SPONSORED BY THE
Federal Ministry
of Education
and Research

PALM-4U Crashcourse

F. Kanani-Sühring

Hints for simulation evaluation

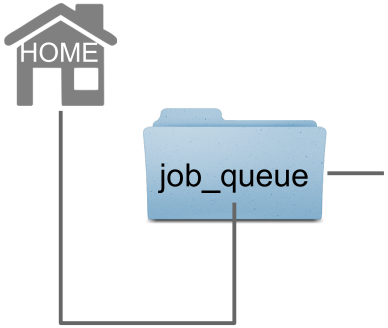
Content

- Overview of output files
- PALM output data standard
- Simulation monitoring
- Evaluation of simulation results

Overview of output files (I)

PALM job log:

- Created before and updated during run-time
- Summary of palmrun configuration & steps
- Information about simulation progress & data output
- Log output in UNIX terminal for interactive run (job start on local PC), output to file for batch run (job submission to queuing system)

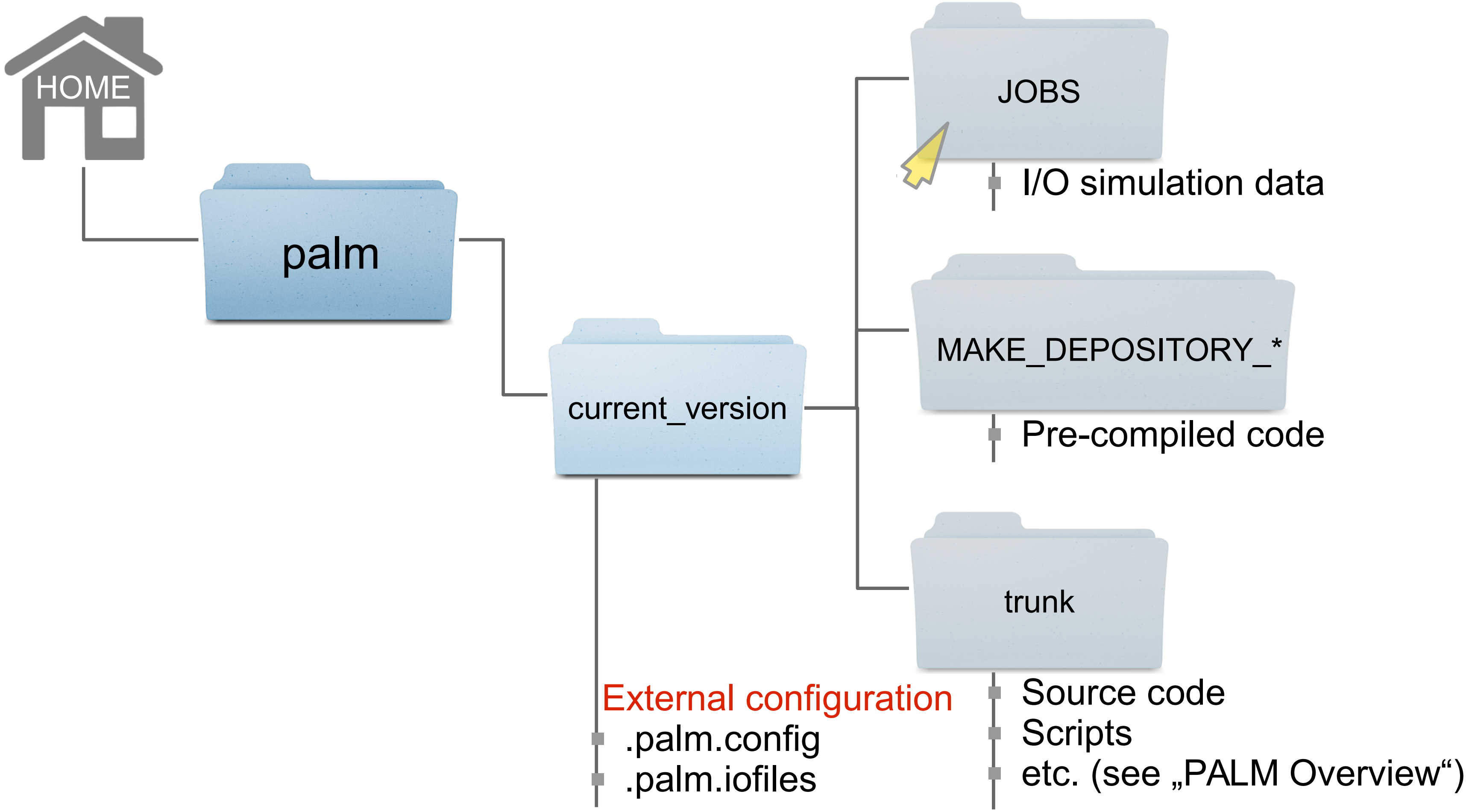


- <configuration>_<jobname>.<cyclenumber>

ASCII-Output

Overview of output files (II)

Flashback: palm directory structure



Internal steering, job monitoring & data output

- I/O simulation data
- Pre-compiled code

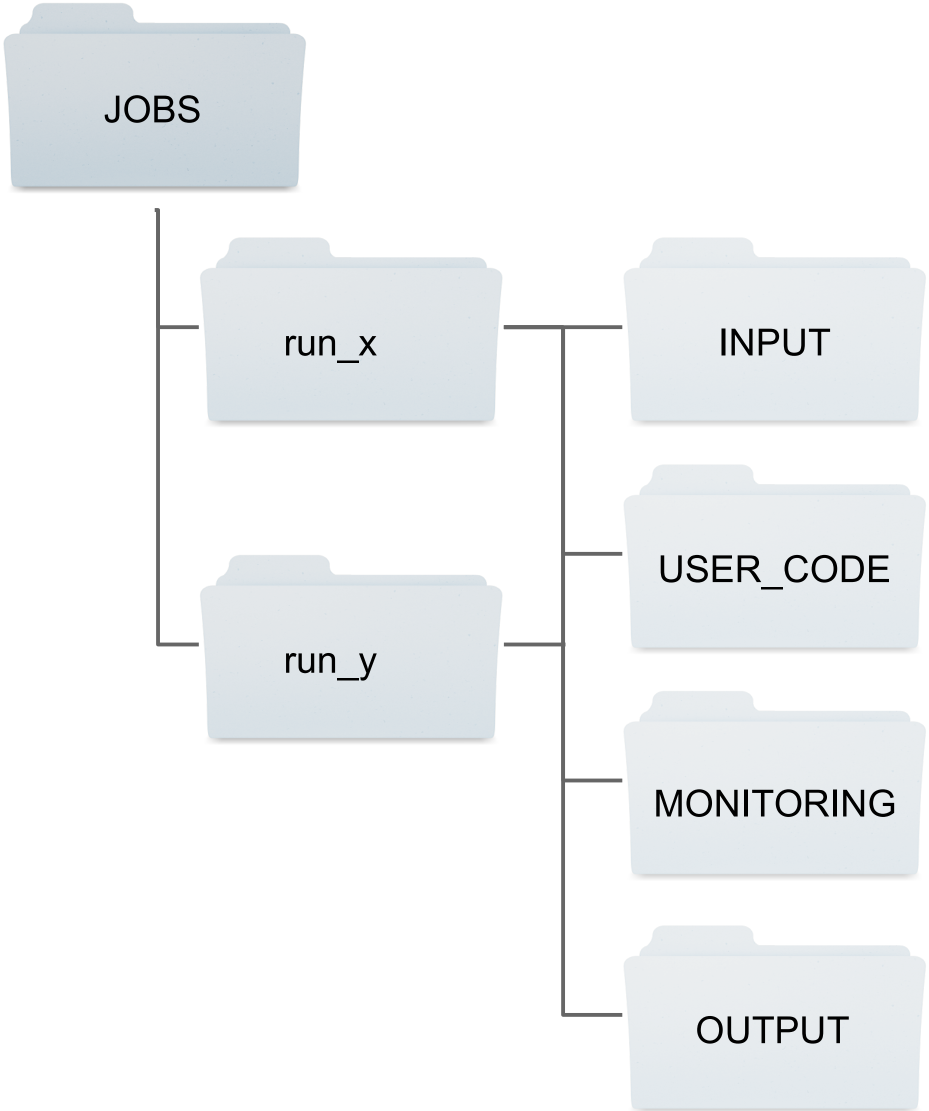
External configuration

- .palm.config
- .palm.iofiles

- Source code
- Scripts
- etc. (see „PALM Overview“)

Overview of output files (III)

🟨 ASCII-Input 🟦 ASCII-Output 🟩 NetCDF



JOBS

run_x

run_y

INPUT

USER_CODE

MONITORING

OUTPUT

- Parameter file for steering
- Other input files (e.g. topography data)

- User-defined source code

- Header file (parameter settings)
- Run control file (parameter settings, timestep information)
- CPU file (computing time measurements)

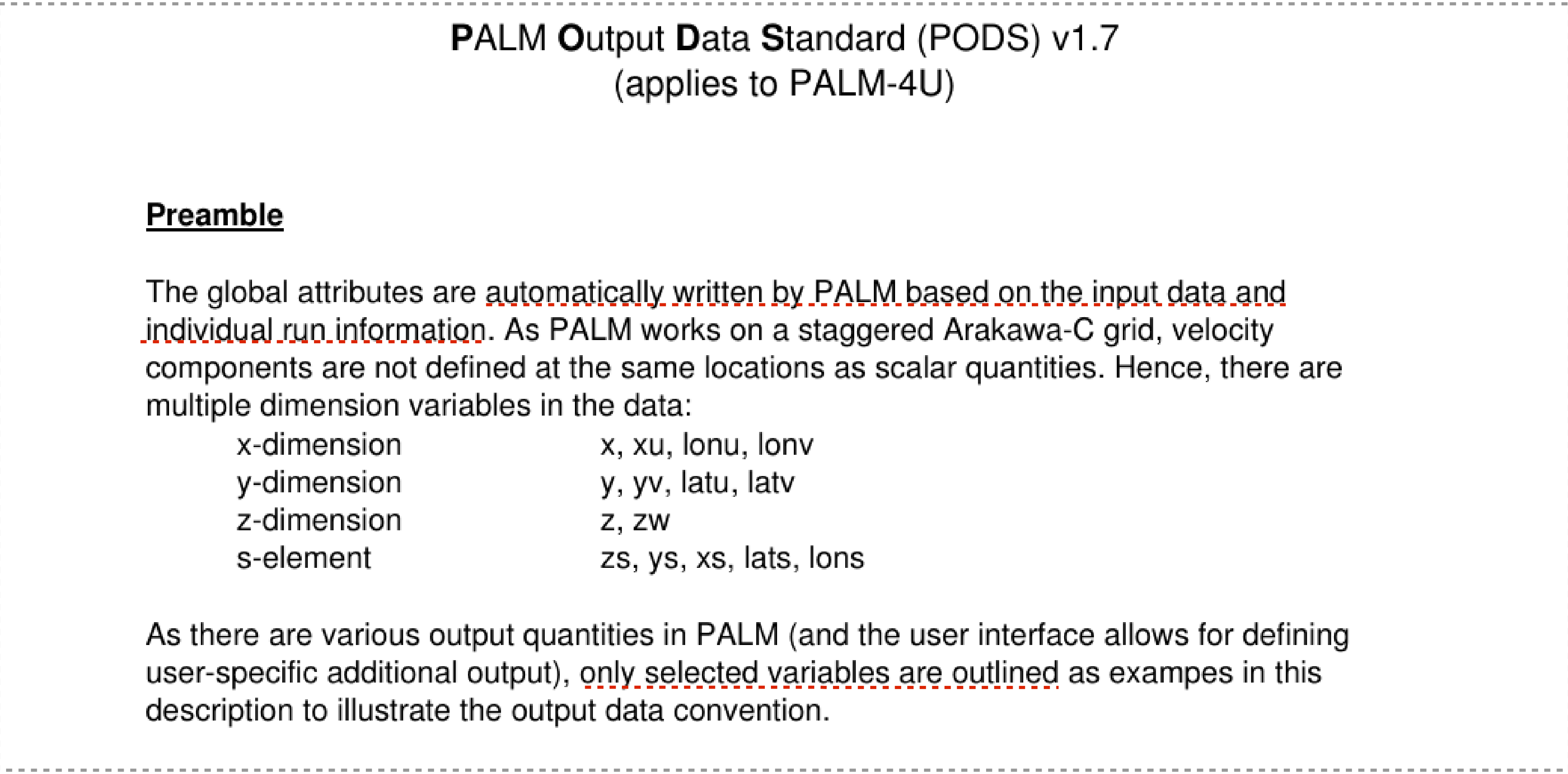
- 1D profiles (domain- & time-averaged)
- 2D sections & 3D data (instantaneous & time-averaged)

Overview of output files (IV)

- MONITORING files (ASCII)
- OUTPUT files (NetCDF)

palmrun takes care of file-copy process, as defined in configuration files .palm.config.<ci> and .palm.iofiles

PALM output data standard (I)



PALM Output Data Standard (PODS) v1.7 (applies to PALM-4U)

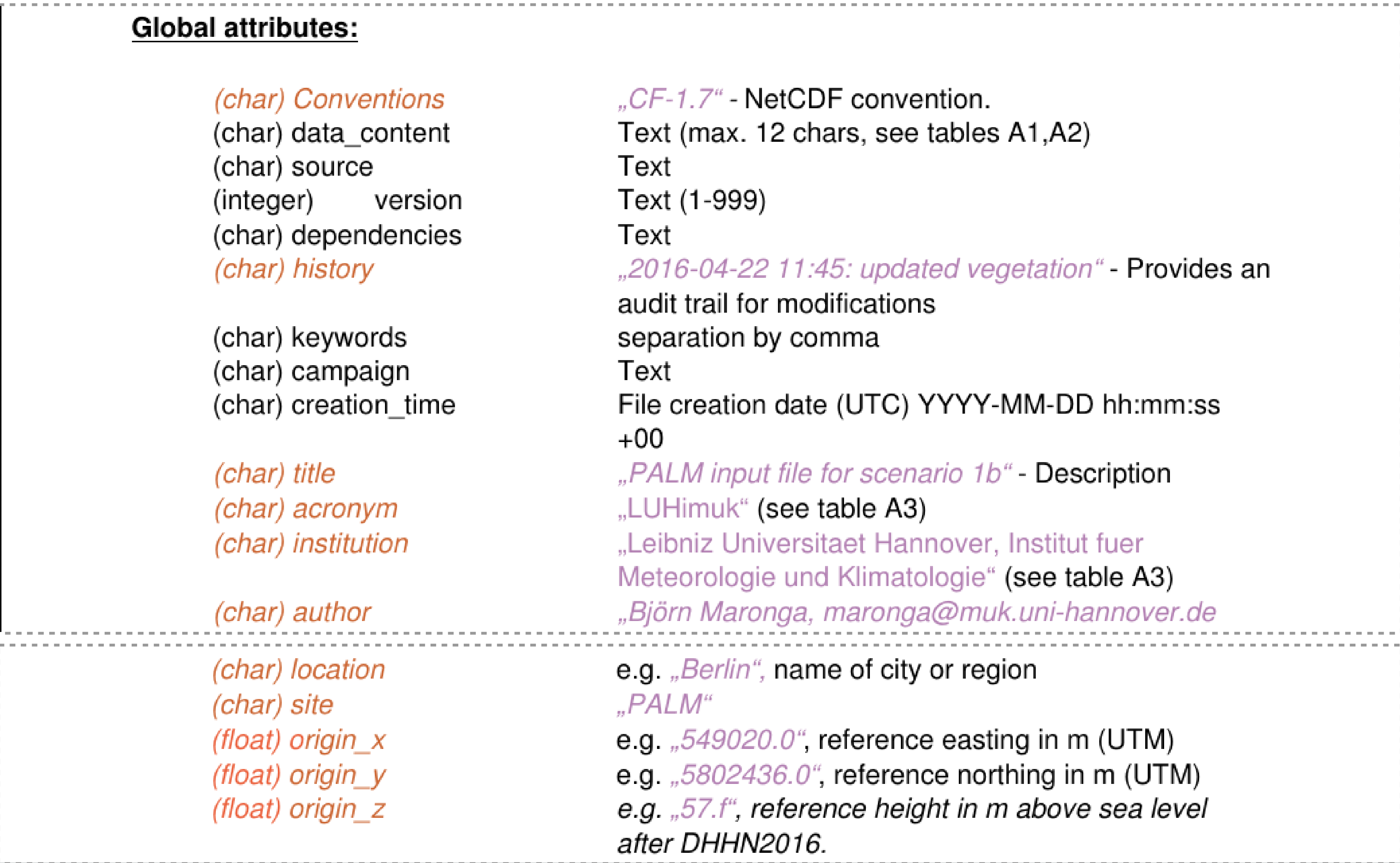
Preamble

The global attributes are automatically written by PALM based on the input data and individual run information. As PALM works on a staggered Arakawa-C grid, velocity components are not defined at the same locations as scalar quantities. Hence, there are multiple dimension variables in the data:

x-dimension	x, xu, lonu, lonv
y-dimension	y, yv, latu, latv
z-dimension	z, zw
s-element	zs, ys, xs, lats, lons

As there are various output quantities in PALM (and the user interface allows for defining user-specific additional output), only selected variables are outlined as exampes in this description to illustrate the output data convention.

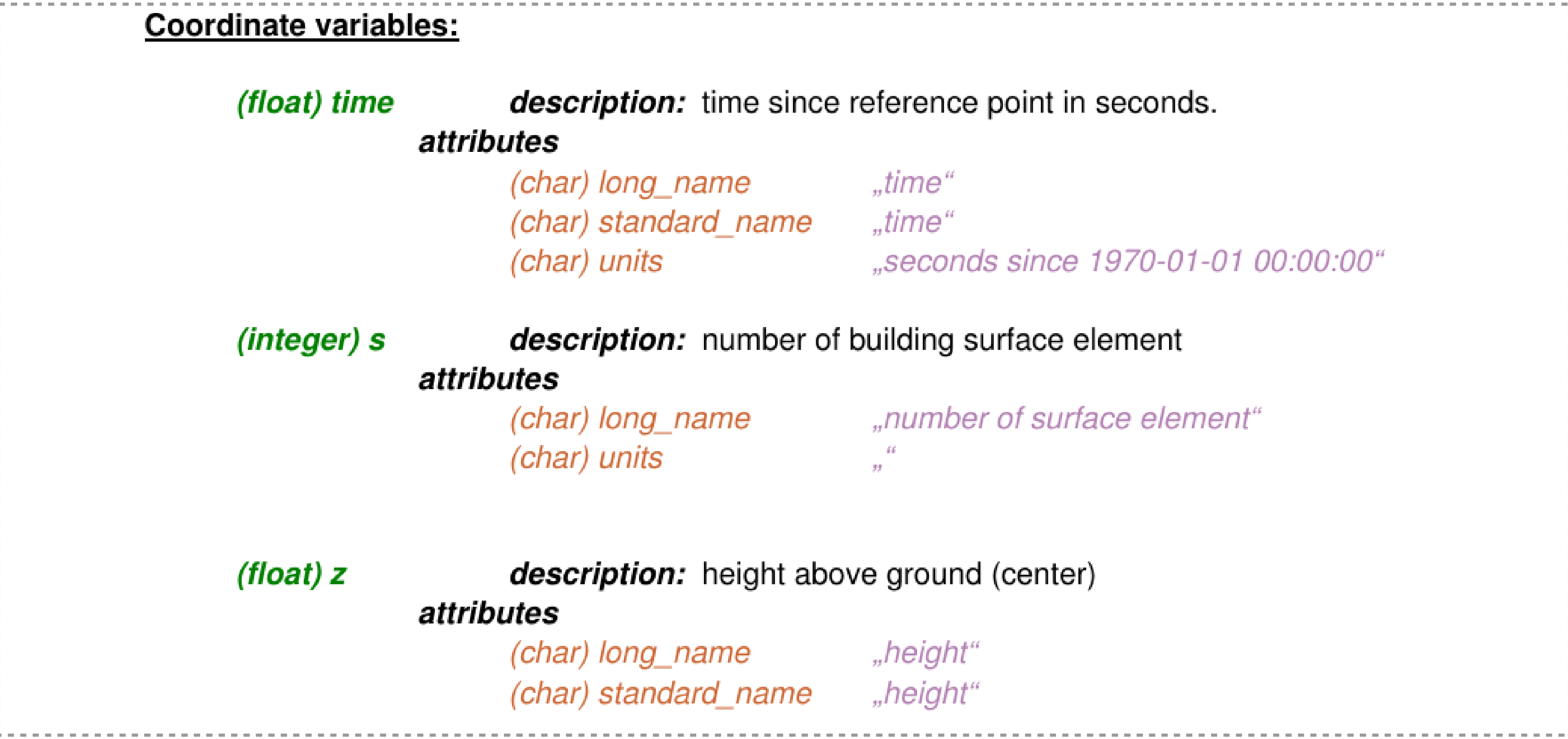
PALM output data standard (II)



Global attributes:

(char) Conventions	*„CF-1.7“* - NetCDF convention.
(char) data_content	Text (max. 12 chars, see tables A1,A2)
(char) source	Text
(integer) version	Text (1-999)
(char) dependencies	Text
(char) history	*„2016-04-22 11:45: updated vegetation“* - Provides an audit trail for modifications
(char) keywords	separation by comma
(char) campaign	Text
(char) creation_time	File creation date (UTC) YYYY-MM-DD hh:mm:ss +00
(char) title	*„PALM input file for scenario 1b“* - Description
(char) acronym	„LUHimuk“ (see table A3)
(char) institution	„Leibniz Universitaet Hannover, Institut fuer Meteorologie und Klimatologie“ (see table A3)
(char) author	*„Björn Maronga, maronga@muk.uni-hannover.de*
(char) location	e.g. *„Berlin“,* name of city or region
(char) site	*„PALM“*
(float) origin_x	e.g. *„549020.0“*, reference easting in m (UTM)
(float) origin_y	e.g. *„5802436.0“*, reference northing in m (UTM)
(float) origin_z	*e.g. „57.f“, reference height in m above sea level after DHHN2016.*

PALM output data standard (III)



Coordinate variables:

(float) time **description:** time since reference point in seconds.

attributes

(char) long_name *„time“*
(char) standard_name *„time“*
(char) units *„seconds since 1970-01-01 00:00:00“*

(integer) s **description:** number of building surface element

attributes

(char) long_name *„number of surface element“*
(char) units *„“*

(float) z **description:** height above ground (center)

attributes

(char) long_name *„height“*
(char) standard_name *„height“*

PALM output data standard (IV)

3D data:

pt (time, z, y, x)

description: potential temperature

type: float

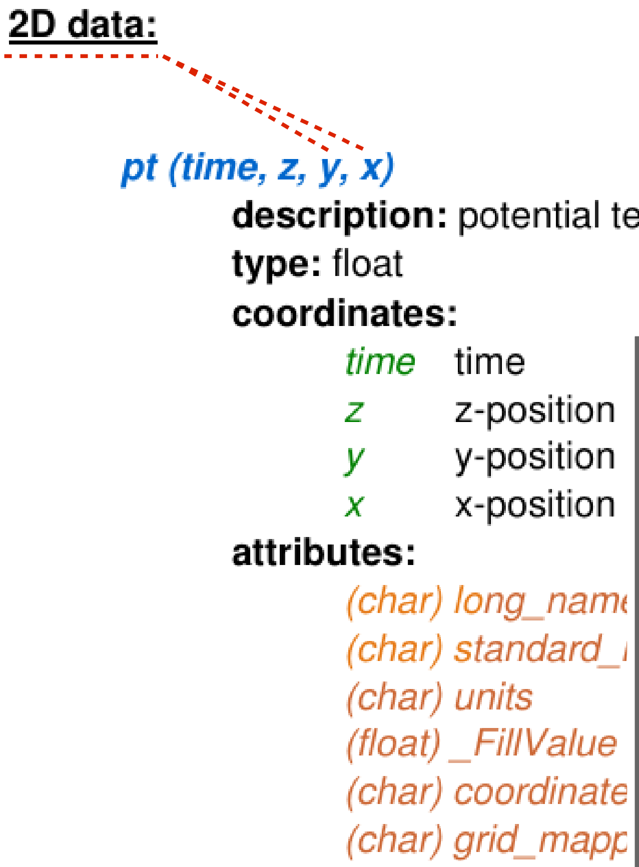
coordinates:

time	time
z	z-position (in m)
y	y-position (in m)
x	x-position (in m)

attributes:

(char) long_name	*„pt“*
(char) standard_name	*„air_potential_temperature“*
(char) units =	*„K“*
(float) _FillValue	= *-9999.9f*
(char) coordinates	= *„E_UTM N_UTM lon lat“*
(char) grid_mapping	= *„crsUTM: E_UTM N_UTM crsETRS: lon lat“*

PALM output data standard (V)



2D data:

pt (time, z, y, x)

description: potential temperature
type: float
coordinates:

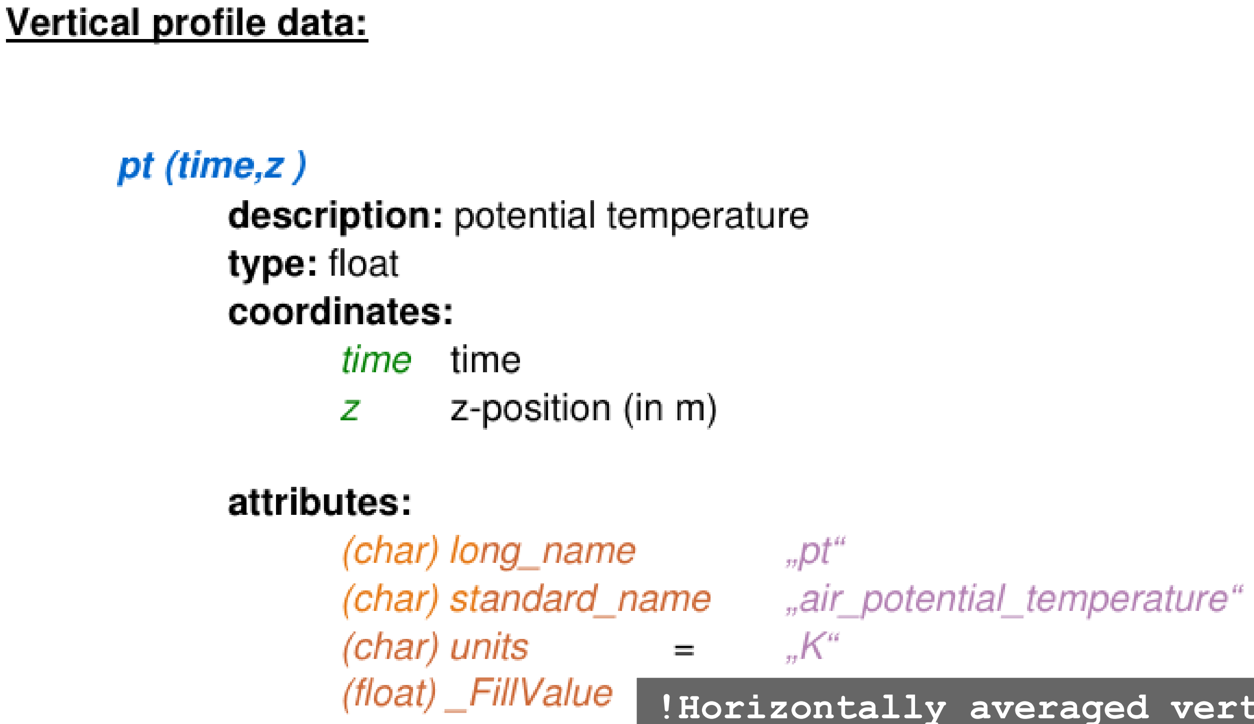
time	time
z	z-position
y	y-position
x	x-position

attributes:

(char) long_name
*(char) standard_*
(char) units
(float) _FillValue
(char) coordinate
(char) grid_mapp

```
!2D/3D data output
do3d_at_begin              = .T.,
dt_data_output             =  600.0,
dt_data_output_av          = 3600.0,
averaging_interval         = 3600.0,

do2d_at_begin              = .T.,
dt_do2d_xy                 = 60.0,
section_xy                 = 0, 1, 2, 3, 5, 10,

data_output = 'u',     'v',     'w',
              'u_av', 'v_av', 'w_av',
              'm_soil_av', 't_soil_av',
              'pt_xy', 'q_xy',
              'pt_xy_av', 'q_xy_av',
```

PALM output data standard (VI)



Vertical profile data:

pt (time,z)

description: potential temperature

type: float

coordinates:

time	time
z	z-position (in m)

attributes:

(char) long_name		*„pt“*
(char) standard_name		*„air_potential_temperature“*
(char) units	=	*„K“*
(float) _FillValue		

```
!Horizontally averaged vertical profiles
dt_dopr                      =   600.0,
averaging_interval_pr        =   600.0,
dt_averaging_input_pr        =   0.0,

data_output_pr = '#u', 'w"u"', 'w*u*', 'wu',
                 'w', 'w*2', 'e', 'e*',
                 'pt',
```

PALM output data standard (VII)

Timeseries data:

```
!Timeseries of domain-averaged or maximum values
!(automatic output of all available quantities)
!-----------------------------------------------
dt_dots                        =    1.0,
```

ol (time)

description: Okukhov length
type: float

Surface data:

```
UNDER DEVELOPMENT
```

t0 (s)

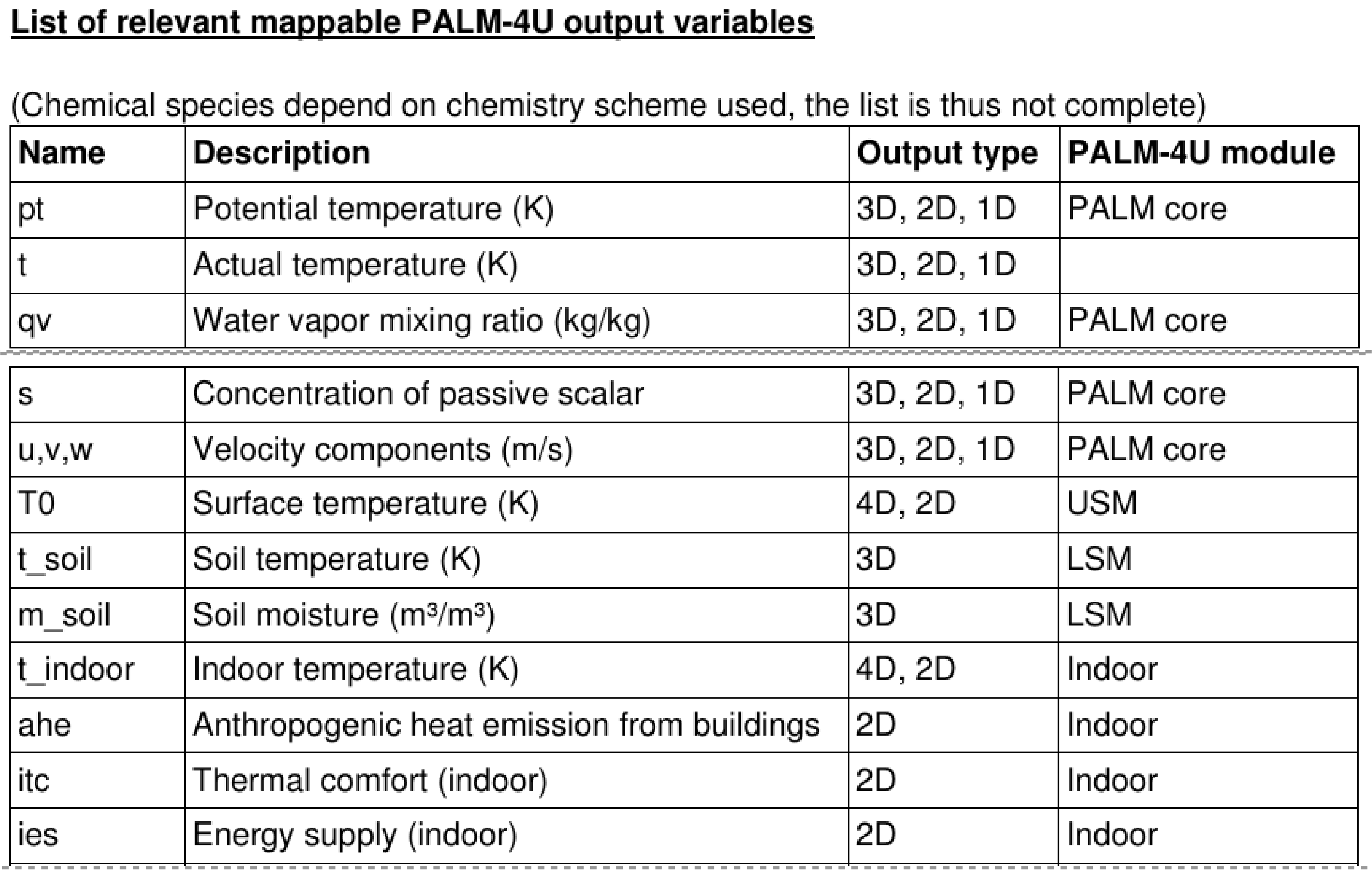
description: Surface temperature
type: float
coordinates:

s	number of surface element

attributes:

long_name		*„Surface temperature“*
standard_name		*„surface_temperature“*
units	=	*„m“*
(char) coordinates	=	*„E_UTM N_UTM lon lat“*
(char) grid_mapping	=	*„crsUTM: E_UTM N_UTM crsETRS: lon lat“*
(float) _FillValue	=	*-9999.9f*

PALM output data standard (VIII-a)

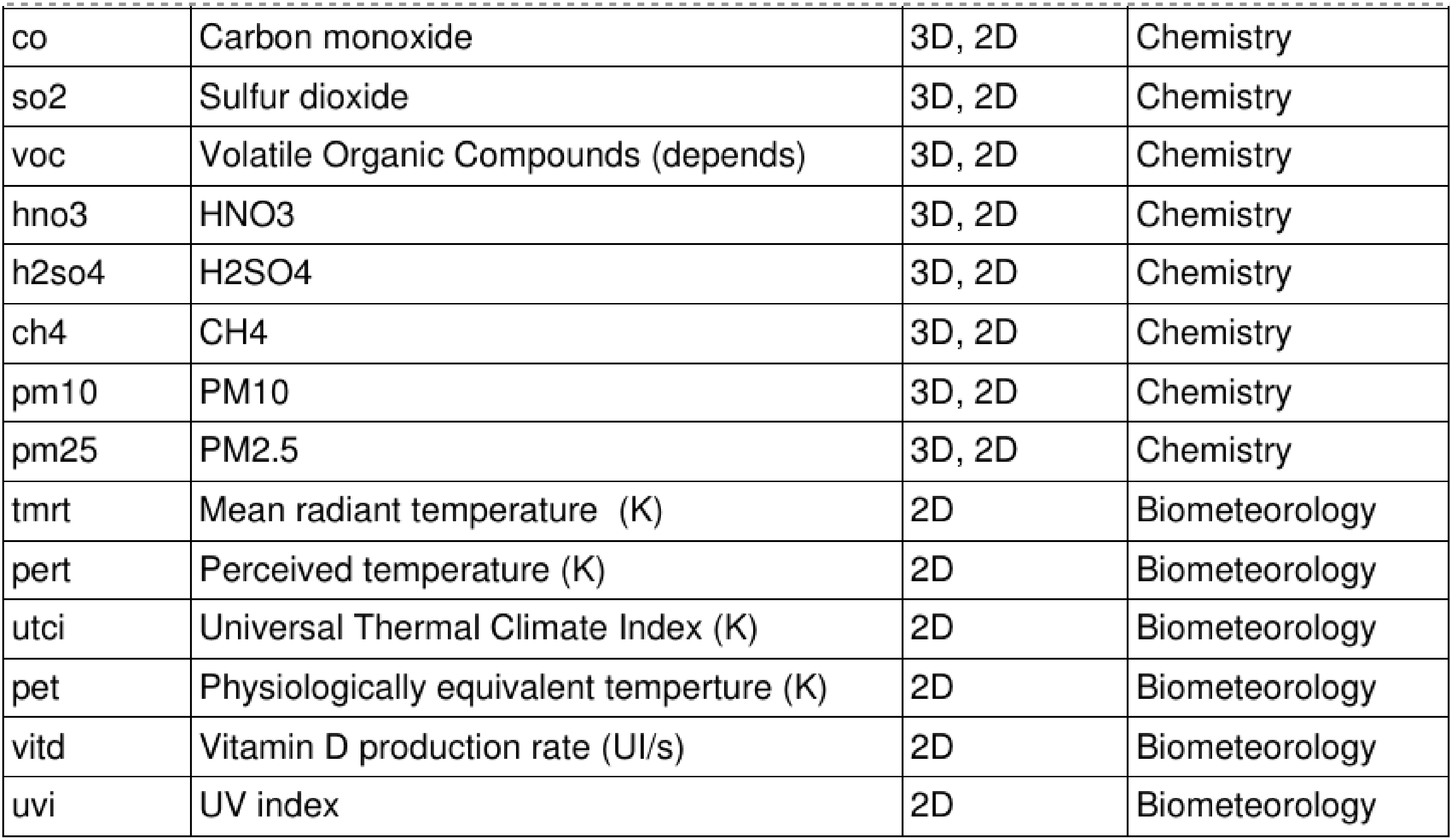


List of relevant mappable PALM-4U output variables

(Chemical species depend on chemistry scheme used, the list is thus not complete)

Name	Description	Output type	PALM-4U module
pt	Potential temperature (K)	3D, 2D, 1D	PALM core
t	Actual temperature (K)	3D, 2D, 1D	
qv	Water vapor mixing ratio (kg/kg)	3D, 2D, 1D	PALM core
s	Concentration of passive scalar	3D, 2D, 1D	PALM core
u,v,w	Velocity components (m/s)	3D, 2D, 1D	PALM core
T0	Surface temperature (K)	4D, 2D	USM
t_soil	Soil temperature (K)	3D	LSM
m_soil	Soil moisture (m^3/m^3)	3D	LSM
t_indoor	Indoor temperature (K)	4D, 2D	Indoor
ahe	Anthropogenic heat emission from buildings	2D	Indoor
itc	Thermal comfort (indoor)	2D	Indoor
ies	Energy supply (indoor)	2D	Indoor

PALM output data standard (VIII-b)



co	Carbon monoxide	3D, 2D	Chemistry
so2	Sulfur dioxide	3D, 2D	Chemistry
voc	Volatile Organic Compounds (depends)	3D, 2D	Chemistry
hno3	HNO3	3D, 2D	Chemistry
h2so4	H2SO4	3D, 2D	Chemistry
ch4	CH4	3D, 2D	Chemistry
pm10	PM10	3D, 2D	Chemistry
pm25	PM2.5	3D, 2D	Chemistry
tmrt	Mean radiant temperature (K)	2D	Biometeorology
pert	Perceived temperature (K)	2D	Biometeorology
utci	Universal Thermal Climate Index (K)	2D	Biometeorology
pet	Physiologically equivalent temperture (K)	2D	Biometeorology
vitd	Vitamin D production rate (UI/s)	2D	Biometeorology
uvi	UV index	2D	Biometeorology

... and many more (for land surface, radiation, plants, etc.)

Simulation monitoring (I-a)

PALM job log (part 1, jobname = example_cbl)

```
********************************************************************************
* Batch Job Information:                                                       *
********************************************************************************
Moab ID:                hannover.1312200
Torque ID:              1312982.hbatch.hsn.hlrn.de
Job Name:               example_cbl.630
HLRN Complex:           hannover
User Name:              nikfarah
HLRN Project Account:   nikfarah
Batch Class:            mpp2testq
Number of Nodes:        1
Node List:              h01718
Start Time:             Fr 27. Okt 14:46:53 CEST 2017
MOM Node:               hxcmom03
Submission Directory:   /home/h/nikfarah/job_queue
********************************************************************************

********************************************************************************
*                                                                              *
*                 Welcome to HLRN-III Complex Hannover                         *
*                 This is node hxcmom03 on "Gottfried"                         *
*                                                                              *
* Documentation:  https://www.hlrn.de/home/view/Service/WebHome                *
* Support:        mailto:support@hlrn.de                                       *
*                                                                              *
********************************************************************************
```

Info about batch system and remote computer (only for batch run)

Simulation monitoring (I-b)

PALM job log (part 2)

```
+ palmrun -d example_cbl -h default -m 1500 -t 600 -q mpp2testq -r 630 -U kanani -a
'd3#' -G 'Rev: 2578:2585M' -X 16 -T 16 -j -u nikfarah -R 130.75.105.106

*** palmrun  1.0 Rev: 2303 $
    will be executed.     Please wait ...

    Reading the configuration file...
    Reading the I/O files...
```

palmrun command (green set by user, other options filled by palmrun)

palmrun version/revision (orange), not to be confused with PALM version/revision

Simulation monitoring (I-c)

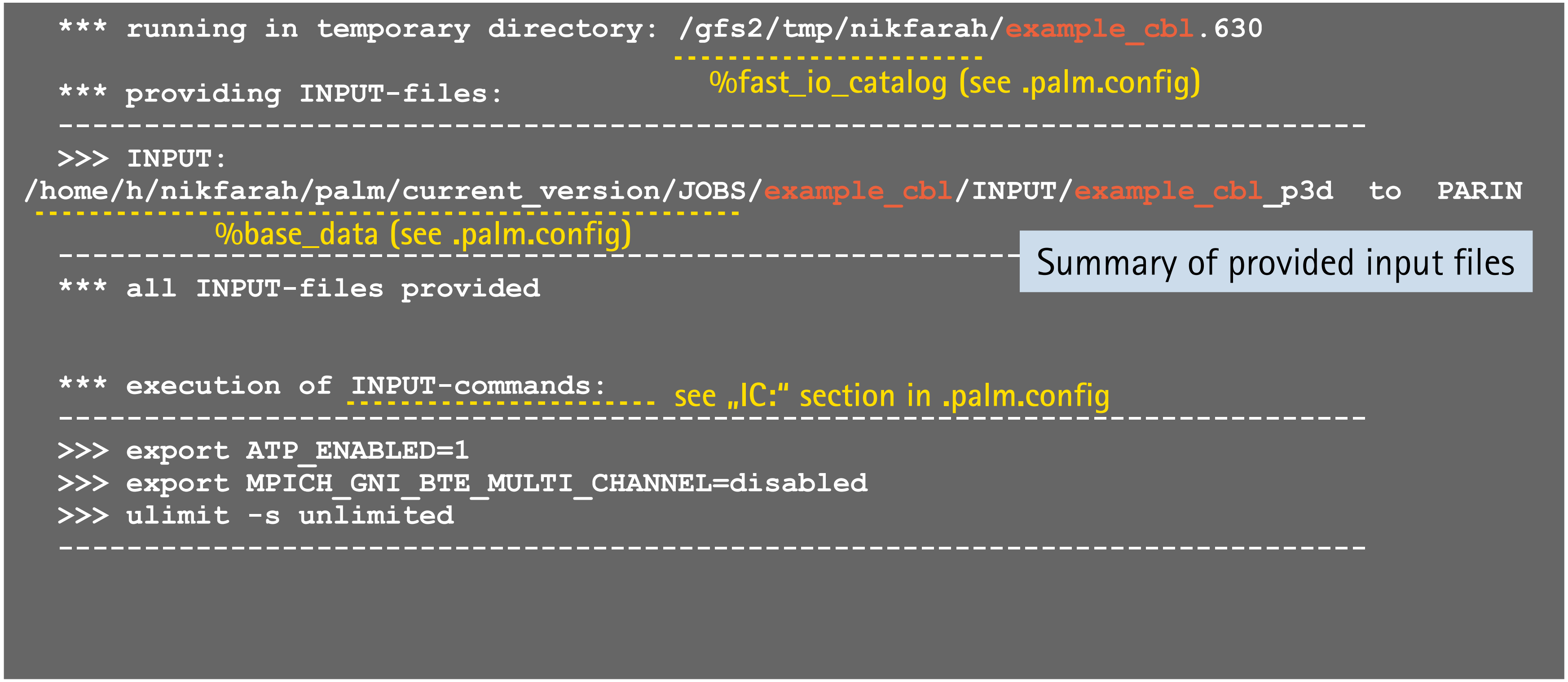
PALM job log (part 3)

```
#------------------------------------------------------------------------#
| palmrun  1.0 Rev: 2303 $                 Fri Oct 27 14:47:07 CEST 2017 |
|                                                                        |
| called on:                hxcmom03                                     |
| execution on:             default (IP:130.75.4.1)                      |
| running in:               batch job mode                               |
| number of cores:          16                                           |
| tasks per node:           16 (number of nodes: 1)                      |
| memory demand / PE:       1500 MB                                      |
| job cpu time (h:m:s):     0:10:0                                       |
|                                                                        |
| cpp directives:           -e Z -DMPI_REAL=MPI_DOUBLE_PRECISION -DMPI_2R |
|                           EAL=MPI_2DOUBLE_PRECISION -D__parallel -D__ne |
|                           tcdf -D__netcdf4 -D__netcdf4_parallel -D__fft |
|                           w                                            |
| compiler options:         -em -O3 -hnoomp -hnoacc -hfp3 -hdynamic      |
| linker options:           -em -O3 -hnoomp -hnoacc -hfp3 -hdynamic -dyna |
|                           mic                                          |
| login init commands:      module unload craype-ivybridge cray-netcdf   |
| module commands:          module load fftw cray-hdf5-parallel cray-netc |
|                           df-hdf5parallel craype-haswell               |
|                                                                        |
| base name of files:       example_cbl                                  |
| activation string list:   d3#                                          |
#------------------------------------------------------------------------#
```

Summary of configuration settings in .palm.config

Simulation monitoring (I-d)

PALM job log (part 4)



```
   *** running in temporary directory: /gfs2/tmp/nikfarah/example_cbl.630

   *** providing INPUT-files:
   ----------------------------------------------------------------------------
   >>> INPUT:
 /home/h/nikfarah/palm/current_version/JOBS/example_cbl/INPUT/example_cbl_p3d  to  PARIN
   ----------------------------------------------------------------------------
   *** all INPUT-files provided


   *** execution of INPUT-commands:
   ----------------------------------------------------------------------------
   >>> export ATP_ENABLED=1
   >>> export MPICH_GNI_BTE_MULTI_CHANNEL=disabled
   >>> ulimit -s unlimited
   ----------------------------------------------------------------------------
```

%fast_io_catalog (see .palm.config)

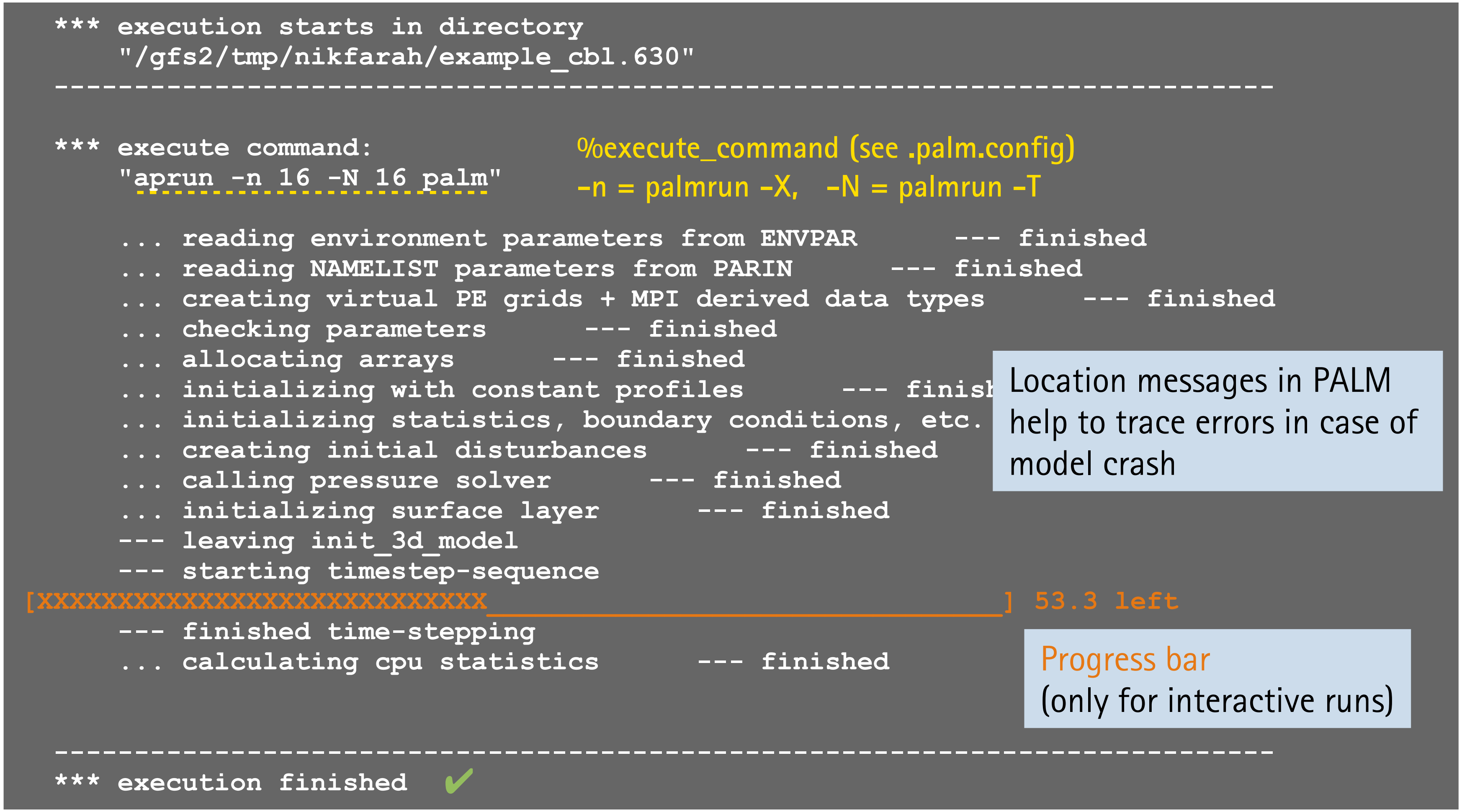
%base_data (see .palm.config)

Summary of provided input files

see „IC:" section in .palm.config

Simulation monitoring (I-e)

PALM job log (part 5)



```
  *** execution starts in directory
      "/gfs2/tmp/nikfarah/example_cbl.630"
  ----------------------------------------------------------------------------

  *** execute command:
      "aprun -n 16 -N 16 palm"

      ... reading environment parameters from ENVPAR      --- finished
      ... reading NAMELIST parameters from PARIN      --- finished
      ... creating virtual PE grids + MPI derived data types      --- finished
      ... checking parameters      --- finished
      ... allocating arrays      --- finished
      ... initializing with constant profiles      --- finish
      ... initializing statistics, boundary conditions, etc.
      ... creating initial disturbances      --- finished
      ... calling pressure solver      --- finished
      ... initializing surface layer      --- finished
      --- leaving init_3d_model
      --- starting timestep-sequence
[XXXXXXXXXXXXXXXXXXXXXXXXXXXX________________________________] 53.3 left
      --- finished time-stepping
      ... calculating cpu statistics      --- finished


  ----------------------------------------------------------------------------
  *** execution finished ✓
```

%execute_command (see .palm.config)

-n = palmrun -X,   -N = palmrun -T

Location messages in PALM help to trace errors in case of model crash

Progress bar
(only for interactive runs)

Simulation monitoring (I-f)

PALM job log (part 6)

```
*** post-processing: now executing "aprun -n 1 -N 1 combine_plot_fields.x" ...

*** combine_plot_fields ***
    uncoupled run


    NetCDF output enabled
    XY-section:   16  file(s) found
                224 array(s) processed
    Required cpu-time:     0.000 sec

    no XZ-section data available

    no YZ-section data available

    NetCDF output enabled
    3D-data:      16  file(s) found
                264 array(s) processed
                 12 array(s) are time-averaged
    Required cpu-time:     0.125 sec
```

Depending on chosen netcdf_data_format (d3par namelist parameter), i.e. in case of non-parallel output, 2D/3D data from each model subdomain (1 file per subdomain) have to be combined after the run. A 1-core job is started for that purpose.

Simulation monitoring (I-g)

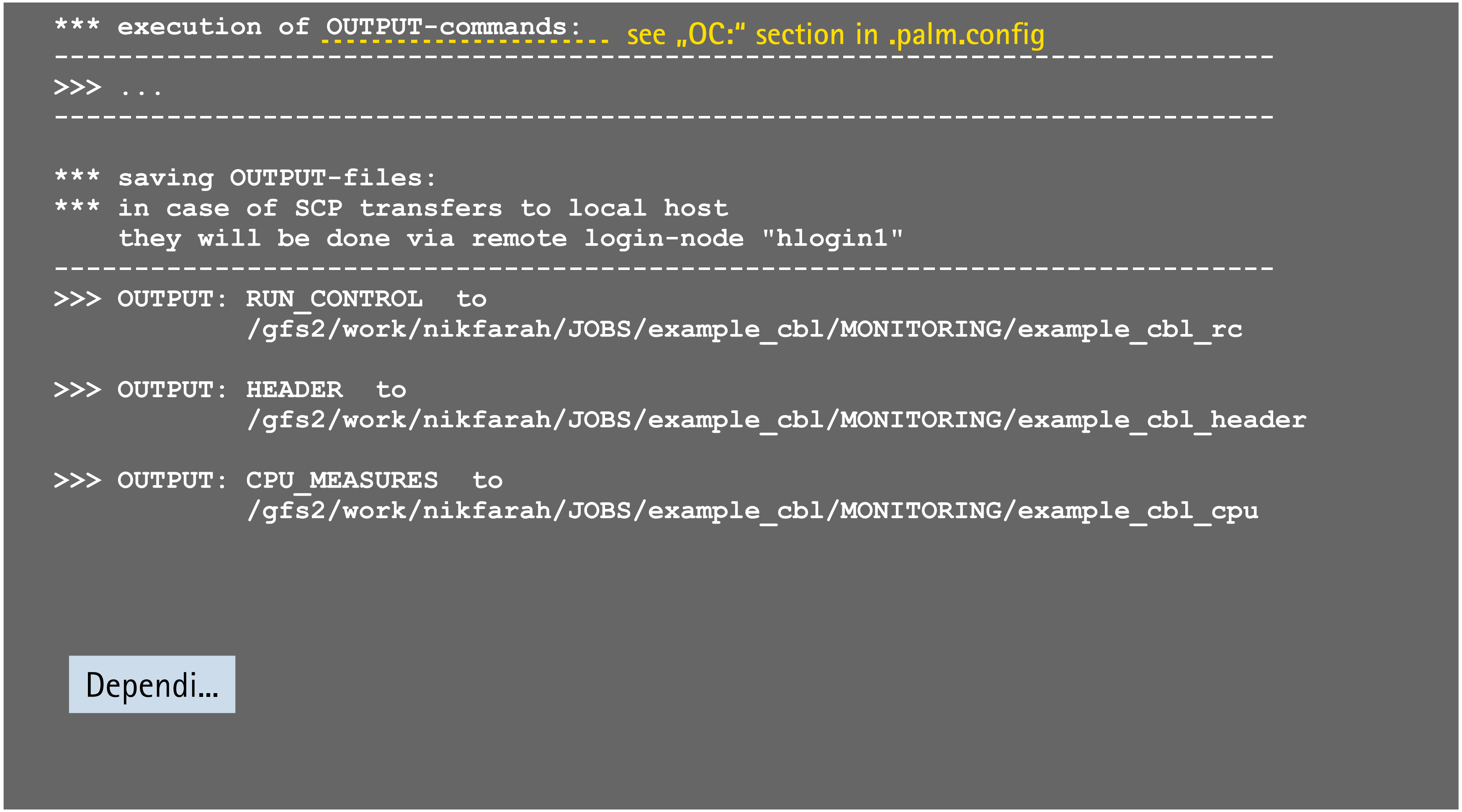
PALM job log (part 7)

```
*** post-processing: now executing "aprun -n 1 -N 1 combine_plot_fields.x" ...

*** combine_plot_fields ***
    uncoupled run


    NetCDF output enabled
    XY-section:   16  file(s) found
                 224 array(s) processed
    Required cpu-time:      0.000 sec

    no XZ-section data available

    no YZ-section data available

    NetCDF output enabled
    3D-data:      16  file(s) found
                 264 array(s) processed
                  12 array(s) are time-averaged
    Required cpu-time:      0.125 sec
```

Depending on chosen netcdf_data_format (d3par namelist parameter), i.e. in case of non-parallel output, 2D/3D data from each model subdomain (1 file per subdomain) have to be combined after the run. A 1-core job is started for that purpose.

Simulation monitoring (I-h)

PALM job log (part 8)



```
*** execution of OUTPUT-commands:
---------------------------------------------------------------------------
>>> ...
---------------------------------------------------------------------------

*** saving OUTPUT-files:
*** in case of SCP transfers to local host
    they will be done via remote login-node "hlogin1"
---------------------------------------------------------------------------
>>> OUTPUT: RUN_CONTROL  to
            /gfs2/work/nikfarah/JOBS/example_cbl/MONITORING/example_cbl_rc

>>> OUTPUT: HEADER  to
            /gfs2/work/nikfarah/JOBS/example_cbl/MONITORING/example_cbl_header

>>> OUTPUT: CPU_MEASURES  to
            /gfs2/work/nikfarah/JOBS/example_cbl/MONITORING/example_cbl_cpu
```

see „OC:“ section in .palm.config

Simulation monitoring (I-i)

PALM job log (part 9)

```
   >>> OUTPUT: DATA_2D_XY_NETCDF  to
               /gfs2/work/nikfarah/JOBS/example_cbl/OUTPUT/example_cbl_xy.nc

   >>> OUTPUT: DATA_3D_NETCDF  to
               /gfs2/work/nikfarah/JOBS/example_cbl/OUTPUT/example_cbl_xy.nc

   +++ temporary OUTPUT-file  DATA_2D_XY_AV_NETCDF  does not exist


   ----------------------------------------------------------------------------
   *** all OUTPUT-files saved

  --> palmrun finished   ✓
```

If you find something like this, something went wrong :-(

```
   ----------------------------------------------------------------------------
   ~~*** execution finished~~    missing „execution finished“     ✗
```

```
   +++ runtime error occured
   *** Execution of ERROR-command:
   >>> [[ normal = execution ]] && cat RUN_CONTROL


 +++ palmrun killed     ✗
```

Simulation monitoring (I–j)

PALM job log: What to do in case of problems

- Job log provides informative, warning and error messages

```
--- informative message ---  ID: PA0430   generated by routine: netcdf_data_input_mod

    File PIDS_STATIC does not exist - input from external files is read from separate
    ASCII files, if required.

    Further information can be found at
    http://palm.muk.uni-hannover.de/trac/wiki/doc/app/errmsg#PA0430
```

```
+++ error message ---  ID: PA0227   generated by routine: init_pegrid

    y-direction: gridpoint number ( 240 ) is not an
    integral divisor of the number ofprocessors ( 22 )

    Further information can be found at
    http://palm.muk.uni-hannover.de/trac/wiki/doc/app/errmsg#PA0227
```

- Got none of that? You need to do debugging if you can. (that's another story)
  - Are you working with user-defined code? That might be the source of error.
- No clue what to do? Or found a bug?
  → Write us a ticket with detailed information about your configuration and setup!
  (palm.muk.uni-hannover.de/trac/newticket)

Simulation monitoring (II)

RUN_CONTROL file

- In job log, find the temporary directory info

```
*** execution starts in directory
    "/gfs2/tmp/nikfarah/example_cbl.630"
-----------------------------------------------------------------------------
```

- Inspect the RUN_CONTROL file during the run by

```
host:...$ cd <path_to_temp_directory>
host:/path_to_temp_directory$ tail -f RUN_CONTROL
```

dt_run_control

```
RUN  ITER. HH:MM:SS.SS    DT(E)      UMAX      VMAX      WMAX     U*     W*       THETA*
--------------------------------------------------------------------------------------------
  0      0 00:00:00.00  20.0000A  -0.2131D -0.2288D  0.1182  0.000  1.38   0.000E+00
  0      1 00:00:20.00  20.0000D  -0.2125  -0.2273   0.1172  0.009  0.69  -0.132E+02
  0      2 00:00:40.00  20.0000D  -0.2117  -0.2252   0.1162  0.009  0.69  -0.133E+02
  0      3 00:01:00.00  20.0000D  -0.2106  -0.2227   0.1152  0.009  0.69  -0.134E+02
  0      4 00:01:20.00  20.0000D  -0.2093  -0.2197   0.1141  0.009  0.69  -0.134E+02
  0      5 00:01:40.00  20.0000D  -0.2077  -0.2164   0.1129  0.009  0.93  -0.134E+02
  0      6 00:02:00.00  20.0000D  -0.2058  -0.2126   0.1117  0.009  0.93  -0.133E+02
  0      7 00:02:20.00  20.0000D  -0.2037  -0.2086   0.1103  0.009  1.22  -0.131E+02
  0      8 00:02:40.00  20.0000D   0.3348D  0.3126D  0.1636  0.011  1.22  -0.105E+02
  ...
```

- After the run, this file is copied and renamed to e.g. .../JOBS/example_cbl/MONITORING/example_cbl_rc

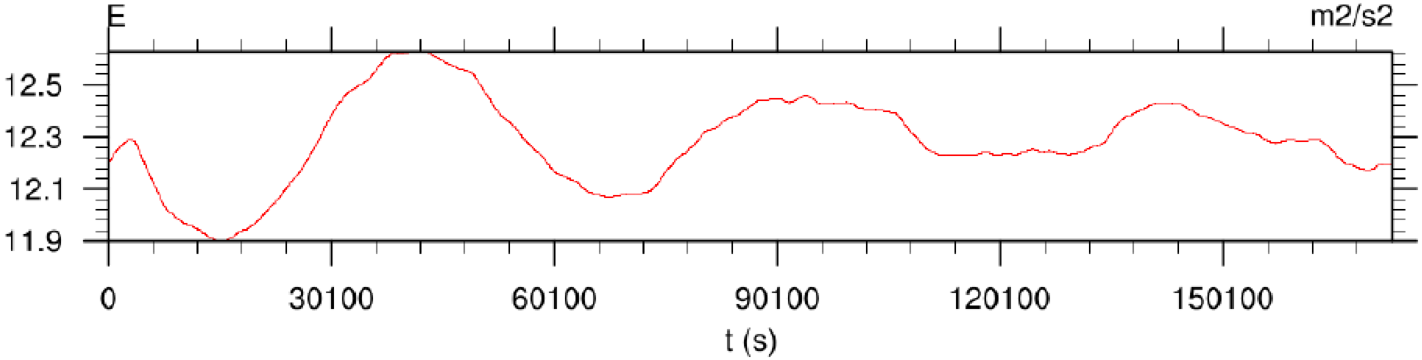
Evaluation of simulation results (I)

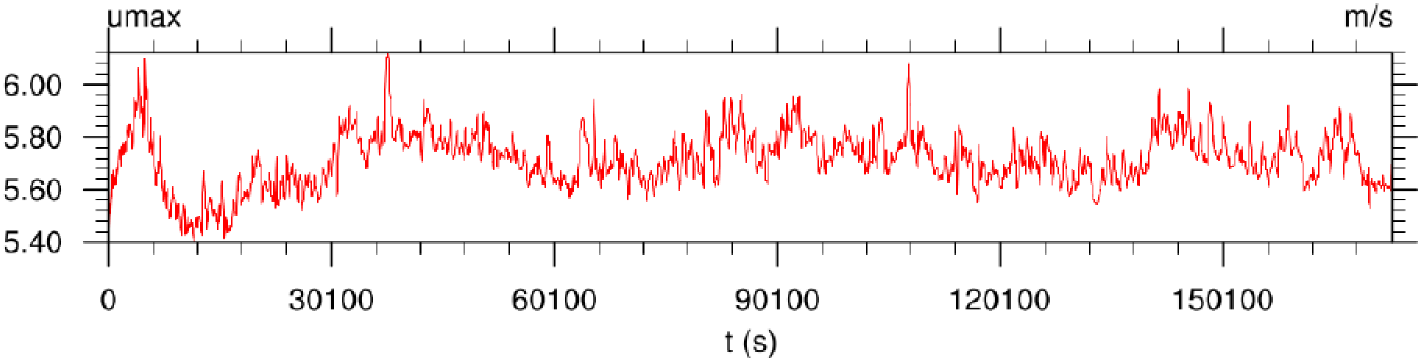
- Some meteorological background required for simulation setup & output of analysis data
- Some general remarks for scenarios you might come along:
  - basically, three different types of simulation:
    - stationary mean conditions
    - quasi-stationary mean conditions
    - instationary mean conditions

Evaluation of simulation results (II-a)

Stationary mean conditions

- Forcing and atmoshperic state remains constant over time
- Mainly the case for pure mechanical forcing (no thermal effects)

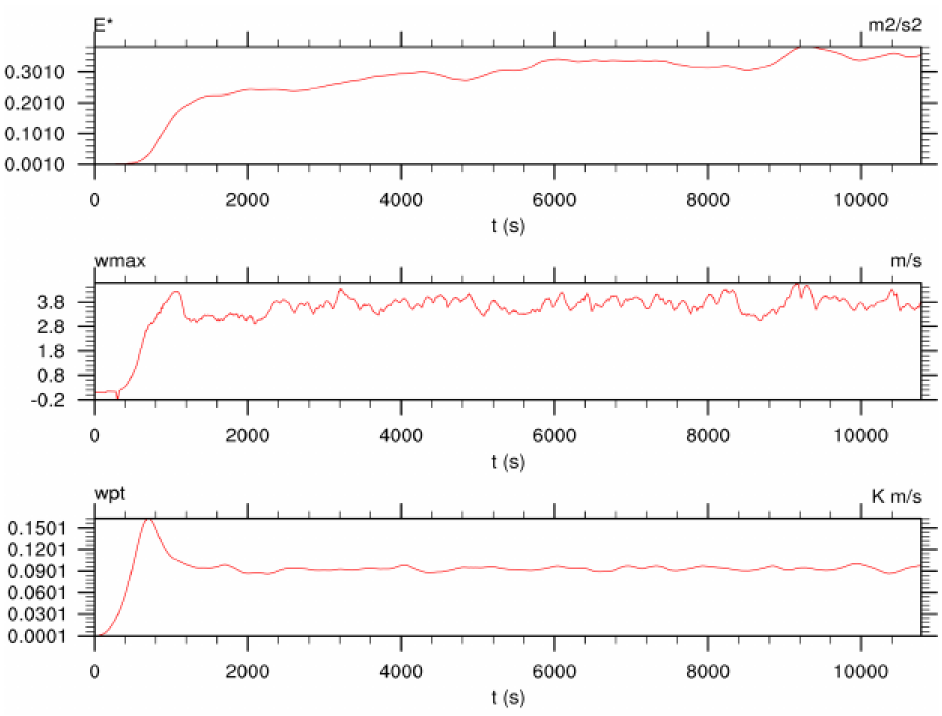


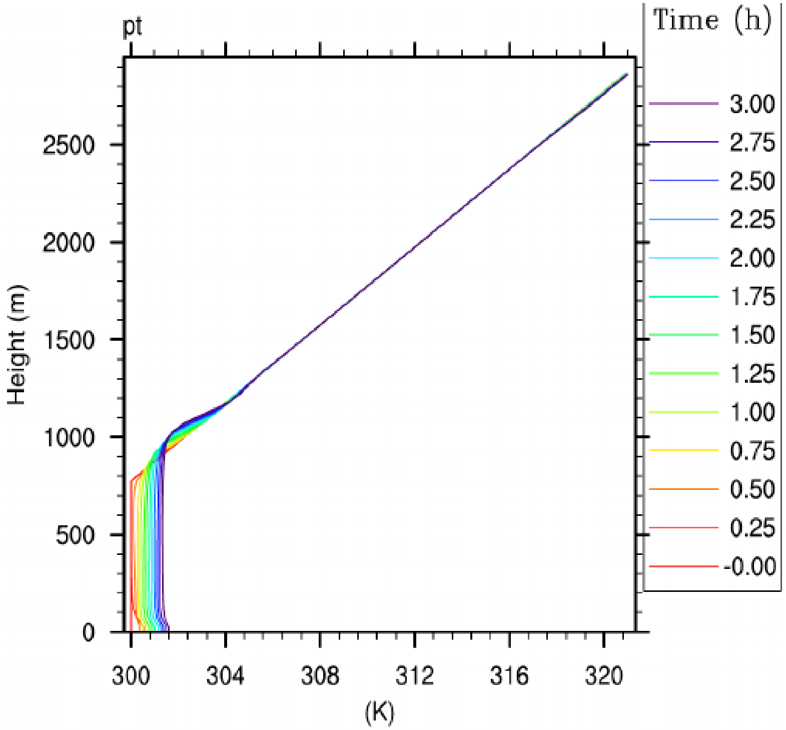


Evaluation of simulation results (II-b)

Quasi-stationary mean conditions

- Forcing constant, but atmospheric state changes over time
- e.g. the case for a time-constantly heated convective boundary layer (constant heat input from the surface, idealistic studies)

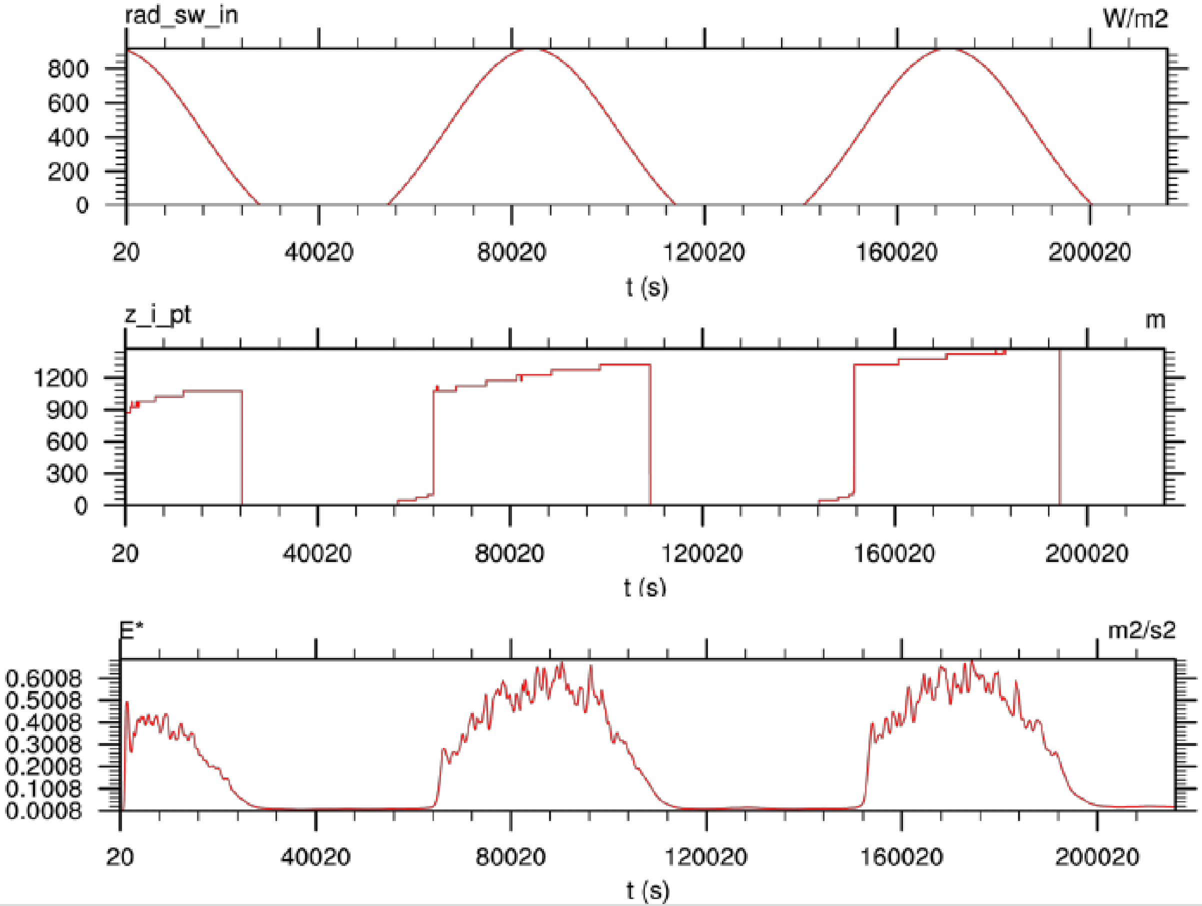




Evaluation of simulation results (II–c)

Instationary mean conditions

- Non-constant forcing, i.e. simulation of daily cycle



Evaluation of simulation results (III)

- For daily cycle simulation normally necessary to run half-a-day in advance as atmospheric spinup
- Concerning turbulence development we don't have to worry much in a city simulation with all the obstacles and different surfaces
  (but it does no harm to set parameter create_disturbances = .T.)
- Concerning averaging intervals, the typically used 30-min average will do
- Parameters for output definition under
  palm.muk.uni-hannover.de/trac/wiki/doc/app/d3par#output

Thank you for your attention