpolating coarse grid COSMO to fine grid and in the case where the soil type ice (1, therefore no soil moisture) of the coarse grid covers larger areas as the fine grid, the interpolation of the soil moisture variable `W_SO` can be modified. In this case, the original coarse grid COSMO soil moisture comes from the nearest grid point which is not an ice point and the interpolated fine COSMO grid points have a meaningful moisture different from 0. This procedure could be used in a similar way for the case of rock (SOILTYP 2 which neither has any soil moisture).

But because this procedure has some risks regarding to soil temperature and soil ice, it is not activated by default and cannot be activated by a namelist switch. The user has to activate it by changing the code (in module `src_coarse_interpol.f90`, subroutine `interpol_coarse_special_lm`, Section 2).

3: Necessary soil variables for the old two- or three-layer version of TERRA

For the old version of TERRA, different fields are necessary. These fields will be read by the COSMO-Model, if `lmulti_layer=.FALSE.` is set (only possible with versions older than 5.05). In the INT2LM, `lmulti_layer_lm=.FALSE.` has to be used correspondingly. The field `WG_3` only is necessary, if `nlgw_ini=3` is set in the namelist input for the COSMO-Model. In INT2LM you have to set `nl_soil_lm = 3`.

If an assimilation cycle is used, the soil fields are governed by the COSMO-Model (no matter, which soil model is used). There is the possibility to run a *Soil Moisture Analysis*, which is not really an external analysis, but adapts the soil moisture in the upper level in a way, that the temperature forecast is adjusted to the observations.

If no assimilation cycle is used, the fields are interpolated from the coarse grid model.

4: Necessary variables for the FLake Model

If the FLake-Model is used (`llake=.TRUE.`), these fields are needed as initial variables. They should only be governed by the COSMO-Model data assimilation cycle. Nevertheless, it is possible to initialize all prognostic lake variables for a cold start using the switch `llake_coldstart`. More details on how to use this switch are given in Section 9.1.

5: Necessary variables for the Sea Ice Scheme

If the FLake-Model or the sea-ice scheme is chosen, variables for the sea-ice temperature and for the thickness of the ice have to be specified as initial variables.

5.1.4 Atmospheric variables

Tab. 5.4 lists all variables necessary to initialize a COSMO-Model forecast. Some of these variables are optional, depending on the chosen configuration. The corresponding namelist switches are listed in the third column. Some remarks for special variables are given below.

Name	Description	Namelist	Remarks
`U`	zonal wind speed		1)
`V`	meridional wind speed		
`W`	vertical wind speed (defined on half levels)		
`T`	temperature		
`P`	(full) pressure		
`PP` (*)	pressure deviation from a reference pressure		
`QV`	specific water vapour content		
`QC`	specific cloud water content		
`HHL` (*)	height of half levels		
`QI`	specific cloud ice content	`lprog_qi`	2)
`QR`	specific rain content	`lprog_qr_qs`	
`QS`	specific snow content	`lprog_qr_qs`	
`QG`	specific graupel content	`lprog_qg`	
`NCICE`	number concentration rain water	`lprog_qi`	3)
`NCCLOUD`	number concentration cloud water	`lprog_qn_crsg`	
`NCRAIN`	number concentration rain	`lprog_qn_crsg`	
`NCSNOW`	number concentration snow	`lprog_qn_crsg`	
`NCGRAUPEL`	number concentration graupel	`lprog_qn_crsg`	

Table 5.4: Initial Atmospheric Variables

1: Basic atmospheric variables

These are the basic fields necessary for any COSMO run.

- P or PP: Some years ago, when only using GRIB 1, the COSMO-Model definitely needed the pressure deviation PP as initial field. The GRIB 1 meta data contained all necessary variables to compute the height of the half levels HHL, the reference pressure P0 and the full pressure P.
  This is not the case any more for GRIB 2, therefore in this case, the full pressure P is needed as initial field.
  Newer versions of the COSMO-Model now accept both fields.

- HHL: If initial data is in GRIB 2, the height of the half levels cannot be computed any more, but a field has to be provided with the initial data.

2: Additional humidity variables

Since the start of the development of the COSMO-Model, more humidity variables have been added to the set of equation. Since values for these fields are not available in older data sets or from certain coarse grid models, their usage can be controlled by specific namelist settings.

In an assimilation cycle all these fields are governed by the data assimilation. If no assimilation cycle is used, they are interpolated from the coarse grid model.

With the namelist variables lana_qi, lana_qr_qs and lana_qg from the COSMO-Model, it can be chosen, whether the fields from a data assimilation cycle (.TRUE.) or interpolated fields (.FALSE.) should be used.

3: Additional variables for the 2-moment scheme

These variables are used in the COSMO 1-moment cloud microphysical schemes. A nesting of runs with the alternative 2-moment scheme (additional number concentrations), which is not yet part of the official COSMO-Code but available as an additional module, is also possible. For this, the following variables are interpolated if they are present in the coarse grid input files and if the corresponding switches in INT2LM namelists (as given in the third column) are specified.

Reading these fields in the COSMO-Model is controlled by the namelist variables lana_qx, as for the humidity variables: lana_qi for NCICE, lana_qr_qs for NCRAIN and NCSNOW, and lana_qg for NCGRAUPEL.

Note that it is possible to run the COSMO-Model with 2-moment scheme without having the number concentrations in the analyis-and boundary files. In this case, the initial number concentrations in the COSMO-Model are guessed from reasonable assumptions on the particle size distribution from the initial specific hydrometeor contents. A certain spinup process of hydrometeor conversion rates and fallspeeds is the consequence.

The hail category in the 2-moment scheme is usually so transient and short-lived that it is not necessary to carry it over in inital- and boundary files.

5.2 Boundary Data for the COSMO-Model

The necessary boundary data for the COSMO-Model are the atmospheric variables and some surface variables. Boundary values have to be provided also for the old two-layer soil model. Some special considerations have to be done for the climate mode.

5.2.1 Soil and surface variables

Tab. 5.5 lists the variables necessary for the soil and surface:

Name	Description	Namelist	Remarks
`T_SNOW`	temperature of snow surface		
`W_SNOW`	water content of snow		
`QV_S`	specific water vapor content at the surface		
`T_M`	temperature between upper and medium layer	`lmulti_layer`	1)
`T_CL`	temperature between medium and lower layer	`= .FALSE.`	
`WG_1`	water content of upper soil layer		
`WG_2`	water content of medium soil layer		
`WG_3` (*)	water content of lower soil layer	`nl_soil_lm = 3`	
`W_CL`	water content of climatological soil layer		

Table 5.5: Boundary Data for Surface and Soil

1: Soil variables for the old version of TERRA

If the old two-layer soil model is used, these variables have to be provided. The field `WG_3` only is necessary, if `nlgw_bd=3` in the namelist input for the COSMO-Model.

Note that the multi-layer version of TERRA does not need such boundary values.

5.2.2 Atmospheric variables

Tab. 5.6 lists all variables necessary as boundary fields for a COSMO-Model forecast. Some of these variables are optional, depending on the chosen configuration. The corresponding namelist switches are listed in the third column. Some remarks for special variables are given below.

Name	Description	Namelist	Remarks
U	zonal wind speed		1)
V	meridional wind speed		
W (*)	vertical wind speed (defined on half levels)		
T	temperature		
P	(full) pressure		
PP (*)	pressure deviation from a reference pressure		
QV	specific water vapour content		
QC	specific cloud water content		
QI	specific cloud ice content	lprog_qi	2)
QR	specific rain content	lprog_qr_qs	
QS	specific snow content	lprog_qr_qs	
QG	specific graupel content	lprog_qg	
NCICE	number concentration rain water	lprog_qi	3)
NCCLOUD	number concentration cloud water	lprog_qn_crsg	
NCRAIN	number concentration rain	lprog_qn_crsg	
NCSNOW	number concentration snow	lprog_qn_crsg	
NCGRAUPEL	number concentration graupel	lprog_qn_crsg	

Table 5.6: Boundary Atmospheric Variables

1: Basic atmospheric variables

- P or PP: Only one of these fields is necessary. See remarks for 5.4.
- W: Depending on the variable lw_freeslip in the namelist input for the COSMO-Model, also the vertical wind speed has to be provided.
  - If lw_freeslip=.TRUE., a free-slip condition is implemented, which does NOT need boundary values.
  - If lw_freeslip=.FALSE., boundary values have to be provided. In INT2LM the namelist variable lvertwind_bd=.TRUE. has to be set in this case.

2: Additional humidity variables

Whether boundary values should be provided for the additional humidity variables is also controlled by the namelist variables lprog_qx. Of course, the coarse model has to provide such fields.

With the namelist variables llb_qi, llb_qr_qs and llb_qg from the COSMO-Model, it can be chosen, whether these fields are contained in the boundary data sets or not.

3: Additional variables for the 2-moment scheme

As for analysis data, the switches `lprog_qx` control the computation for the additional number concentrations of the 2-moment cloud microphysical scheme. Again, these are optional, as for the initial data.

Reading these fields in the COSMO-Model is controlled by the namelist variables `llb_qx`, as for the humidity variables: `llb_qi` for `NCICE`, `llb_qr_qs` for `NCRAIN` and `NCSNOW`, and `llb_qg` for `NCGRAUPEL`.

The 2-moment scheme in the COSMO-Model can also be used without these fields at the boundaries, but at the expense of a certain spinup zone near the inflow boundaries.

5.2.3 Special considerations for the climate mode

If the COSMO-Model is run in climate mode, additional fields are necessary for the boundary updates, depending on the options chosen:

Name	Description	Namelist	Remarks
`T_S`	surface temperature		
`PLCOV`	plant cover		
`LAI`	leaf area index		
`Z0`	roughness length		
`ROOTDP`	root depth		
`VIO3`	total vertically integrated ozone content		
`HMO3`	height of maximum ozone concentration		
`AER_SO4`	Tegen aerosol type sulfate drops	`itype_aerosol=2/5`	
`AER_DUST`	Tegen aerosol type mineral dust		
`AER_ORG`	Tegen aerosol type organic		
`AER_BC`	Tegen aerosol type black carbon		
`AER_SS`	Tegen aerosol type sea salt		
`TAU_KINNE`	Kinne aerosol optical thickness	`itype_aerosol=3`	
`SSA_KINNE`	Kinne aerosol single scattering albedo		
`GKINNE`	Kinne aerosol asymmetry parameter		
`ALB_DIF`	solar diffuse surface albedo	`itype_albedo=3`	

Table 5.7: Additional Boundary Data in Climate Model

Section 6

Input Files for the INT2LM

The INT2LM requires several input files.

- An ASCII-file, called `INPUT`, that contains the namelist variables. The form of this file is described in Section 6.1 The namelist groups, the variables, their meanings and possible values are described in Chapter 7.
- 2 files with external parameters for the COSMO-Model and for the coarse grid model. These files can be in Grib or NetCDF format.
- Files from the coarse grid model to compute the initial and/or boundary values. The name of these files are described in Section 6.6. These files can be in Grib or NetCDF format. If GME is used and the GME-files contain data that were selected by a bitmap, this bitmap must also be provided to INT2LM.

6.1 File for Namelist Input

The INT2LM uses `NAMELIST`-input to specify runtime parameters. The parameters are splitted into the groups

- `CONTRL` – parameters for the model run
- `GRID_IN` – specifying the domain and the size of the coarse grid
- `LMGRID` – specifying the domain and the size of the COSMO-Model grid
- `DATA` – controlling the grib input and output
- `PRICTR` – controlling grid point output

NEW: Since INT2LM 2.1 the namelist group `DATABASE` is not required any more.

The program provides default values for all parameters. To change a default value, an appropriate `NAMELIST` statement has to appear in the ASCII-file `INPUT`. The form of a `NAMELIST` statement depends on the specific platform you are using but is always similar to the following (refer to the Language Reference Manual of your system):

1. The ampersand (&) character, followed immediately by the name of the namelist group.
2. A sequence of zero or more

   `parameter=value,`

   statements.
3. / to terminate the NAMELIST group.

Example:
In the following example new values are set for the parameters in the Namelist group lmgrid:

```
&LMGRID
  startlat_tot = -10.4, startlon_tot = -3.025,
  pollat=32.5,          pollon=-170.0,
  dlat=0.025,           dlon=0.025,
  ielm_tot=72,          jelm_tot=92,
/
```

For a complete reference of all NAMELIST parameters see Chapter 7. An example INPUT-file can be seen in Figure 7.1.

6.2 External Parameters

For both models, the coarse grid input model and the COSMO-Model, external parameters are required to perform the interpolations. The necessary and / or available external parameters for the COSMO-Model are listed in Section 5.1

If the file with the external parameters for the COSMO-Model does not exist, they could be interpolated from the coarse grid model. But this works only for the configuration GME → COSMO. Note, that this is not recommended for practical simulations!

Necessary parameters for the coarse grid model

HSURF (*)	Height of surface topography
FIS (*)	(alternatively) Geopotential of surface
FR_LAND	Fraction of land in the grid cell
SOILTYP (**)	Soil type of the land (keys 0-9)

(*) For ICON as coarse grid model, HSURF is not needed in the external parameter file, but it is read with the HHL-file (see Section 6.5).

(**) The soil type of some coarse grid models is not compatible to the soil types used in the COSMO-Model. These soil types are not used.

If external parameters for the COSMO-Model have to be interpolated from the coarse grid model, the following parameters are also necessary:

Z0	Roughness length
PLCOV	(Actual) plant cover during vegetation period
ROOTDP	(Actual) root depth

6.3 External Parameter Files for the COSMO-Model

External parameter files for the COSMO-Model are provided for different rotated coordinates, resolutions and domains. The software package which produces these files is called EXTPAR[1] and is official COSMO software. The files can be obtained from DWD, or from the web-based frontend tool WebPEP[2] (Preparation of External Parameters) of the CLM-Community.

The following table shows different files available from DWD's ftp-server with a short characterization. The filename contains information about the domain (e.g. d0, d1, d5), the resolution in meters (e.g. _07000_ for about 7000 meters) and the size of the fields in grid points (e.g. 961x769). External parameter files for other domains can be produced by DWD on request, if the domain (in rotated coordinates), the rotation (pollat, pollon) and the size of the domain (in grid points) is specified.

(Some of) these files already contain the external parameters for the Subgrid Scale Orography scheme, the minimum stomata resistance of plants, the thermal radiative surface emissivity, the normalized differential vegetation index and the monthly mean values for the Tegen (1997) aerosol climatology. The files with a (*) are older and only contain the Subgrid Scale Orography data.

cosmo_d0_02800_1605x1605.g1_2013111400	dlat = dlon = 0.025	
cosmo_d1_07000_961x769.g1_2009121700	dlat = dlon = 0.0625	(*)
cosmo_d1_14000_481x385.g1_2009121700	dlat = dlon = 0.125	(*)
cosmo_d1_21000_321x257.g1_2009121700	dlat = dlon = 0.1875	(*)
cosmo_d1_28000_241x193.g1_2009121700	dlat = dlon = 0.25	(*)
cosmo_d1_56000_121x113.g1_2009121700	dlat = dlon = 0.5	(*)
cosmo_d5_07000_965x773.g1_2014060200	dlat = dlon = 0.0625	
cosmo_d5_14000_483x387.g1_2013111400	dlat = dlon = 0.125	
cosmo_africa_0.0625_1776x1617.g1	dlat = dlon = 0.0625	
cosmo_asia_0.0625_1921x1601.g1	dlat = dlon = 0.0625	
cosmo_southeastasia_0.0625_1761x1281.g1	dlat = dlon = 0.0625	
cosmo_zimbabwe_0.025_1201x1201.g1	dlat = dlon = 0.025	

The domains d1 are for a rotated grid with pollat=32.5 and pollon=-170.0. Domains d0 and d5 are for a rotated grid with pollat=40.0 and pollon=-170.0. pollat and pollon

[1] EXTPAR: www.cosmo-model.org/content/support/software/ethz/EXTPAR_user_and_implementation_manual.pdf
[2] WebPEP: www.clm-community.eu

give the coordinates of the rotated north pole in real geographical coordinates. The domains `africa`, `asia`, `southeastasia` and `zimbabwe` are unrotated (`pollat=90.0` and `pollon=-180.0`) and the coordinates are ordinary geographic.

The area covered by the different domains is given in the next table.

	Rotated coordinates				Geographical coordinates			
	`startlat`	`endlat`	`startlon`	`endlon`	`startlat`	`endlat`	`startlon`	`endlon`
`d0`	-20.05	20.05	-20.05	20.05	27.11 N	63.69 N	11.21 W	56.59 E
`d1`	-38.75	9.25	-26.75	33.25	14.54 N	51.49 N	11.26 W	70.36 E
`d5`	-24.125	24.125	-30.125	30.125	20.00 N	60.16 N	19.17 W	77.01 E
`africa`	-50.50	50.50	-30.50	80.4375	50.50 S	50.50 N	30.50 W	80.4375 E
`asia`	-20.00	80.00	30.00	150.00	20.00 S	80.00 N	30.00 E	150.00 E
`se-asia`	-30.00	50.00	60.00	170.00	30.00 S	50.00 N	60.00 E	170.00 E
`zimbabwe`	-30.00	0.00	10.00	40.00	30.00 S	0.00 N	10.00 E	40.00 E

6.4 Available External Parameter Files for the GME

For DWD's global model GME, the following external parameter files are available on the ftp-server:

`invar_i128a`	resolution about 60 km
`invar_i192a`	resolution about 40 km
`invar_i192a.new`	resolution about 40 km; new raw data set
`invar_i384a`	resolution about 20 km

The file `invar_i192a.new` must be used for GME data after October, 24^{th}, 2007, 12 UTC!

6.5 Available External Parameter and additional Files for ICON

To run ICON2LM, 3 different files are necessary besides the ICON forecast or analysis data:

- **External Parameters:**
  As for all other coarse grid input models, an ICON external data file is needed, which contains the land-sea-mask (`FR_LAND`) and the ICON soil typ used (`SOILTYP`).

  If your ICON dataset is a cut-out domain produced by ICON-SUB and just covers a certain COSMO-domain (this should the case for most external operational users), the corresponding ICON external parameters do not contain the global fields, but only fields for the cut-out domain. This is in contrast to GME and is due to the special ICON data structure.

NOTE: Up to now the external data file can only be given as NetCDF File.

- **ICON Grid Files:**
  The computation of the ICON grid is very expensive (on the order of hours!) and cannot be done by INT2LM. Therefore all grid specifications are read from an external grid file.

  As for the external parameters, a special ICON grid file will be created for every ICON-SUB domain.

  NOTE:
  This is a special ICON feature. For all other coarse grid input models, INT2LM can compute the corresponding grids and takes necessary information from namelist input.

  NOTE:
  The ICON grid file is available only in NetCDF format!

- **ICON HHL:**
  ICON as a non-hydrostatic model also uses the new general vertical coordinate (similar to COSMO). To specify the vertical ICON grid, a three-dimensional field called HHL (height of half levels) is necessary for the computations. The lowest level of this field just specifies the height of the orography, `HSURF`. HHL can either be available in the first ICON data file, or it must be given by a separate file called "HHL-file".

  NOTE:
  The optional file containing ICON's HHL field has to be in GRIB2 format.

6.6 Conventions for File Names

The initial and boundary fields needed for the model are provided either in Grib or in NetCDF format. Also for the output files, one can choose between Grib or NetCDF. Restart files are written in binary format with full precision. There is one file for the initial fields and also for every set of boundary fields. The following conventions apply for the filenames.

A file name for the COSMO-Model or the INT2LM has the general form

`yheader // ydate // yextension` (for Grib files),

or

`yheader // ydate // yextension // '.nc'` (for NetCDF files),

where `yheader`, `ydate` and `yextension` have the following meaning:
`yheader`: File header (usually 3 characters, only in case of ICON 4 characters)

- first character: specifies the model **(except for ICON, see below!)**

  `g`: GME (global model)
  `l`: COSMO-Model
  `e`: ECMWF model IFS (Integrated Forecast System)
  `c`: A general (global) climate model

- second character **(except for ICON)**:

  `a:` analysis file (uninitialized)

  `i:` analysis file (initialized)

  `b:` boundary file

  `f:` forecast files

  `r:` restart files

- third character **(except for ICON)**: specifies the region covered by the data

  `f:` full model domain

  `s:` subdomain

`ydate`: There are two forms of specifying the date, either with the full date or relative to the start date:

- In the name of analysis files (second character in the header `a` or `i`) the full date is specified **(not for ICON at the moment!)**: `ydate` = `'yyyymmddhh'` with `yyyy`: year; `mm`: month; `dd`: day; `hh`: hour.

  Example:

`laf1992072100`	COSMO-Model, uninitialized analysis for full model domain from July, 21st, 1992.

- In forecast, boundary or restart files, `ydate` consists of a single character (the time unit of forecast range, `ytunit`), followed by a string.

  `ydate` = `ytunit // '`*string*`'`

  Depending on `ytunit`, the string has the following meaning:

  `t:` timestep mode: forecast range given in timesteps

  `f:` forecast mode: the forecast range is given in the form `ddhhmmss`, where `dd`: day, `hh`: hour, `mm`: minute, `ss`: second **(only possibility for ICON at the moment!)**

  `c:` climate mode: the forecast range is given in the form `yyydddhh`, where `yyy`: year, `ddd`: day of the year, `hh`: hour

  `d:` day mode: the full date is given in the form `'yyyymmddhh'`, where `yyyy`: year; `mm`: month; `dd`: day; `hh`: hour

  NOTE: Since Version 1.20, the date for the *day mode* and for the analysis file can also be given with additional 4 digits (2 for minutes, 2 for seconds). The user can activate this by specifying the initial date in this form.

`yextension` (1 character, optional): Extension, e.g. data interpolated from model to pressure levels.

Examples:

`lbff00050000`	COSMO-Model, file with boundary values for hour 5
`lfff01233000`	COSMO-Model, forecast at day 1, 23 hours and 30 minutes.
`lrff01000000`	COSMO-Model, restart file for day 1.

ICON: The `yheader` is given by the namelist parameter `yicon_input_prefix` (free, up to 32 characters, default `'igfff'`). ICON data do not have strict prefixes for analysis- and forecast-data and do not follow the COSMO logic in this respect. For INT2LM, we expect the ICON data to consist of the prefix followed by a forecast time stamp, e.g.,

```
igfff00000000

igfff00030000

igfff00060000

igfff00090000
```

...

If this should be not the case with your data, rename your files accordingly or create links using the above name convention.

Section 7

Namelist Input for INT2LM

The execution of INT2LM can be controlled by 6 `NAMELIST`-groups:

- `CONTRL` – parameters for the model run
- `GRID_IN` – specifying the domain and the size of the coarse grid
- `LMGRID` – specifying the domain and the size of the LM grid
- `DATABASE` – specification of database job
- `DATA` – controlling the input and output
- `PRICTR` – controlling grid point output

NEW: Since INT2LM 2.1 the namelist group `DATABASE` is not required any more.

All `NAMELIST`-groups have to appear in the input file `INPUT` in the order given above. Every group is read in a special subroutine called `input_`*groupname*. These subroutines set default values for all parameters and check most parameters that have been changed for correctness and consistency.

The `NAMELIST` variables can be specified by the user in the run-scripts for the INT2LM, which then create the `INPUT` file. An example of `INPUT` is shown in Figure 7.1.

```
&CONTRL
 ydate_ini='2012051400', ydate_bd='2012051312',
 hstart=0.0, hstop=6.0, hincbound=1.0,
 linitial=.TRUE.,  lboundaries =.TRUE.,
 nprocx=4, nprocy=8, nprocio=0, lreorder=.FALSE.,
 yinput_model='ICON',
 lfilter_oro=.TRUE., eps_filter=0.1,
 ilow_pass_oro=1, ilow_pass_xso=0, rxso_mask=0.0,
 lfilter_pp=.FALSE., itype_balance_pp=2, lmultlay_deepsoil_clim_hcorr=.TRUE.,
 itype_fast_waves_lm=2, itype_profiles_vert_interp=2, norder_filter=5,
 lmulti_layer_in=.TRUE., lmulti_layer_lm=.TRUE., lprog_rho_snow=.TRUE.,
 lprog_qi=.TRUE., lprog_qr_qs=.TRUE., luvcor=.TRUE.,
 lsso=.TRUE., lforest=.TRUE., llake=.FALSE., lbdclim=.FALSE., lt_cl_corr=.TRUE.,
 lseaice=.TRUE., itype_ndvi=0, idbg_level=2,
 itype_albedo=3, itype_aerosol=1,
 lcheck_uuidOfHGrid=.TRUE., nproma_icon=8,
/
&GRID_IN
 ke_soil_in=7,
 ke_in_tot=90,
 yicon_grid_cat = '/path/to/icon/grid/file/',
 yicon_grid_lfn = 'icon_grid_0026_R03B07_G.nc',
 nrootdiv_icon  = 3, nbisect_icon = 7, vcflat_icon = 16000.0,
 nlevskip=0
/
&LMGRID
 startlat_tot  = -20.0, startlon_tot  = -18.0,
 pollat=40.0,        pollon=-170.0,
 dlon=0.0625,        dlat=0.0625,
 ielm_tot=665,         jelm_tot=657,        kelm_tot=40,
 ke_soil_lm=7, ivctype=2, irefatm=2, delta_t=75.0, h_scal=10000.0,
 lanalyt_calc_t0p0=.TRUE., lnewVGrid=.true.,
/
&DATABASE
/
&DATA
 ie_ext=965, je_ext=773,
 ylmext_lfn='lm_d5_07000_965x773.sso.mol.g1',
 ylmext_cat='/e/rhome/routfor/routfox/lm/const/',
 ylmext_form_read='apix',
 yinext_lfn='icon_extpar_0026_R03B07_G_20140731.nc',
 yinext_form_read='ncdf',
 yinext_cat='/e/rhome/routfor/routfox/icon/const/',
 yin_form_read='apix',
 yin_cat='/e/uscratch/uschaett/ICON/data/',
 yinput_type='forecast',
 ylm_cat='/e/uscratch/uschaett/COSMO_EU_input/'
 ylm_form_write='grb1',
 nprocess_ini = 131, nprocess_bd = 132,
 nl_soil_in=2, nl_soil_lm=2,
/
&PRICTR
 lchkin=.TRUE., lchkout=.TRUE.,
/
```

Figure 7.1: Example file INPUT for interpolation of ICON data

7.1 CONTRL — Parameters for the Model Run

Initial time and forecast range

Name	Type	Definition / Purpose / Comments	Default
ydate_ini	CHAR	start of the forecast	' '
ydate_bd	CHAR	start of the forecast of the boundary model (if older forecast data shall be used)	' '
itype_calendar	LOG	to specify a certain type of calender 0: Gregorian calendar (at the moment we still have the Julian calendar) 1: every year has 360 days 2: every year has 365 days	0
hstart	REAL	start of the forecast in hours	0.0
hstop	REAL	end of the forecast in hours	0.0
hincbound	REAL	time increment (in hours)	0.0
nincwait	INT	seconds to wait until next attempt if a ready file is not available	0
nmaxwait	INT	maximum seconds to wait until abort if a ready file is not available	0
ytrans_in	CHAR	directory for reading ready-files	' '
ytrans_out	CHAR	directory for writing ready-files	' '

Domain decomposition and parallelization

Name	Type	Definition / Purpose / Comments	Default
`nprocx`	INT	number of PEs in x-direction of the LM-grid	1
`nprocy`	INT	number of PEs in y-direction of the LM-grid	1
`nprocio`	INT	number of extra PEs for asynchronous I/O	0
`nboundlines`	INT	number of boundary lines of a subdomain	1
`lreorder`	LOG	if .TRUE., the PEs can be reordered for the cartesian MPI-communicator	.TRUE.
`lasync_io`	LOG	to run the model with extra (asynchronous) processors for I/O	.FALSE.
`ldatatypes`	LOG	if .TRUE., MPI-Datatypes for some communications shall be used	.FALSE.
`ncomm_type`	INT	type of communication	1

Basic Control

Name	Type	Definition / Purpose / Comments	Default
lgme2lm	LOG	eliminated from Version 1.14 on, now yinput_model=’GME’	.TRUE.
lec2lm	LOG	eliminated from Version 1.14 on, now yinput_model=’IFS’	.FALSE.
llm2lm	LOG	eliminated from Version 1.14 on, now yinput_model=’COSMO’	.FALSE.
lcm2lm	LOG	eliminated from Version 1.14 on, now yinput_model=’CM’	.FALSE.
yinput_model	CHAR*5	string to identify the input model; valid options are: ’COSMO’ limited area model COSMO ’CM’ climate model standard format ’GME’ former global model of DWD ’IFS’ Integr. Forecast System of ECMWF New since Version 2.1 ’ICON’ new global model of DWD	’ ’
lanafg	LOG	eliminated from Version 1.9 on. Use analyses as input data (first guess) replaced by yinput_type in /DATA/	.FALSE.
linitial	LOG	compute initial data for the COSMO-Model	.FALSE.
lboundaries	LOG	compute lateral boundaries for the COSMO-Model	.TRUE.
lbdclim	LOG	produce additional boundary fields that are needed for long term simulations.	.FALSE.
leps_ana	LOG	produce initial data for ensemble mode Introduced in Version 2.03.	.FALSE.
leps_bc	LOG	produce boundary data for ensemble mode	.FALSE.
lseaice	LOG	interpolate sea ice variables from coarse grid to COSMO-Model	.FALSE.
luvcor	LOG	correct winds for given surface pressure tendency	.TRUE.
l_chemistry	LOG	Eliminated in Version 1.22 switch to process additional chemistry fields Replaced by l_art	.FALSE.
l_art	LOG	switch to process additional fields for COSMO-ART	.FALSE.

Name	Type	Definition / Purpose / Comments	Default
l_art_nested	LOG	switch to process additional fields for COSMO-ART in case of nesting runs	.FALSE.
l_smi	LOG	switch for using an optional interpolation for the soil humidity	.FALSE.
lmixcld	LOG	switch for using an optional treatment of the humidity	.FALSE.
l_cressman	LOG	switch for using a cressman scheme during 'M'atch interpolation	.FALSE.
l_bicub_spl	LOG	switch for using a bicubic spline interpolation	.FALSE.
idbg_level	INT	Selects the verbosity of ASCII output during a model run. The higher the value, the more debug output is written to standard output.	2
lprintdeb_all	LOG	In most cases, the debug output is only written from one processor (with ID=0). With lprintdeb_all=.TRUE., all processors will print the debug output.	.FALSE.
ltime_proc	LOG	detailed timings of the program (per PE)	.FALSE.
ltime_mean	LOG	detailed timings of the program (mean value)	.FALSE.
lroutine	LOG	run the program as routine-job	.FALSE.
lextpar_lm_only	LOG	Process only the external parameters from COSMO at initial time. At all other times, atmospheric fields for boundary data can be processed as usual if lboundaries=.TRUE. Introduced in Version 2.03.	.FALSE.

Special Control Variables for ICON2LM

Name	Type	Definition / Purpose / Comments	Default
l_use_vn	LOG	use the normal velocity (VN: velocity normal to edges) instead of wind components U and V.	.FALSE.
nproma_icon	INT	chunk length for ICON fields internally used. With this variable, the size of the innermost loops for ICON variables can be adjusted. On cache based processors, a small value (8-16) is beneficial, while on vector processors, a value representing the machine internal vector length (e.g. 256 for NEC-SX9) would be used.	8

Name	Type	Definition / Purpose / Comments	Default
lcheck_uuid_OfHGrid	LOG	to check the *unique universal Identifier* for ICON's horizontal grid. If true, the UUID of the records contained in the ICON grid file is checked against the UUIDs of the records read from the ICON forecast data. If they do not match, INT2LM aborts, because it must be assumed that different grids are used.	.TRUE.
		The next three variables have been introduced in Version 2.07.	
l_write_precalc	LOG	If .TRUE., RBF coefficients for interpolating ICON data are computed and written to the following files. These files can be read by subsequent runs then. • RBF_COEFFS_s.nc: scalar grid points • RBF_COEFFS_u.nc: u grid points • RBF_COEFFS_v.nc: v grid points	.FALSE.
l_read_precalc	LOG	If .TRUE., RBF coefficients for interpolating ICON data are not computed, but read from files to save computing time.	.FALSE.
y_precalc_cat	LOG	Directory, where to write / read the files with precalculated RBF coefficients.	'./'
lprog_prrcon_prscon	LOG	If .TRUE., interpolation of the 3D convective rain rate (ICON shortName = "PRR_CON") and snow rate ("PRS_CON") from ICON input to COSMO for initial and boundary files. In COSMO, these rates are used to retain the moisture contained in convective precipiation either in the humidity field or in the grid scale rain- and snow hydrometeors to compensate their instantaneous removal from the atmosphere in ICON. THIS FEATURE IS STILL EXPERIMENTAL AND NOT YET AVAILABLE IN OFFICIAL COSMO!	.FALSE.

Controlling use of additional external parameters

Name	Type	Definition / Purpose / Comments	Default
lforest	LOG	read external parameters for the ground fraction of evergreen and deciduous forest	.FALSE.
lemiss	LOG	read a map from the external parameters for the thermal radiative surface emissivity	.FALSE.
lsso	LOG	read external parameters for the subgrid scale orography scheme	.FALSE.
lskinc	LOG	read external parameter for the skin conductivity	.FALSE.
l_s_oro	LOG	read external parameter for the slope of orography	.FALSE.
lradtopo	LOG	read external parameters for the topographic corrections of radiation	.FALSE.
nhori	INT	number of sectors for the horizon array used by the topographic correction of the radiation	24
lstomata	LOG	read a map from the external parameters for the minimum stomata resistance of plants	.FALSE.
llake	LOG	read and process external parameters for lakes	.FALSE.
llake_coldstart	LOG	initialize prognostic lake variables for cold start	.FALSE.
lurban	LOG	read and process the external parameter URBAN for an urban module. NOTE: Not connected to the new COSMO urban module TERRA_URB! It is for another urban module which is NOT official COSMO code. This module has to be implemented on your own!	.FALSE.
lterra_urb	LOG	for the new urban module TERRA_URB in COSMO, read and process the external parameters AHF (anthropogenic heat flux) and ISA respectively FR_PAVED (Impervious surface area respectively paved fraction — both names denote the same field, and their name depends on the EXTPAR version) Currently only works for COSMO external parameters in grib1 or netcdf format, and if grib1, the reading has to be done by the old libgrib1 and not by grib_api (ylmext_form_read = 'grb1', not 'apix'!). Introduced in Version 2.03.	.FALSE.

Control over variables that have to be written for the COSMO-Model

Name	Type	Definition / Purpose / Comments	Default
lvertwind_ini	LOG	compute vertical wind for initial data	.TRUE.
lvertwind_bd	LOG	compute vertical wind for boundary data	.FALSE.
lprog_qi	LOG	compute initial and boundary values for the cloud ice scheme	.FALSE.
lprog_qrqs	LOG	Renamed to lprog_qr_qs in Version 1.22	.FALSE.
lprog_qr_qs	LOG	compute initial and boundary values for rain and snow	.FALSE.
lprog_qg	LOG	compute initial and boundary values for the graupel scheme	.FALSE.
lprog_qni	LOG	compute initial and boundary values for the number densities QNICE	.FALSE.
lprog_qn_crsg	LOG	compute initial and boundary values for the number densities QNCLOUD, QNRAIN, QNSNOW and QNGRAUPEL	.FALSE.
qvmin	REAL	security minimum value for water vapor	1E-12
qcmin	REAL	security minimum value for cloud water	1E-12
qimin	REAL	security minimum value for cloud ice content	1E-12
lmulti_layer_lm	LOG	compute fields for multi-layer soil model Note that the old 2-layer soil model is no longer available in the COSMO-Model since version 5.04f. For using older versions, this switch is still available in INT2LM.	.FALSE.
lmulti_layer_in	LOG	use multi-layer soil model variables from input fields	.FALSE.
lprog_rho_snow	LOG	read and interpolate the snow density to the COSMO-Model grid	.FALSE.

Name	Type	Definition / Purpose / Comments	Default
itype_w_so_rel	INT	to select the type of relative soil moisture input 0: use an artificial profile relative to the pore volume 1: relative to pore volume (read from coarse grid data) 2: relative to field capacity (read from coarse grid data) 3: the soil moisture profile in relation to the pore volume is kept constant below the deepest layer of the input soil model 4: the soil moisture profile related to field capacity is kept constant below the deepest layer of the input soil model	1
itype_t_cl	INT	to select the source for the deep soil (climatological) temperature 0: take t_cl from coarse grid model. 1: take t_cl from the external parameters for the COSMO-Model	0
itype_rootdp	INT	to select the treatment of the external parameter for root depth. See 5.1.2 for more information. 0: input from external parameter file for the COSMO-Model is taken and modified with an annual cycle. 1: input from external parameter file for the COSMO-Model is taken as is but with a maximal value of 0.12. 2: input from external parameter file for the COSMO-Model is taken and modified with an annual cycle. In addition, the values are adapted to ECOCLIMAP niveau. 3: input from external parameter file for the COSMO-Model is taken and modified with an annual cycle but without maximum cut off. 4: input from external parameter file for the COSMO-Model is taken without modifications.	0

Name	Type	Definition / Purpose / Comments	Default
itype_ndvi	INT	to select the treatment of plant cover and leaf area index. See 5.1.2 for more information. 0: plant cover and leaf area index for the COSMO-Model and for a special day are produced by using the data sets for vegetation and for rest. These are modified with an annual cycle. 1: plant cover and leaf area index for the COSMO-Model and for a special day are produced by using only the data set for vegetation and an averaged ndvi ratio. This ndvi ratio is computed as a weighted mean between monthly mean values, which are taken from the external parameter data set for the COSMO-Model. 2: plant cover, leaf area index and roughness lenght for the COSMO-Model and for a special day are produced by using 12 monthly climatological mean values for plant cover, leaf area index and roughness length. These values are read from the external parameter data set for the COSMO-Model.	0

Name	Type	Definition / Purpose / Comments	Default
itype_aerosol	INT	to select the treatment of aerosols. See 5.1.2 for more information. 1: No additional fields for aerosol types are read from the external parameters. The COSMO-Model has to run with constant values for the aerosol distribution on rural areas, urban areas, desert areas or the sea. 2: Additional external parameters for aerosol types of sulfate, mineral dust, organic, black carbon and sea salt are read as monthly mean values (Tegen et al. (1997)). Actual values for the current day are computed and given to the COSMO-Model. The next three options have been introduced in Version 2.05. Note: Experimental from COSMO Priority Project T^2(RC)^2. Not usable in an official COSMO-Model version right now! 3: Additional external parameters for aerosol optical thickness, single scattering albedo and asymmetry parameter are read as monthly mean values (Kinne et al. (2013)). Actual values for the current day are computed and given to the COSMO-Model. 4: Processing of CAMS aerosols from IFS (actual global aerosol forecasts). 5: Processing of mineral dust aerosol from ICON-ART.	1
itype_albedo	INT	to select the type of solar surface albedo. See 5.1.2 for more information. Introduced in Version 1.19. 1: surface albedo is a function of soiltype (method up to now and still default) 2: surface albedo is determined by two external fields for dry and for saturated soil. 3: A background albedo is prescribed by external fields, which give average values for every month. 4: The vegetation albedo is modified by forest fraction.	1

Name	Type	Definition / Purpose / Comments	Default
lt_cl_corr	LOG	if .TRUE., perform an alternative height correction for t_cl	.FALSE.
luse_t_skin	LOG	if .TRUE., use ECMWF skin temperature for surface	.FALSE.
lante_0006	LOG	if .TRUE., force to use ECMWF dataset before 27 June 2000	.FALSE.
lpost_0006	LOG	if .TRUE., force to use ECMWF dataset after 27 June 2000	.FALSE.

Treatment of orography and filtering

Basically the filtering of the orography should not be done with INT2LM because some other parameters (roughness length, topographical corrections in the radiation scheme, subgrid scale orography scheme) depend on the orography you choose. Therefore all the possible filtering mechanism described below have also been implemented in the program for the generation of the external parameters, called EXTPAR (during 2014).

Name	Type	Definition / Purpose / Comments	Default
lfilter_oro	LOG	if .TRUE., filter the orography	.FALSE.
lfilter_pp	LOG	if .TRUE., filter the pressure deviation after vertical interpolation	.FALSE.
lbalance_pp	LOG	if .TRUE., compute a hydrostatic balanced pp after vertical interpolation in LM2LM. Removed in INT2LM 2.03; now hardcoded always .TRUE.!	.FALSE.
itype_balance_pp	INT	Type of method for hydrostatic pressure balancing for nonhydrostatic input models: 1: Old method: very old code, perhaps consistent with the Leapfrog core? 2: New method: consistent to RK dynamical core, code overtaken from COSMO model (src_artifdata.f90). The exact mathematical formulation for option 2 depends on new namelist parameter itype_fast_waves_lm. Introduced in Version 2.03.	2
itype_fast_waves_lm	INT	Relevant in case of itype_balance_pp=2: Type of fast-waves solver in the COSMO model, should be consistent with the COSMO run. Introduced in Version 2.03.	2

Name	Type	Definition / Purpose / Comments	Default
itype_profiles_vert_interp	INT	To choose the method for vertical adaptation of input profiles from the input orography to the COSMO orography, in case of non-hydrostatic input models (COSMO, ICON, UM, CM). "1" is the previous method, "2" is the new method. Introduced in Version 2.03.	2
lmultlay_deepsoil_clim_hcorr	LOG	Effective in case of multi-layer soil model for input and output: to specify if blending to a climatological height correction of soil temperatures for deeper soil layers is desired. The normal method is to preserve the temperature differences to the lowest atmospheric temperature across the interpolation, and this method is then only applied in the upper soil levels with a decreasing weight with increasing depth. This should reduce the danger that, e.g., local extremes in near-surface air temperatures, which may have been artificially created by the PBL-profile adaption from coarse to fine orography, create also artificial extremes in deep-soil temperatures where they have a long memory. The climatological height correction is based on a constant temperature gradient of 0.007 K/m with respect to orography difference coarse minus fine levels. In this respect it is similar to the already existing option lt_cl_corr for the climatological deep-soil temperature t_cl in the old 2-layer soil model. Introduced in Version 2.03.	.TRUE.
eps_filter	REAL	parameter for the Raymond (1988) filtering of the orography	10.0
norder_filter	INT	p value of the Raymond orography filtering	1
ilow_pass_oro	INT	type of low-pass filter for orography 1: use the Raymond filter with eps_filter and norder_filter 3: 9-point filter (approx. cutoff=3) 4: 9-point filter (approx. cutoff=4) 5: 13-point filter (approx. cutoff=5) 6: 9-point filter (approx. cutoff=6) 8: 13-point filter (approx. cutoff=8)	1
numfilt_oro	INT	number of sequential applications of filter	1

Name	Type	Definition / Purpose / Comments	Default
ilow_pass_xso	INT	type of low-pass filter for extra smoothing of steep orography	0
numfilt_xso	INT	number of sequential applications of xso filter	1
lxso_first	LOG	do an extra smoothing of orography first	.FALSE.
rxso_mask	REAL	mask for extra smoothing of steep orography	0.0
rfill_valley	REAL	mask for valley filling	0.0
ifill_valley	REAL	type of valley filling	1
l_topo_z	LOG	additional smoothing of the topography for LM_Z	.FALSE.
llbc_smooth	LOG	run with a smooth orography transition at the lateral boundaries	.FALSE.
nlbc_smooth	INT	number of grid points for a smooth orography transition at the lateral boundaries	.FALSE.
hmax_sea	REAL	maximal height in coastal areas, below which no filtering is done for sea points. This variable influences the filtering of the orography in coastal areas for grid points, which have a fraction of land less than 0.5 (sea points), but are given an orography height higher than 0.0. If such a grid point is surrounded by enough other sea points, and if the orography height is not heigher than hmax_sea, the filtered value is set back to the original height of the orography. ($0 \leq$ hmax_sea ≤ 20.0)	10.0

7.2 GRID_IN — Specifying the Domain and the Size of the coarse Grid

grid_in contains variables that specify the size and resolution of the coarse grid.

Name	Type	Definition / Purpose / Comments	Default
ni_gme	INT	resolution of GME	128
i3e_gme	INT	number of levels in the vertical	51
yicon_grid_cat	CHAR	Directory of the NetCDF file describing the horizontal ICON grid	' '
yicon_grid_lfn	CHAR	Name of the NetCDF file describing the horizontal ICON grid	' '
lcontrol_fi	INT	to check, whether a control level for geopotential has been read. If it has not been read, abort the program. If it is .FALSE. and a level could not be read, such a level is computed by the program. Introduced in Version 2.06	.FALSE.
kcontrol_fi	INT	control level for geopotential	15
pcontrol_fi	REAL	pressure of control level for geopotential	-1.0
ie_in_tot	INT	ie for input grid (total domain)	141
je_in_tot	INT	je for input grid (total domain)	71
ke_in_tot	INT	ke for input grid (total domain)	60
nrootdiv_icon	INT	Number of root divisions for the ICON grid (the xx in RxxByy). Mandatory for ICON input. Introduced in Version 2.02.	-99
nbisect_icon	INT	Grid level for the ICON grid (the yy in RxxByy). Mandatory for ICON input. Introduced in Version 2.02.	-99
vcflat_icon	REAL	Height in m above which the model levels are flat in ICON input grid. Relevant for the determination of the height level index at which $p \approx p_s - 150\,\text{hPa}$ for extrapolation of the PBL profiles towards the COSMO orography. Relevant for itype_profiles_vert_interp=1. Introduced in Version 2.03.	16000.0
nlevskip	INT	number of missing levels in input grid, counted from model top	0
ke_soil_in	INT	number of levels in input multi-layer soil model	6

Name	Type	Definition / Purpose / Comments	Default
czml_soil_in	REAL	depth of main input soil layers (layer centers). The default specification is (in cm) / 0.005, 0.02, 0.06, 0.18, 0.54, 1.62, 4.86, 14.58 /	see left
pollat_in	REAL	geographical latitude of rotated north pole (in degrees, north > 0)	90.0
pollon_in	REAL	geographical longitude of rotated north pole (in degrees, east > 0)	180.0
dlon_in	REAL	Mesh size in east-west direction	0.5
dlat_in	REAL	Mesh size in north-south direction	0.5
startlat_in_tot	REAL	latitude of the lower left grid point of the input domain (in degrees, north > 0, rotated coordinates)	-35.0
startlon_in_tot	REAL	longitude of the lower left grid point of the input domain (in degrees, east > 0, rotated coordinates)	-30.0
endlat_in_tot	REAL	latitude of the upper right grid point of the input domain (in degrees, north > 0, rotated coordinates)	0.0
endlon_in_tot	REAL	longitude of the upper right grid point of the input domain (in degrees, east > 0, rotated coordinates)	-40.0
p0sl_in	REAL	constant reference pressure on sea-level	10000.0
t0sl_in	REAL	constant reference temperature on sea-level	288.15
dt0lp_in	REAL	d (t0) / d (ln p0)	42.0
lushift_in	LOG	Only for yinput_model='CM': use staggered grid for input u-velocity	.FALSE.
lvshift_in	LOG	Only for yinput_model='CM': use staggered grid for input v-velocity	.FALSE.
east_add_in	INT	Only for yinput_model='CM' and NetCDF input: add an extra column to the east of the input domain taken from the west (periodic BCs), needed for COSMO domains across the date line. The data set of the coarse model has to be global or a global belt, not a regional subset.	0
west_add_in	INT	Only for yinput_model='CM' and NetCDF input: add an extra column to the west of the input domain taken from the east (periodic BCs), needed for COSMO domains across the date line. The data set of the coarse model has to be global or a global belt, not a regional subset.	0

Name	Type	Definition / Purpose / Comments	Default
south_add_in	INT	Only for yinput_model='CM' and NetCDF input: add an extra column to the south of the input domain taken from the south (0-gradient), needed for COSMO domains across North or South Pole. The data set of the coarse model has to be global or circumpolar (covering the COSMO domain).	0
north_add_in	INT	Only for yinput_model='CM' and NetCDF input: add an extra column to the north of the input domain taken from the north (0-gradient), needed for COSMO domains across North or South Pole. The data set of the coarse model has to be global or circumpolar (covering the COSMO domain).	0

7.3 LMGRID — Specifying the Domain and the Model Grid

lmgrid contains variables that specify the model domain in the rotated grid and the size of the total domain.

Name	Type	Definition / Purpose / Comments	Default
pollat	REAL	geographical latitude of rotated north pole (in degrees, north > 0)	32.5
pollon	REAL	geographical longitude of rotated north pole (in degrees, east > 0)	-170.0
polgam	REAL	Angle between the north poles of two rotated grids (in degrees, east > 0); necessary for transformation from one rotated grid to another rotated grid	0.0
dlon	REAL	Mesh size in east-west direction	0.0625
dlat	REAL	Mesh size in north-south direction	0.0625
startlat_tot	REAL	latitude of the lower left grid point of the total domain (in degrees, north > 0, rotated coordinates)	-14.375
startlon_tot	REAL	longitude of the lower left grid point of the total domain (in degrees, east > 0, rotated coordinates)	-6.875
ielm_tot	INT	number of gridpoints of the total domain in east-west direction	213
jelm_tot	INT	number of gridpoints of the total domain in north-south direction	213
kelm_tot	INT	number of gridpoints of the total domain in vertical direction	20
ke_soil_lm	INT	number of levels in multi-layer soil model for the COSMO-Model	6
czml_soil_lm	REAL	depth of main soil layers for the COSMO-Model. The default specification is / 0.005, 0.02, 0.06, 0.18, 0.54, 1.62, 4.86, 14.58 /	see left
czvw_so_lm	REAL	artificial volumetric soil water content profile for the COSMO-Model.	/0.75/

Name	Type	Definition / Purpose / Comments	Default
irefatm	INT	type of reference atmosphere 1: Default as used up to now 2: The reference atmosphere is based on a temperature profile $t_0(z) = (t_{0sl} - \Delta t) + \Delta t \cdot \exp\left(\frac{-z}{h_{scal}}\right)$, where $z =$ hhl(k) is the height of a model grid point. If this option is used, the values for $\Delta t =$ delta_t and $h_{scal} =$ h_scal have also to be set.	1
lanalyt_calc_t0p0	LOG	if set to .TRUE., the values for the reference state of t_0 and p_0 are computed analytically. If it is set to .FALSE., they are only averaged between the half levels.	.FALSE.
ivctype	INT	kind of vertical coordinate system 1: reference pressure based hybrid coordinate. 2: height based hybrid Gal-Chen coordinate. 3: height based hybrid SLEVE coordinate. In comparison to option 2, there is an additional blending to a smoothed orography before blending to horizontal coordinates at the height given by the parameter vcflat below. (not extensively tested). The exact blending behaviour can be influenced by the parameters svc1 and svc2 below. 4: Introduced in Version 2.01 generalized SLEVE coordinate with a modified vertical decay of the topographic signature with height. See Leuenberger et al. (2010)	2
lnewVGrid	INT	to indicate, that a new vertical grid file HHL has to be created	.FALSE.
vcflat	REAL	coordinate value where system changes back to z-system	0.220

Name	Type	Definition / Purpose / Comments	Default
vcoord_d	REAL	vertical coordinate parameter list of pressure (ivctype=1) or height (ivctype=2/3/4) values. The usage of the fortran utility vcoord.f90 for preparation of the parameter values is recommended. For ivctype=1, values have to cover the range 0 to 1 in increasing order (σ coordinates). For ivctype=2, values are heights in m and have to be in decreasing order from the desired model top height down to 0.0 m. Some default sets of height values already exist in the source code for ivctype=2 and kelm_tot=40,50. ivctype=3/4 and kelm_tot=50. If one of these sets should be used, only specify the respective kelm_tot and ivctype and leave vcoord_d out.	(missing)
p0sl	REAL	constant reference pressure on sea-level	10000.0
t0sl	REAL	constant reference temperature on sea-level	288.15
dt0lp	REAL	d (t0) / d (ln p0)	42.0
delta_t	REAL	temperature difference between sea level and stratosphere (for irefatm=2)	75.0
h_scal	REAL	scale height (for irefatm=2)	10000.0
svc1	REAL	decay rate for large-scale part of topography	10000.0
svc2	REAL	decay rate for small-scale part of topography	10000.0
nfltvc	REAL	number of filter applications for topo decomposition	100

7.4 DATABASE — Specification of Database Job

NOTE: Since INT2LM 2.1 this namelist group is obsolete.

Name	Type	Meaning	Default
yinit_order	CHAR	string for initializing csodaban	’ak=nix’
yana_tab	CHAR	database for initial data	’******’
ybd_tab	CHAR	database for boundary data	’******’
nout_sockets	INT	number of sockets for database output per PE (0 means File-IO)	0
nin_sockets	INT	number of sockets for database input per PE (0 means File-IO; has to be ≤ 1)	0
iretry	INT	number of seconds to retry on database failure	0
ibackup_size	INT	size of incore backup space (in bytes) by a database failure	-1
ybackup_dir	CHAR	directory for outcore backup	’.’
idbg_level	INT	debug level for mpe_io	0

7.5 DATA – Controlling the Input and Output

General control variables

Name	Type	Definition / Purpose / Comments	Default
ncenter	INT	originating center identification	78
nprocess_ini	INT	generating process identification for initial values	131
nprocess_bd	INT	generating process identification for boundary values	132
nrbit	INT	pack-rate for the grib code (in bit)	16
nbitmap	INT	user dimension for bitmaps	6000*2
nl_soil_lm	INT	number of prognostic soil water levels for the COSMO data	2
nl_soil_in	INT	number of prognostic soil water levels for the input data	2
nvers	INT	for documenting purposes (mainly in GRIB-Code)	1
ymode_read	CHAR	specify open mode for reading	’r ’
ymode_write	CHAR	specify open mode for writing	’w ’
yvarini	CHAR	list of variables for LM initial data	
yvarbd	CHAR	list of variables for LM boundary data	
l_ke_in_gds	LOG	write the number of vertical levels explicitly to the grid description section	.TRUE.
ytunit_in	CHAR	time unit for input data	’f’
ytunit_out	CHAR	time unit for output data	’f’
yinput_type	CHAR	type of input data ’forecast’ forecast data ’analysis’ analysis data ’ana_init’ initialized analysis data	’forecast’
llockfiles	LOG	indicates whether to use log files or not. Introduced in Version 2.05.	.FALSE.
ltestsuite	LOG	indicates whether to write output for the test-suite. Introduced in Version 2.07.	.FALSE.

Variables for external data and HHL-files (needed for GRIB2)

Name	Type	Meaning	Default
ylmext_cat	CHAR	directory of the external fields for LM/HM	' '
ylmext_lfn	CHAR	name of the file with the external fields for LM/HM	' '
ylm_hhl	CHAR	name of the vertical grid HHL-file that has to be processed in case of COSMO GRIB2 output files. This file has to be in the directory ylmext_cat	' '
yinext_cat	CHAR	directory of the external fields for GME	' '
yinext_lfn	CHAR	name of the file with the external fields for GME	' '
yin_hhl	CHAR	name of the vertical grid HHL-file that has to be read in case of COSMO GRIB2 input files. This file has to be in the directory yinext_cat	' '
ylmext_form_read	CHAR	input format of external data	'grb1'
yinext_form_read	CHAR	input format of external data from coarse grid 'grb1' input is read with DWD's Grib library and has to be GRIB1. 'apix' input is read with ECMWF's Grib library grib_api and can be Grib1 or Grib2. 'ncdf' input format is NetCDF.	'grb1'
ie_ext	INT	west-east size of fields with external parameters	1081
je_ext	INT	north-south size of fields with external parameters	1081

Variables for the models

Name	Type	Definition / Purpose / Comments	Default
yin_cat	CHAR	directory of the GME-fields	' '
yin_form_read	CHAR	input format of data from coarse grid	'grb1'
yicon_input_prefix	CHAR	Prefix of the input data. ICON data do not have strict prefixes for analysis- and forecast-data and do not follow the COSMO logic in this respect. For INT2LM, we expect the ICON data to consist of the prefix given in the namelist variable yicon_input_prefix, followed by a forecast time stamp, e.g., igfff00000000 igfff00030000 igfff00060000 igfff00090000 If this should not be the case with your data, rename your files accordingly or create links using the above name convention.	'igfff'
ybitmap_cat	CHAR	directory of an optional bitmap for GME data	' '
ybitmap_lfn	CHAR	name of the file with an optional bitmap for GME data	' '
ylm_cat	CHAR	directory of the output COSMO-Model data	' '
ylm_form_write	CHAR	output format of COSMO-Model data 'grb1' data are written with DWD's Grib library in GRIB1. 'api1' data are written with ECMWF's Grib library grib_api in Grib1 'api2' data are written with ECMWF's Grib library grib_api in Grib2 'ncdf' data are written in NetCDF. 'nc-4' data are written in NetCDF4. Introduced in Version 2.06.	'grb1'

Name	Type	Definition / Purpose / Comments	Default
ypackingType	CHAR	Introduced in Version 2.06a. To choose GRIB 2 packing type. Up to Version 2.06a this was hard-coded as ’grid_simple’. For GRIB 2, additional packing types have been introduced, which offer an additional packing and therefore result in smaller Grib files (see comments below). The following packing types can now be used by INT2LM (2.06a) and COSMO (depends on configuration settings of eccodes!) ’grid_simple’ default; as is up to now. ’grid_ccsds’ CCSDS packing. ’grid_jpeg’ JPEG 2000 Code Stream Format. ’grid_png’ Using the png compression.	’grid_simple’
npstrframe	INT	thickness of output frames	8
lbd_frame	LOG	if .TRUE., boundary fields include only frames	.FALSE.

Notes on the GRIB 2 packing types:

The following packing types can now be used by INT2LM (2.06a) and COSMO (5.06a) (depends on configuration settings of eccodes!)

- ’grid_simple’: default; as is up to now.
- ’grid_ccsds’: The Consultative Committee for Space Data Systems uses a Golomb-Rice packing scheme, which is rather efficient in terms of size and encoding size for transmitting gridded data. This algorithm can also be applied to meteorological data and is available in GRIB 2 now. Note that this is a lossless packing, but acts upon the grid_simple packing, which looses accuracy.
- ’grid_jpeg’: JPEG 2000 Code Stream Format (Joint Photographic Experts Group) is also a lossless compression method, which acts upon the grid_simple packing.
- ’grid_png’: The png compression engine is based on the deflate method, which is a widely used, patent-free algorithm for universal, lossless data compression.

To use these additional packing types, they also have to be activated for eccodes (with -DENABLE_JPG=ON, -DENABLE_PNG=ON, -DENABLE_AEC=ON). See also the installation instructions for eccodes. The additional packing also takes additional computational time.

Additional specifications for NetCDF-IO

Name	Type	Definition / Purpose / Comments	Default
yncglob_institution	CHAR	originating center name	'-'
yncglob_title	CHAR	title string for the output	'-'
yncglob_source	CHAR	program name and version	'-'
yncglob_contact	CHAR	contact e.g. email address	'-'
yncglob_project_id	CHAR	identification of the project of simulation	'-'
yncglob_experiment_id	CHAR	identification of the experiment of simulation	'-'
yncglob_references	CHAR	URL, report etc.	'-'
ncglob_realization	INT	number of the realization of the experiment	1

7.6 PRICTR — Controlling grid point output

Name	Type	Definition / Purpose / Comments	Default
nlev1pr	INT	k-index for printing the first model layer	10
nlev2pr	INT	k-index for printing the second model layer	20
igp_tot	INT	i-index for printing selected grid points (max. nmaxgp)	
jgp_tot	INT	j-index for printing selected grid points (max. nmaxgp)	
lprps	LOG	print some ps- and fis-fields	.FALSE.
lprt	LOG	print t at 2 levels (nlev1pr,nlev2)	.FALSE.
lpru	LOG	print u at 2 levels (nlev1pr,nlev2)	.FALSE.
lprv	LOG	print v at 2 levels (nlev1pr,nlev2)	.FALSE.
lprgrh	LOG	print grh at 2 levels (nlev1pr,nlev2)	.FALSE.
lprqv	LOG	print qv at 2 levels (nlev1pr,nlev2)	.FALSE.
lprqc	LOG	print qc at 2 levels (nlev1pr,nlev2)	.FALSE.
lprud	LOG	print ud (divergent wind correction)	.FALSE.
lprvd	LOG	print vd (divergent wind correction)	.FALSE.
lprdpdt	LOG	print dpdt (tendency of surface pressure)	.FALSE.
lprgp	LOG	print profiles at selected grid points	.FALSE.
lchkin	LOG	print check-values of input-fields	.FALSE.
lchkout	LOG	print check-values of output-fields	.FALSE.

7.7 EPSCTL — Characterizations for the Ensemble of Boundary Data

The namelist group EPSCTL is only read, if the switch(es) leps_ana and/or leps_bc in the group CONTRL is/are set to .TRUE..

Name	Type	Definition / Purpose / Comments	Default
iepsmem_ana	INT	ID of the member in the ensemble of initial conditions (must be ≥ 0). introduced in Version 2.03	-1
iepstyp_ana	INT	ID of the initial conditions ensemble generation type (must be ≥ 0). Introduced in Version 2.03	-1
iepstot_ana	INT	total number of initial conditions ensemble members (must be ≥ 0). Introduced in Version 2.03	0
lchk_bc_ana	LOG	if .TRUE., check member ID of input data for initial conditions. Introduced in Version 2.03	.FALSE.
iepsmem_bc	INT	ID of the member in the ensemble of boundary conditions (must be ≥ 0).	-1
iepstyp_bc	INT	ID of the boundary ensemble generation type (must be ≥ 0).	-1
iepstot_bc	INT	total number of boundary ensemble members (must be ≥ 0).	0
lchk_bc_typ	LOG	if .TRUE., check member ID of input data for boundary conditions.	.FALSE.

Section 8

Driving Models for the COSMO-Model

Since INT2LM Version 2.1 this chapter is included to describe in more detail, how different coarse grid models can be used as driving models for COSMO. This chapter is especially necessary for DWD's new global model ICON, because there are now some additional features and things necessary to run INT2LM in the ICON2LM mode.

8.1 Using data from ICON

8.1.1 About ICON

The ICON (ISOsahedral Nonhydrostatic) modeling framework is a joint project between the Deutscher Wetterdienst (DWD) and the Max-Planck-Institute for Meteorology (MPI-M) for developing a unified next-generation global NWP and climate modeling system.

The main goals formulated in the initial phase of the collaboration were

- better conservation properties than in the existing global models, with the obligatory requirement of exact local mass conservation and mass-consistent transport,
- better scalability on future massively parallel high-performance computing architectures, and
- the availability of some means of static mesh refinement. ICON is capable of mixing one-way nested and two-way nested grids within one model application, combined with an option for vertical nesting in order to allow the global grid to extend into the mesosphere (which greatly facilitates the assimilation of satellite data) whereas the nested domains extend only into the lower stratosphere in order to save computing time.

In addition, the data of ICON simulations can be used to drive regional models, namely the COSMO-Model. But there is a speciality about ICON, which make the interpolation of ICON data to the COSMO grid different than interpolations implemented so far. This is the grid used for ICON and its implementation in the ICON code. In principal the ICON grid is similar to the GME grid, but in contrast to GME it is implemented as an unstructured grid. All neighboring relations are implemented via indirect addressing.

This technical issue, but also the algorithms used to construct the grid, make the grid generation a very expensive process, which cannot be done during INT2LM runs. In order to process ICON data, it is necessary to load precalculated horizontal grid information as an input. This information is stored within so-called grid files. These grid files are provided either for the global ICON grid or for special cut-out domains tailored to encompass specific COSMO-Model domains.

8.1.2 Compiling and Linking INT2LM for ICON2LM

Reading and interpolating ICON forecast data is implemented in a new module named `src_icon_interpol.f90` (similar to `src_gme_interpol.f90`).

To activate the interpolation of ICON data to the COSMO grid, INT2LM has to be compiled and linked with the NetCDF library, because the ICON grid file is available in NetCDF only. Therefore the pragma `-DNETCDF` has to be specified for compiling.

ICON forecast data can be given as GRIB2 or as NetCDF files. If GRIB2 is used, INT2LM has to be compiled and linked with the `grib_api` or the `eccodes` library. Therefore the pragma `-DGRIBAPI` has to be specified for compiling.

8.1.3 Running INT2LM for ICON2LM

The following steps are necessary to run INT2LM successfully with ICON data:

1. Basic Settings:

   - The basic namelist switch to activate the interpolation of ICON data is:

     `yinput_model='ICON'`

     This namelist variable is already contained in the group `/CONTRL/`.

   - For users who received data from the global DWD-ICON but cutted to a limited domain (not the entire globe), it was mandatory to avoid the UUID check for the grid before INT2LM Version 2.05, because the check failed for cut-out domains, since no UUID was contained in the corresponding grid files. Since autumn 2018 and INT2LM Version 2.05, this check can be activated!

     `lcheck_uuidOfHgrid='.TRUE.'`.

     Note that, if `lcheck_uuidOfHgrid='.FALSE.'`, strange errors happen when running INT2LM with a wrong grid file.

2. Specification of the ICON grid:
   The ICON grid file necessary for the COSMO-Domain has to be specified. This can be done by setting the new namelist variables in the group `/GRID_IN/`.

   - `yicon_grid_cat`: directory of the netcdf file describing the horizontal ICON grid

- `yicon_grid_lfn`: name of the netcdf file describing the horizontal ICON grid

Note, that these files are available only in NetCDF format!

Also, the ICON grid specification (RxxByy) has to be given explicitly by two new namelist variables:

- `nrootdiv_icon`: Number of root divisions (the "xx"),
- `nbisect_icon`: Grid level (the "yy"),

and the height, above which ICON levels become flat:

- `vcflat_icon`: specify, if different from default (= 16000.0 m).
  (Introduced in INT2LM 2.03)

In addition, the (already existing) namelist variables have to be set in `/GRID_IN/`:

- `ke_in_tot`: number of vertical levels of ICON data.
  `ke_in_tot` has to be specified to the number of levels originally used to compute the ICON data, regardless whether ICON vertical levels are skipped by using `nlevskip > 0` (see below).

  Example: If ICON was run with 90 levels, `ke_in_tot` has to be set to 90 even if you use `nlevskip > 0` and your ICON data input file may or may not contain less than 90 levels.
- `nlevskip` (optional): number of missing levels in input grid.
  With `nlevskip`, the number of vertical layers actually used for computations can be decreased (counted from model top) to save computing time and memory. `nlevskip` has to be smaller than the level index of the first ICON model layer above the intended COSMO-Model domain.

  Note that, if using ICON grib2-input format, different values of nlevskip might lead to slightly different results because of the vertical cubic spline interpolation.

3. ICON external data:
   An ICON external data file is needed, which contains the land-sea-mask (`FR_LAND`) and the ICON soiltyp (`SOILTYP`) used. Up to now the external file can only be given as NetCDF File.

   Also for GME an external file must be specified, but while the GME external file contains the global fields, the ICON external file only contains the fields for a specified COSMO domain, which must correspond to the horizontal grid determined by the ICON grid file (see 2).

   The following namelist variables in group `/DATA/` have to be specified:

   - `yinext_cat`: directory which contains file with ICON external parameters
   - `yinext_lfn`: name of the file with ICON external parameters
   - `yinext_form_read`: only `'ncdf'` possible at the moment.

   Note that it is not possible to interpolate the ICON external parameters to the COSMO grid. The namelist setting `ylmext_lfn = 'interpolate'` is not possible with ICON as input model.

4. ICON HHL:
   ICON as a non-hydrostatic model also uses the new general vertical coordinate (similar to COSMO). To specify the vertical ICON grid, a field called HHL (height of half levels) is necessary for the computations.

   HHL can either be available in the first ICON data file, or it must be given by a separate file. The filename of an extra file must be given in namelist group `/DATA/` by the variable:

   - `yin_hhl`: name of the input HHL file

   The file has to be in the directory yinext_cat, specified for the external parameters.

   Note: The optional file containing ICON's HHL field has to be in GRIB2 format.

5. ICON driving data:
   From DWD, at the moment only ICON forecast data (not the analysis files) can be used as initial and boundary data for COSMO. The reason is that due to the large data amount DWD does not store every analysed prognostic field necessary for COSMO in its data base, because the start time vv=0 of every forecast already contains the analyses and double storage is to be avoided.

   ICON forecast data can be given in GRIB2 or in NetCDF. DWD will only provide GRIB2 data. These data have the prefix `'igfff'`. However, in future data from other providers might have a different suffix, which then can be specified via the namelist parameter `yicon_input_prefix` in group `/DATA/`. Its default is `'igfff'`.

   The fields necessary from ICON are the same that are necessary from GME, but an additional field for the ground temperature (`T_G`) is needed.

   Fields necessary to produce initial data for the COSMO-Model:

   - atmospheric fields: `U, V, W, T, P, QV, QC, QI, QR, QS` (`W` is only necessary, if `lvertwind_ini=.TRUE.`, which is usually the case)
   - surface fields: `T_SNOW, W_SNOW, FRESHSNW, RHO_SNOW, T_G, W_I, QV_S, H_ICE, T_ICE`
   - soil fields: `T_SO, W_SO`

   Fields necessary to produce boundary data for the COSMO-Model:

   - atmospheric fields: `U, V, W, T, P, QV, QC, QI, QR, QS` (`W` is only necessary, if `lvertwind_bd=.TRUE.)`
   - surface fields: `T_SNOW, W_SNOW, T_G`
   - soil fields: `T_SO(0)` (which in fact is a surface field)

8.1.4 ICON domain check

Since INT2LM 2.03 there is a domain check for ICON. This is relevant for ICON data from a nest domain (such as DWD's ICON-EU), or for regional data sets which have been cut-out from the global ICON data by the utility ICON-SUB to reduce the data amount for internet transfers.

In previous versions, INT2LM just crashed with a segmentation fault if the COSMO domain was outside the ICON-covered part of the globe. This is not nice to the user and therefore a new domain checking for ICON input has been implemented:

- If the debug level "high enough", (`idbg_level` $>$ 10), new ASCII output files

  - `YUICON_COSMO_DOMAINS_[suv]`

  ("s"=scalar points, "u"=u-points, "v"=v-points) are generated in the directory from which the INT2LM has been started, showing the ICON- and overlapping COSMO domains as ASCII-ART geographic maps. The annotated coordinates are always geographic longitudes and latitudes, even if the COSMO domain is rotated. Fig. 8.1 shows an example of the contents of such a file. This is a valuable information for the user even if there are no actual mismatches.

- In case of not fully overlapping domains, (a) similar file(s)

  - `YUERROR_ICON_COSMO_DOMAINs_MISMATCH_[suv]`

  is/are generated regardless of the debug level, and INT2LM stops with a corresponding error message. Fig. 8.2 shows an example of such a file. The generated ASCII-ART maps immediately show graphically where the problem is located.

Additionally, for ICON-SUB generated data sets, the parameters of the rotated lat/lon grid are checked and additional diagnostic error messages are generated.

```
Graphical depiction of ICON-domain and excessing COSMO domain
-------------------------------------------------------------


  I = ICON domain points
  C = COSMO domain points INSIDE ICON domain
  X = COSMO domain points OUTSIDE ICON domain


         17.50     20.00     22.50     25.00     27.50     30.00     32.50     35.00

           |         |         |         |         |         |         |         |
           |         |         |         |         |         |         |         |
           |         |         |         |         |         |         |         |
           |         |         |         |         |         |         |         |
           |         |         |         |         |         |         |         |
           |         |         |         |         |         |         |         |
           |         |         |         |         |         |         |         |
  5.00    -+---------+---------+---------+---------+---------+---------+---------+---
           |         |         |         |         |         |         |         |
           |         |         |         |         |         |         |         |
           |         |         |         |         |         |         |         |
           |         |         |         |         |         |         |         |
           |         |         |         |         |         |         |         |
           |         |         |         |         |         |         |         |
           |         |         |II   I  II   II  I I    I    |IIII  I  | IIII   I|
           |         |       IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
           |         |     IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
  2.50    -+---------+---IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
           |         |  IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
           |         |  IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
           |         |  IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
           |         |  IIIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |   IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |  IIIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |   IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |   IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |   IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
  0.00    -+---------+---IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |   IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |   IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |   IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |  IIIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |  IIIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |   IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |   IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |   IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |   IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
-  2.50   -+---------+--IIIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |  IIIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |   IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
           |         |   IIIIIIIIIIIICCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIII
```

Figure 8.1: Example of a file `YUICON_COSMO_DOMAINS_s` for an ICON cut-out domain in East Africa. The grid coordinates are geographic longitudes and latitudes. An **I** denotes regions where ICON grid cells are located and at **C** locations the COSMO points are all within enclosing ICON grid cells. At **X** locations there would be **problematic** COSMO points outside the ICON region, but this is not the case here.

```
Graphical depiction of ICON-domain and excessing COSMO domain
-------------------------------------------------------------


  I = ICON domain points
  C = COSMO domain points INSIDE ICON domain
  X = COSMO domain points OUTSIDE ICON domain


           15.00     17.50     20.00     22.50     25.00     27.50     30.00

             |         |         |         |         |         |         |
             |         |         |         |         |         |         |
             |         |         |         |         |         |         |
             |         |         |         |         |         |         |
             |         |         |         |         |         |         |
             |         |         |         |         |         |         |
             |         |         |         |         |         |         |
 20.00    ---+---------+---------+---------+---------+---------+---------+---------
             |         |         |         |         |         |         |
             |         |         |         |         |         |         |
             |         |         |         |         |         |         |
             |         |         |         |         |         |         |
             |         |         |         |         |         |         |
             |         |         |         |         |         |         |
             |         |         |         |II   I  II   II  I I    I    |IIII  I
             |         |         |       IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
             |         |         |     IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
  2.50    ---+---------+---------+---IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
             |         |         |  IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
             |         |         |  IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
             |         |         |  IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
  0.00    ---+---------+-XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
-  2.50   ---+---------+-XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
             |         | XXXXXXXXXXXCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCIIIIIIIIIIIIIIIII
```

Figure 8.2: Example of a file YUERROR_ICON_COSMO_DOMAINs_MISMATCH_s for the same cut-out domain as in Fig. 8.1, but here some COSMO domain points are outside the ICON-covered region (**X**). The user would have to move the COSMO domain to the east or request a larger cut-out domain from DWD.

Section 9

More Details on Special Options

Since INT2LM Version 2.1 this chapter is included to describe some special features of INT2LM in more detail. First section included is on the fresh water lake model (FLake) and how to get initial data for the corresponding prognostic variables.

9.1 Cold Start of the Lake Parameterisation Scheme FLake

As the external-parameter fields of lake fraction and lake depth are generated and the logical switch `llake` is set `.TRUE.`, initial values of FLake prognostic variables should be specified. This can be done on the basis of observational data. Some caution is required since FLake prognostic variables are functionally related through the mean temperature of the water column and cannot be set completely independent. In most situation of interest in NWP and climate modeling, no empirical information is available to initialise FLake prognostic variables for all lakes present in the model domain. Then, initial values of FLake variables can be taken from previous COSMO-Model runs performed with the same numerical domain and horizontal resolution. For NWP applications this procedure is strongly recommended!

If such runs have not been performed or their results are inappropriate for the initialisation for one reason or other, a *cold start* initialisation of FLake prognostic variables should be performed. This can be switched on by setting the namelist variable `llake_coldstart` from group `/CONTRL/` to `.TRUE.` (default value is `.FALSE.`.

`llake_coldstart`	initialize all prognostic lake variables for cold start

The initial prognostic variables of FLake are:

`SALT_LK`	lake salinity [g/kg]
`T_B1_LK`	temperature at the bottom of the upper layer [K]
`H_B1_LK`	thickness of the upper layer [m]
`T_WML_LK`	mixed-layer temperature [K]
`T_MNW_LK`	mean temperature of the water column [K]
`T_BOT_LK`	temperature at the water-bottom sediment [K]
`C_T_LK`	shape factor with respect to the temperature profile in lake thermocline [-]
`H_ML_LK`	thickness of the mixed-layer [m]

For the salinity an extra external parameter `SALT_LK` can be provided, but this is not done by EXTPAR. The CLM-Community has some means to do that. If this parameter is not available, the salinity is initialized to zero.

No ice is assumed at the cold start; the ice thickness is set to zero and the ice surface temperature is set to the fresh-water freezing point. The mixed-layer temperature is set equal to the water surface temperature from the COSMO sea surface temperature (SST) analysis. If data from the SST analysis are not available, as may be the case for climate runs, an alternative data source should be used to provide a reasonable estimate of the water surface temperature. In any case, an estimate of the water surface temperature for the lake-type grid boxes should be available at the cold start. The mixed-layer thickness is set to 10 m or to the lake depth, whichever is smaller, i.e. mixing down to the lake bottom is assumed for the grid boxes with the lake depth smaller than 10 m. The bottom temperature is set to the temperature of maximum density of fresh water if the lake depth exceeds 10 m and to the mixed-layer temperature otherwise. The shape factor with respect to the temperature profile in the thermocline is set to its minimum value. Finally, the mean temperature of the water column can be computed. Once a cold start is made, FLake runs freely, i.e. without any correction of the FLake variables. Notice that the cold start initialisation should be performed only once for each COSMO-model configuration in terms of the model domain and horizontal resolution. The *warm start* initialisation procedure utilises the values of FLake prognostic variables from the end of the previous run (e.g., this occurs every time a new COSMO-model run is started during the assimilation cycle).

For the non-lake-type grid boxes (ocean/sea and land), the mixed-layer depth is set to zero, the shape factor with respect to the temperature profile in the thermocline is set to its minimum value, and the mixed-layer temperature, the bottom temperature and the mean temperature of the water column are set equal to the temperature of maximum density of fresh water. This setting of the FLake variables is kept during all COSMO-Model runs. Although the setting of the FLake temperatures for the non-lake-type boxes is formal and should not directly affect the model results, the temperatures should never be reset to zero as it results in a two orders of magnitude temperature difference between lake-type and non-lake-type grid boxes and may lead to a loss of accuracy due to GRIB encoding-decoding.

As mentioned in Section 12 of Part II of the COSMO-Model Documentation, snow over lake ice is not considered explicitly. For the lake-type grid boxes, the snow surface temperature is equal to the ice surface temperature and the snow thickness remains zero. For the non-lake-type grid boxes, the initial values of the snow and ice characteristics are specified by the surface analysis procedure. The values of the ice thickness and the ice surface temperature depend on whether the sea ice parameterisation scheme is activated.

A word of caution is appropriate concerning the performance of FLake following a cold start. Since the cold-start values of FLake variables are rather ad hoc, they may not satisfactorily reflect the actual situation in lakes in question. Then, the simulated lake characteristics may fail to agree with observations until the memory of initial conditions is faded. Care should be exercised in the interpretation of model results during the spin-up period. For deep stratified lakes the spin-up period of several months may be required. Experience suggests, however, that a few weeks of spin-up are usually sufficient if the cold start is made during late spring or early summer and a reasonable estimate of the water surface temperature is available.

References

Hess, R. (2001). Assimilation of screen-level observations by variational soil moisture analysis. *Meteorol. Atmos. Phys. 77*, 155–166.

Jacobsen, I. and E. Heise (1982). A new economic method for the computation of the surface temperature in numerical models. *Contr. Atmos. Phys. 55*, 128–141.

Kinne, S., D. O'Donnel, P. Stier, S. Kloster, K. Zhang, H. Schmidt, S. Rast, M. Giorgetta, T. F. Eck, and B. Stevens (2013). MAC-v1: A new global aerosol climatology for climate studies. *J. Adv. Model. Earth Sy. 5*(4), 704–740.

Leuenberger, D., M. Koller, and C. Schär (2010). An improved formulation of the SLEVE coordinate. *Mon. Wea. Rev. 138*(9), 3683–3689.

Louis, J.-F. (1979). A parametric model of vertical eddy fluxes in the atmosphere. *Bound. Layer Meteor. 17*, 187–202.

Lynch, P., D. Girard, and V. Ivanovici (1997). Improving the efficiency of a digital filtering scheme. *Mon. Wea. Rev. 125*, 1976–1982.

Mironov, D. (2008). Parameterization of lakes in numerical weather prediction. Description of a lake model. COSMO Technical Report 11, Consortium for Small-Scale Modelling, 41 pp. doi:10.5676/DWD_pub/nwv/cosmo-tr_11 (available from http://www.cosmo-model.org).

Mironov, D. and B. Ritter (2004). Testing the new ice model for the global NWP system GME of the German Weather Service. Technical documentation, 1220, WMO, WMO, Geneve, Switzerland.

Raymond, W. H. (1988). High-Order Low-Pass Implicit Tangent Filters for Use in Finite Area Calculations. *Mon. Wea. Rev. 116*, 2132–2141.

Rieger, D., M. Bangert, I. Bischoff-Gauss, J. Förstner, K. Lundgren, D. Reinert, J. Schröter, H. Vogel, G. Zängl, R. Ruhnke, and B. Vogel (2015). ICON-ART 1.0 - a new online-coupled model system from the global to regional scale. *Geosci. Model Dev. 8*(6), 1659–1676.

Rieger, D., A. Steiner, V. Bachmann, P. Gasch, J. Förstner, K. Deetz, B. Vogel, and H. Vogel (2017). Impact of the 4 April 2014 Saharan dust outbreak on the photovoltaic power generation in Germany. *Atmos. Chem. Phys. 17*(21), 13391.

Ritter, B. and J.-F. Geleyn (1992). A comprehensive radiation scheme for numerical weather prediction models with potential applications in climate simulations. *Mon. Wea. Rev. 120*, 303–325.

Schär, C., D. Leuenberger, O. Fuhrer, D. Lüthi, and C. Girard (2002). A new terrain-following vertical coordinate formulation for atmospheric prediction models. *Mon. Wea. Rev. 130*, 2459–2480.

Schraff, C. (1996). Data assimilation and mesoscale weather prediction: A study with a forecast model for the Alpine region. Publication 56, Swiss Meteorological Institute, Zürich.

Schraff, C. (1997). Mesoscale data assimilation and prediction of low stratus in the Alpine region. *Meteorol. Atmos. Phys. 64*, 21–50.

Schraff, C., H. Reich, A. Schomburg, K. Stephan, A. Periáñez, and R. Potthast (2016). Kilometre-scale ensemble data assimilation for the COSMO model (KENDA). *Q. J. R. Meteorol. Soc. 142*, 1453–1472.

Skamarock, W. and J. Klemp (1992). The stability of time-split numerical methods for the hydrostatic and nonhydrostatic elasic equations. *Mon. Wea. Rev. 120*, 2109–2127.

Stephan, K., S. Klink, and C. Schraff (2008). Assimilation of radar-derived rain rates into the convective-scale model COSMO-DE at DWD. *Quart. J. R. Meteor. Soc. 134*, 1315–1326.

Tegen, I., P. Hollrig, M. C. I. Fung, D. Jacob, and J. Penner (1997). Contribution of different aerosol species to the global aerosol extinction optical thickness: Estimates from model results. *J. Geophys. Res. 102*, 23895–23915.

Thomas, S., C. Girard, G. Doms, and U. Schättler (2000). Semi-Implicit Scheme for the DWD Lokal-Modell. *Meteorol. Atmos. Phys. 73*, 105–125.

Tiedtke, M. (1989). A comprehensive mass flux scheme for cumulus parameterization in large-scale models. *Mon. Wea. Rev. 117*, 1779–1799.

Wicker, L. and W. Skamarock (2002). Time-splitting methods for elastic models using forward time schemes. *Mon. Wea. Rev. 130*, 2088–2097.