# Offline SURFEX Data Assimilation in the NWP model HARMONIE-AROME

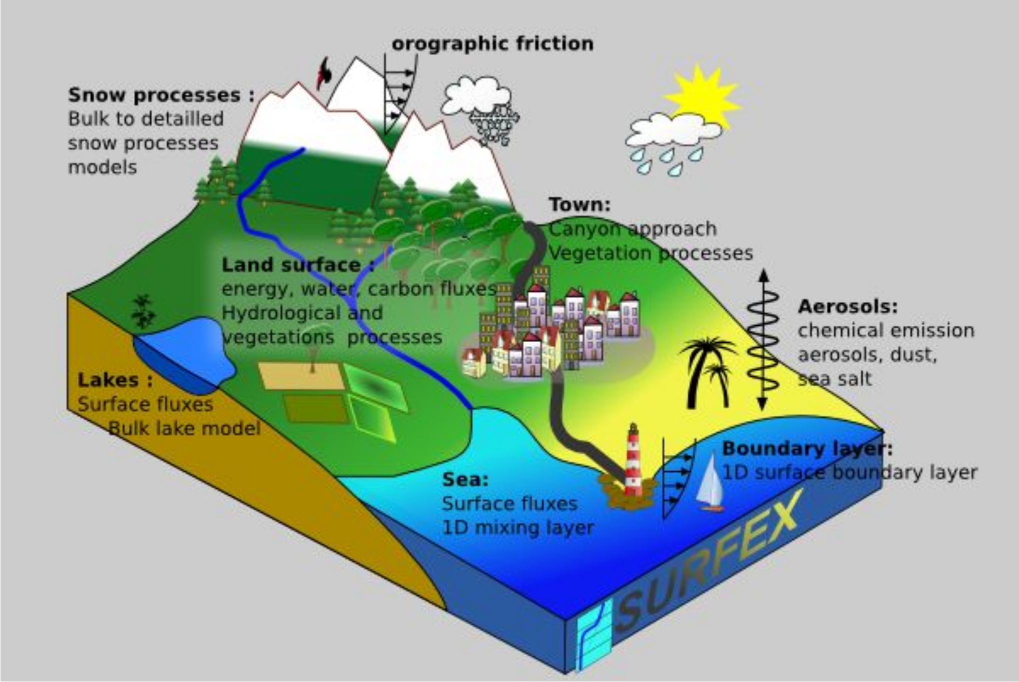
Trygve Aspelien

SURFEX User Workshop
Toulouse, March 18th 2019

# SURFEX properties

- Has common source code entry points which can called from applications
- Several IO methods built in (how to exchange values between the calling application and the SURFEX library)

  CSURF_FILETYPE/HPROGRAM



# NWP cyclomania...

- I will focus on cy43h2 and SURFEX 8.1 which is used in HARMONIE

Tromsø, the “birth place” of SURFEX 8.1 in cy43h2. HIRLAM Surface working week May 2018

# Fileformats (*CSURF_FILETYPE* in SURFEX)

cy40hX:

- "LFI" as intermediate format used in SURFEX applications (*.lfi)
- "FA" as fileformat in AROME applications (*.sfx)

cy43h2:

- "FA" format is only NWP file format and also possible to use in SURFEX applications (*LFAGMAP=.TRUE.*)
- Possible to override *CSURF_FILETYPE* for advanced users, e.g. run with NetCDF file format which is more suitable for offline runs

# (offline) SURFEX concepts

- PGD (PhysioGraphical Data)
  - Entry point: PGD_SURF_ATM
  - ECOCLIMAP, sand, clay, tree height data etc
- PREP (Prepare the initial data for a simulation)
  - Entry point: PREP_SURF_ATM
  - Read an ECMWF grib file and initialize SURFEX. Alt: SURFEX to SURFEX (extern)
- OFFLINE (Run an offline simulation)
  - Entry point: COUPLING_SURF_ATM_n
- SODA (Surfex Offline Data Assimilation)
  - Entry point: ASSIM_SURF_ATM_n
- All applications share the same entry points for:
  - INIT_SURF_ATM_n (application initialization)
  - WRITE_SURF_ATM_n (write restart files)
  - WRITE_DIAG_SURF_ATM_n (write diagnostics)

# SURFEX data assimilation

- Entry point: **ASSIM_SURF_ATM_n**
  - Decision taken Workshop on SURFEX data assimilation in March 2012 (J-F. Mahfouf et.al)
  - ASSIM_SEA_n
    - INPUT/NONE
  - ASSIM_INLAND_WATER_n
    - INPUT/NONE
  - ASSIM_NATURE_n
    - OI/EKF/NONE
    - Snow assimilation
  - ASSIM_TOWN_n
    - ROADT/NONE

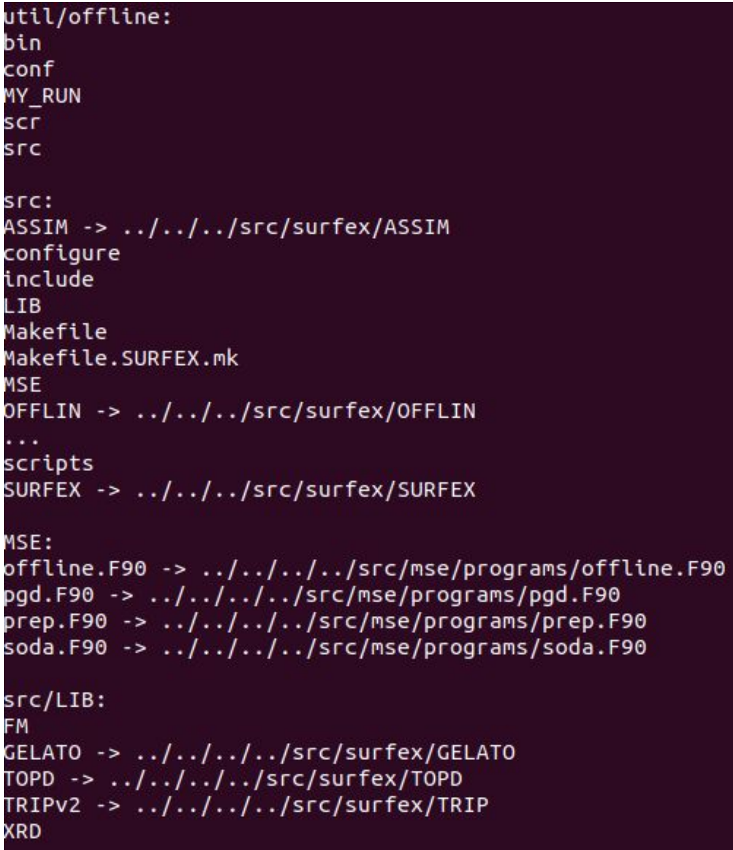
- Entry point: **OI_CONTROL**
  - Used to be an OI_main binary only
  - Only OI which can be used for soil
  - Update some specific surfex variables for all tiles in the same routine

Used by Meteo France and others

Both entry points enable inline/offline approach!
It is just about which “source code tree” which is called and actually used….

# Assimilation developments (SURFEX 8.1 (cy43h2))

- All tiles have SCHEME options (also NONE)
- Assimilation input is generalized having one "source code entry point" independent of the file format (also on multiple processors)
- Gathering and writing of increments are done one the IO process and written to log file
- Setup of offline SURFEX inside the inline code inside util/offline
  - Same code inline/offline
  - enables the parallell possibilities of PGD/PREP/OFFLINE/SODA



```
util/offline:
bin
conf
MY_RUN
scr
src

src:
ASSIM -> ../../../src/surfex/ASSIM
configure
include
LIB
Makefile
Makefile.SURFEX.mk
MSE
OFFLIN -> ../../../src/surfex/OFFLIN
...
scripts
SURFEX -> ../../../src/surfex/SURFEX

MSE:
offline.F90 -> ../../../../src/mse/programs/offline.F90
pgd.F90 -> ../../../../src/mse/programs/pgd.F90
prep.F90 -> ../../../../src/mse/programs/prep.F90
soda.F90 -> ../../../../src/mse/programs/soda.F90

src/LIB:
FM
GELATO -> ../../../../src/surfex/GELATO
TOPD -> ../../../../src/surfex/TOPD
TRIPv2 -> ../../../../src/surfex/TRIP
XRD
```

# Input to SURFEX assimilation (1)

| SEA: | ASCII filename | FA filename |
|---|---|---|
| - *CFILE_FORMAT_SST* | SST_SIC.DAT | SST_SIC |
| Extrapolations: | | |
| - *CFILE_FORMAT_LSM* | LSM.DAT | FG_OI_MAIN |
| OI for nature: | | |
| - *CFILE_FORMAT_FG* | FIRST_GUESS_YYMMDDHH.DAT | FG_OI_MAIN |
| - *CFILE_FORMAT_CLIM* | CLIMATE.DAT | clim_isba |

CANARI

OFFLINE

# Observations (the reason for surface analysis :-))

SURFEX can use 5 observation types:

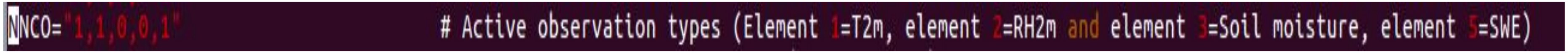
- T2M
- RH2M
- WG1
- WG2
- SWE/SD
  - *LSWE=.TRUE. / .FALSE.*

Proxys for update of soil temperature and/or soil moisture

Used in experiments with EKF and satellite products

Observations are snow depth, the model’s prognostic variable is Snow Water Equivalent. Challenge: Need snow density for conversion

sms/config_exp.h:



```
NNCO="1,1,0,0,1"                    # Active observation types (Element 1=T2m, element 2=RH2m and element 3=Soil moisture, element 5=SWE)
```

# Input to SURFEX assimilation (2): Observations

| Observations: | ASCII filename | FA filename |
|---|---|---|
| - *CFILE_FORMAT_OBS* | *OBSERVATIONS_*YYMMDDHH.DAT | CANARI |
| OFFLINE | | |

How do we create the “observations” for surfex assimilation?

# Methods to create observations (T2m/RH2m/SD)

## CANARI

- ➔ ODB
- ➔ CANARI
  - ◆ Quality control
  - ◆ OI
  - ◆ *SURFEX ENTRY POINT*
    - ● ASSIM_SURF_ATM_n
    - ● OI_control

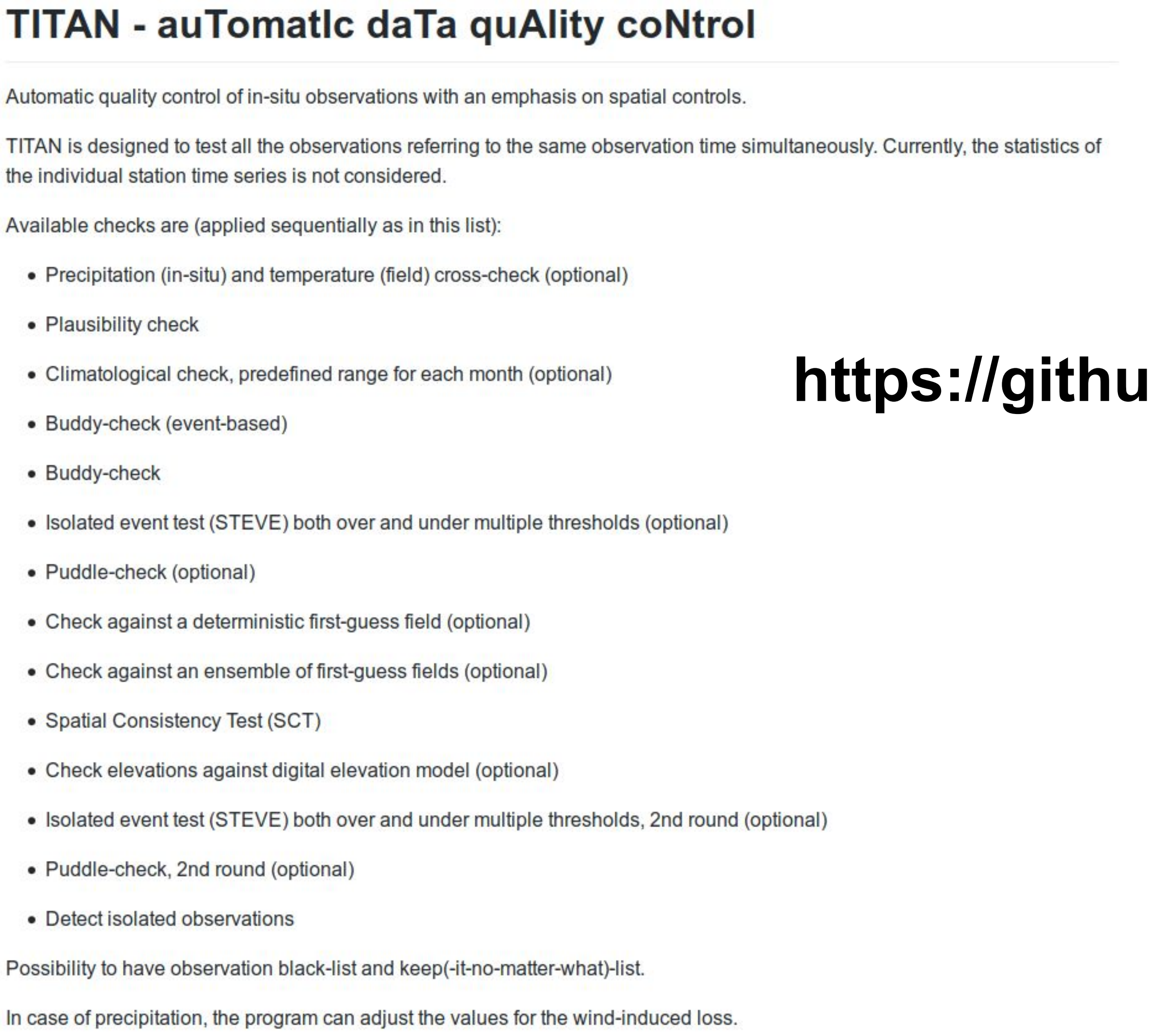
Inline

## gridpp

- ➔ ASCII observations
- ➔ Quality control (TITAN)
- ➔ gridPP
- ➔ Soda
  - ◆ ASSIM_SURF_ATM_n

Offline

Results should be identical assuming same values as arguments and same namelist settings



# TITAN - auTomatIc daTa quAlity coNtrol

Automatic quality control of in-situ observations with an emphasis on spatial controls.

TITAN is designed to test all the observations referring to the same observation time simultaneously. Currently, the statistics of the individual station time series is not considered.

Available checks are (applied sequentially as in this list):

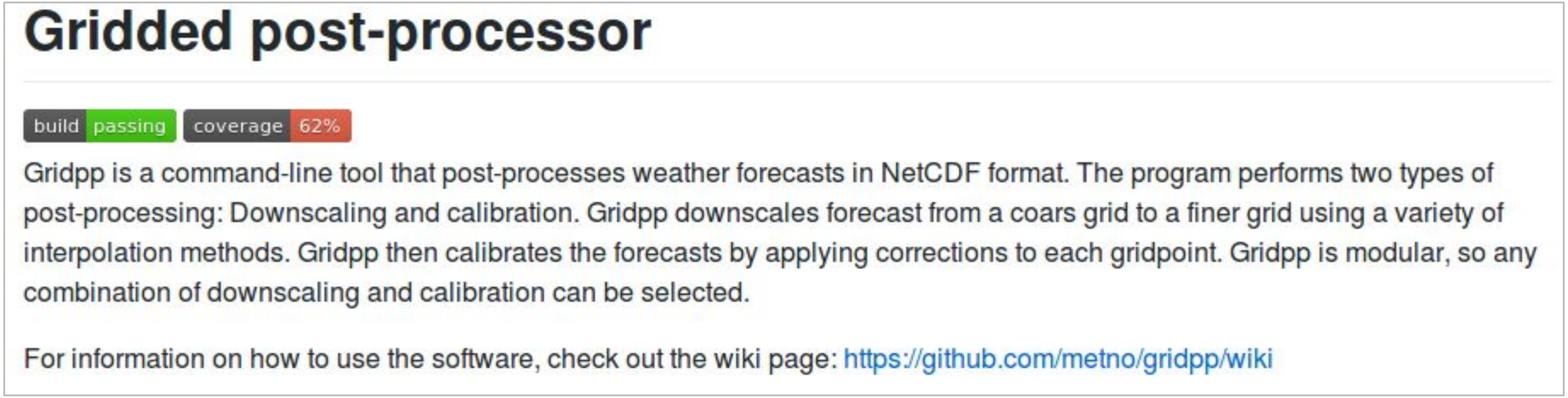
- Precipitation (in-situ) and temperature (field) cross-check (optional)
- Plausibility check
- Climatological check, predefined range for each month (optional)
- Buddy-check (event-based)
- Buddy-check
- Isolated event test (STEVE) both over and under multiple thresholds (optional)
- Puddle-check (optional)
- Check against a deterministic first-guess field (optional)
- Check against an ensemble of first-guess fields (optional)
- Spatial Consistency Test (SCT)
- Check elevations against digital elevation model (optional)
- Isolated event test (STEVE) both over and under multiple thresholds, 2nd round (optional)
- Puddle-check, 2nd round (optional)
- Detect isolated observations

Possibility to have observation black-list and keep(-it-no-matter-what)-list.

In case of precipitation, the program can adjust the values for the wind-induced loss.

**https://github.com/metno/TITAN**

# gridpp



## Gridded post-processor

Gridpp is a command-line tool that post-processes weather forecasts in NetCDF format. The program performs two types of post-processing: Downscaling and calibration. Gridpp downscales forecast from a coars grid to a finer grid using a variety of interpolation methods. Gridpp then calibrates the forecasts by applying corrections to each gridpoint. Gridpp is modular, so any combination of downscaling and calibration can be selected.

For information on how to use the software, check out the wiki page: https://github.com/metno/gridpp/wiki

**Variable name (in file)**: -v variable-name

**OI calibrator**: -c oi

**OI options**: d=X h=Z useEns=0 sigma=S elevGradient=0 minObs=0 landOnly=1 diaFile=name-of-file

**Parameter file (observations)**: -p $param type=netcdf dimName=coefficient varName=coefficients

**Quality control calibrator and options:** -c qc min=0.00001 max=1

# Generation of offline SURFEX forcing

https://github.com/metno/offline-surfex-forcing

- Written in python (2.7)
  - Installed with pip
- Flexible
  - Different input for each forcing variable
  - Convert to forcing property from derived parameters
- Supports grib/netcdf input (ascii).
  - Can be extended to FA with EPyGrAM
- Output in NetCDF or ASCII
- Used in offline runs or control + perturbed runs for EKF

Also has some procedures for reading SURFEX output

# Offline runs with inline source code and scripts

ALT1:

Harmonie start DTG=YYYYMMDDHH PLAYFILE=offline CSURF_FILETYPE=NC ……. other variables….

ALT 2:

Emulate the needed variables for the Harmonie scripts system (util/offline/scr/main.sh)

Time stepping loop calling main.sh (util/offline/scr/offline_soda_loop.sh)

# Projections

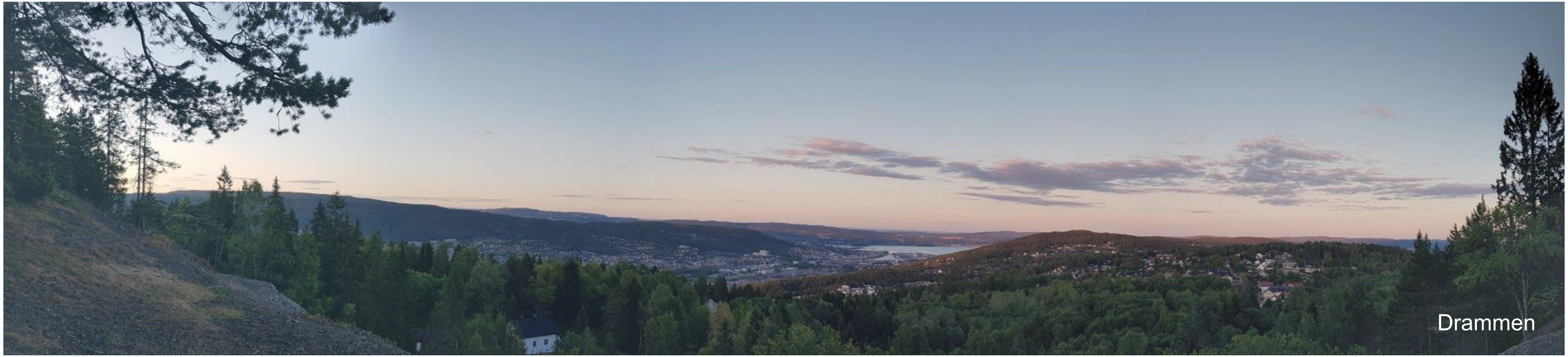
- Today projections can easily be handled with external software like the proj4 library
- The standard projection UTM is not supported in SURFEX. Used by e.g. the hydrological agency in Norway.
- The SURFEX source code is written in Fortran. Most of the projection routines are in-house made routines. The basic usage is transformation to/from grid coordinates to/from geographical coordinates.
- It exists Fortran software which has support for e.g. proj4. An example is: http://fortrangis.sourceforge.net/

QUESTIONS:

- Anyone having similar plans/ideas?
- Any objections/alternatives to using an external software like FortranGIS?

# Technical sidetracks...

- LNOWRITE
  - modd_write_surf_atm.F90 -> dimension 50000
  - test_record_len (counter increased)
    - Dimension not tested -> memory corruption
    - Counter increased on each processor and nproma block
- MPI_Communicator
  - Testing sub-communicators for each tile (Yurii Batrak)
- MPI implementation (-DSFX_MPI) breaks AROME/CANARI MPI
  - Works now several places because we are lucky and ranks are not set when using HPROGRAM=AROME/BUFFER
  - Assimilation routines has a work-around (LPIO set to true if you are on the IO processor in SURFEX/CANARI)
  - SURFEX should be possible to treat as an external library when calling it from AROME.

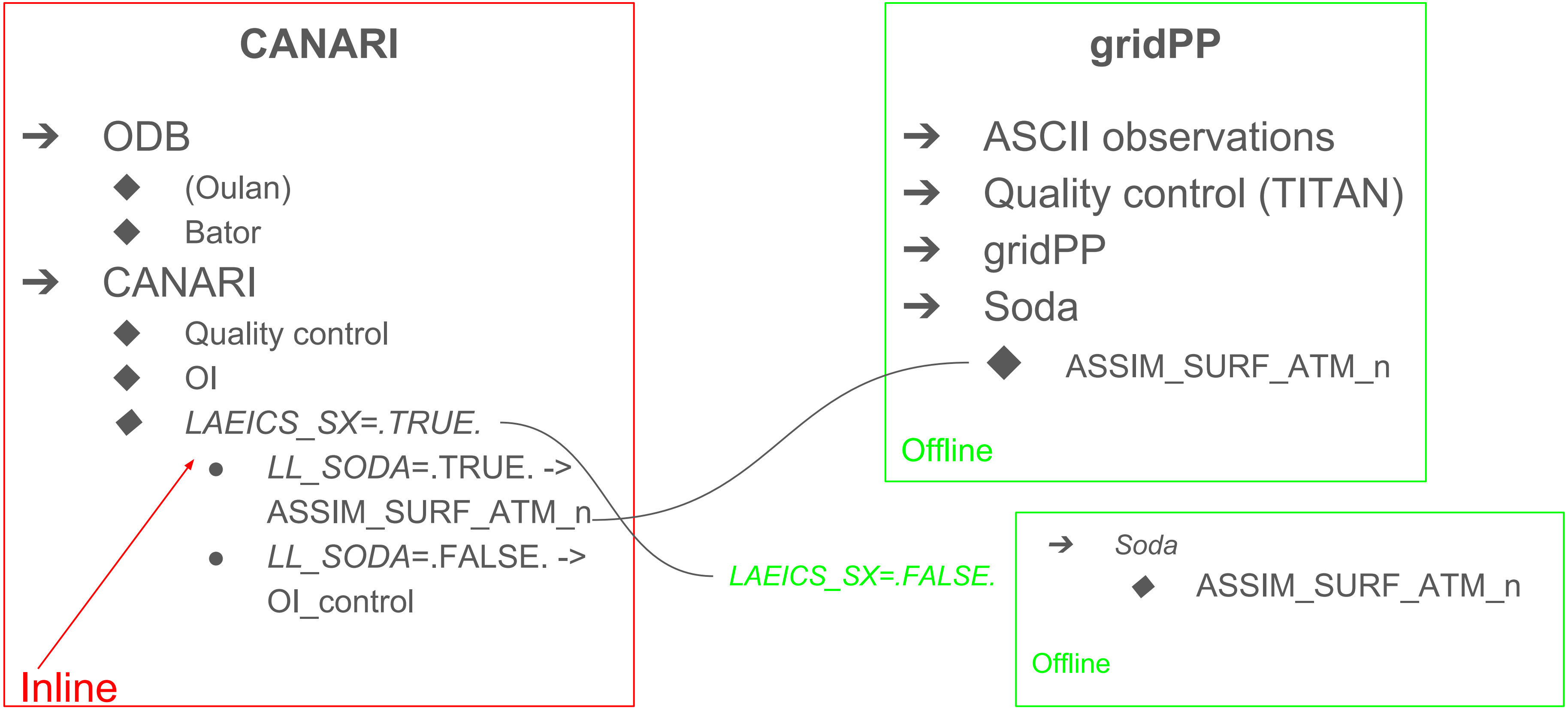


# Merci beaucoup!

# Bonus material

Konnerudkollen

# Methods to create observations (T2m/RH2m/SD)



**CANARI**

- ➔ ODB
  - ◆ (Oulan)
  - ◆ Bator
- ➔ CANARI
  - ◆ Quality control
  - ◆ OI
  - ◆ *LAEICS_SX=.TRUE.*
    - *LL_SODA*=.TRUE. -> ASSIM_SURF_ATM_n
    - *LL_SODA*=.FALSE. -> OI_control

Inline

**gridPP**

- ➔ ASCII observations
- ➔ Quality control (TITAN)
- ➔ gridPP
- ➔ Soda
  - ◆ ASSIM_SURF_ATM_n

Offline

*LAEICS_SX=.FALSE.*

- ➔ *Soda*
  - ◆ ASSIM_SURF_ATM_n

Offline

# Some CANARI namelist settings in &NACTEX

http://www.umr-cnrm.fr/gmapdoc/IMG/ps/canari_doc_cy25t1.ps

nam/harmonie_namelists.pm:

- T2m OI analysis: LAET2M=.TRUE./.FALSE.
- RH2m OI analysis: LAEH2M=.TRUE./.FALSE.
- Snow Water Equivalent (SWE) OI analysis: LAESNM=.TRUE./.FALSE.
  - The first 2 OI methods use different correlation functions than for snow
  - Snow has also a vertical correlation function. For temperature and humidity a vertical correlation function will be used if the LMESCAN setting is activated together with the wanted length scales (Not covered here).
- Do SST analysis: LAESST =.FALSE.
  - We never do this
- Use ECMWF SST field read from SST_SIC: LECSST = .TRUE./.FALSE.
  - We first interpolate SST from the ECMWF boundary file and extrapolate values into fjords

# CANARI length scales in meters (&NAM_CANAPE)

RH2m: REF_A_H2 = 85000.

T2m: REF_A_T2 = 80000.

SWE: REF_A_SN = 30000.

These length scales should be depending on the observation properties, but in practice it also reflects the observation density.

Remark: CANARI need some surface variables which do not exist when running with SURFEX.
They are copied from the climate files when running the task Addsurf

# Tile by tile….

Øvre Leitjern

# ASSIM_SEA_n

- *SST*
  - *CASSIM_SEA*=NONE
    - No update
  - *CASSIM_SEA*=INPUT
    - *LAESST*
      - Read SST from PTS_IN
    - .NOT. *LAESST*
      - Read SST from PSST_IN (in our case ECMWF SST)
- SIC
  - Always done. Only SICE which has it implemented
  - Updates the SIC because SST has changed and this determines where we have sea ice

*LEXTRAP_SEA* should probably always be used with method INPUT and LAESST=.TRUE. if LSM in inconsistent with SURFEX. For us it is default FALSE.

# ASSIM_INLAND_WATER_n

- *CASSIM_WATER*=NONE
  - No update
- *CASSIM_WATER*=INPUT
  - LWATERTG2=.TRUE.
    - All water points set to the undefined
    - All water points also having soil fraction get the value from TG2 (root zone temperature)
  - LWATERTG2=.FALSE.
    - TS_WATER set to PTS_IN

By design *LEXTRAP_WATER* must be used if LWATERTG2=.TRUE.

Should probably always be used with method INPUT if LSM in inconsistent with SURFEX

NB! No assimilation for FLAKE (yet?)

# ASSIM_NATURE_n

Konnerudkollen

# SNOW (ASSIM_ISBA_n)

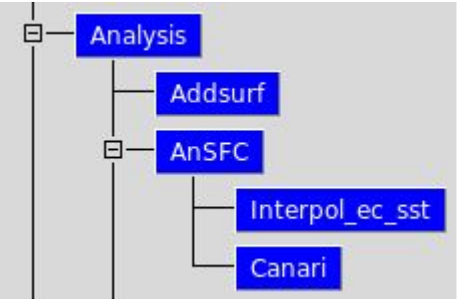
At the moment only implemented for D95. It will be adapted for ISBA-ES.

- LAESNM=.TRUE.
  - ASSIM_ISBA_UPDATE_SNOW
    - Update prognostic SWE based on input SWE/SD
    - Input is assumed to be a grid average so in a multi-patch framework the input SWE is assumed to have the same patch distribution as the first guess

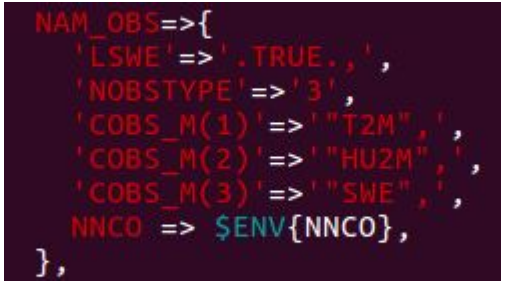
# CASSIM_ISBA=OI

- OI coefficients: fort.61
- OI_CACSTS
  - Adapted from the models without SURFEX
  - Empirical relation between increments in screen level humidity and temperature to ISBA soil temperature, humidity and ice.
  - Temperature is always updated
  - Moisture/ice updates are turned off if the weather conditions are assumed to have a weak connection from screen level e.g.
    - wind
    - precipitation
    - frozen soil
- Update of:
  - WG1
  - WG2
  - TG1
  - TG2
  - WGI2

GUI example

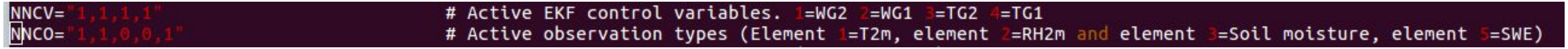


nam/surfex_namelists.pm



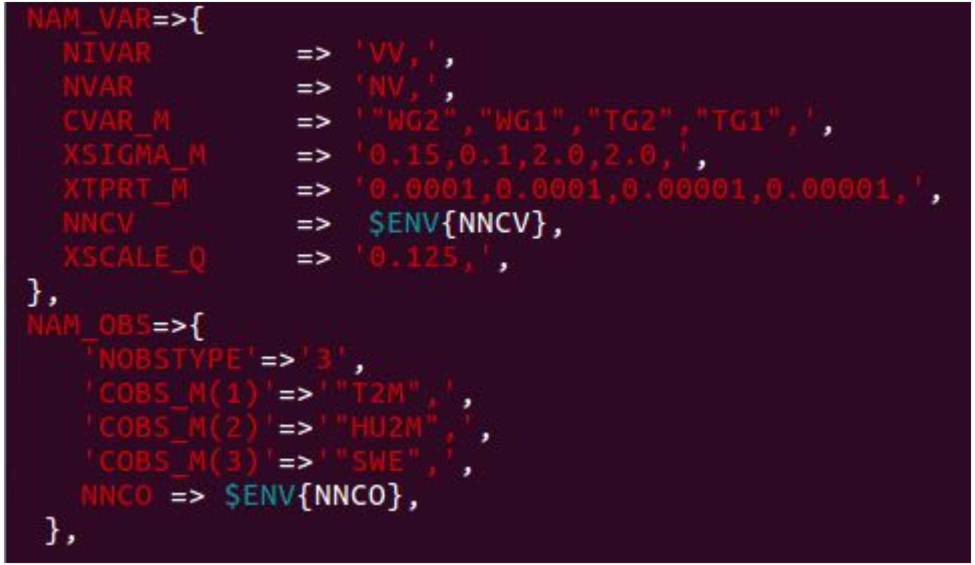
```
NAM_OBS=>{
  'LSWE'=>'.TRUE.,',
  'NOBSTYPE'=>'3',
  'COBS_M(1)'=>'"T2M",',
  'COBS_M(2)'=>'"HU2M",',
  'COBS_M(3)'=>'"SWE",',
  NNCO => $ENV{NNCO},
},
```

# CASSIM_ISBA=EKF  (experimental)

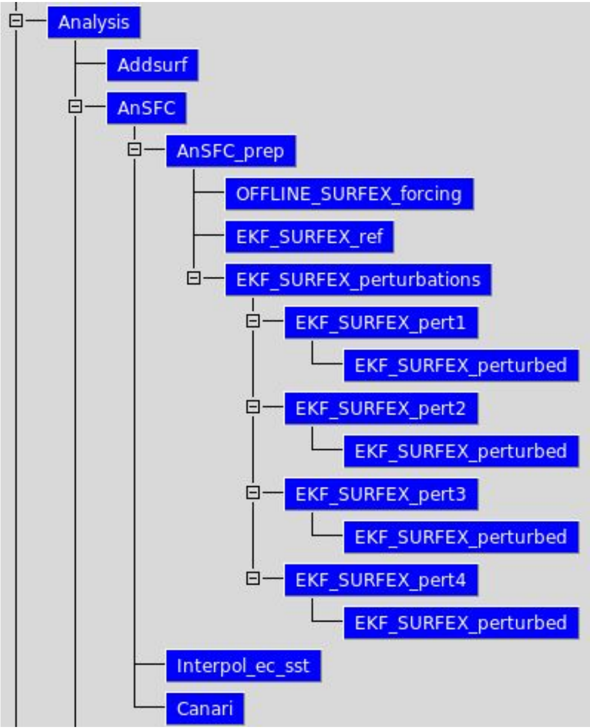


```
NNCV="1,1,1,1"                    # Active EKF control variables. 1=WG2 2=WG1 3=TG2 4=TG1
NNCO="1,1,0,0,1"                  # Active observation types (Element 1=T2m, element 2=RH2m and element 3=Soil moisture, element 5=SWE)
```

- Need forcing
- One control and 1-4 perturbed runs



```
NAM_VAR=>{
  NIVAR          =>  'VV,',
  NVAR           =>  'NV,',
  CVAR_M         =>  '"WG2","WG1","TG2","TG1",',
  XSIGMA_M       =>  '0.15,0.1,2.0,2.0,',
  XTPRT_M        =>  '0.0001,0.0001,0.00001,0.00001,',
  NNCV           =>   $ENV{NNCV},
  XSCALE_Q       =>  '0.125,',
},
NAM_OBS=>{
   'NOBSTYPE'=>'3',
   'COBS_M(1)'=>'"T2M",',
   'COBS_M(2)'=>'"HU2M",',
   'COBS_M(3)'=>'"SWE",',
  NNCO => $ENV{NNCO},
 },
```



- Analysis
  - Addsurf
  - AnSFC
    - AnSFC_prep
      - OFFLINE_SURFEX_forcing
      - EKF_SURFEX_ref
      - EKF_SURFEX_perturbations
        - EKF_SURFEX_pert1
          - EKF_SURFEX_perturbed
        - EKF_SURFEX_pert2
          - EKF_SURFEX_perturbed
        - EKF_SURFEX_pert3
          - EKF_SURFEX_perturbed
        - EKF_SURFEX_pert4
          - EKF_SURFEX_perturbed
    - Interpol_ec_sst
    - Canari

GUI example

# Extrapolation

- *LEXTRAP_NATURE=.TRUE./.FALSE.*
  - Both snow (LAESNM=.TRUE.) and soil temperature/moisture can be extrapolated based on the Land Sea Mask (LSM)
  - Snow also take into account the snow fraction

-> Was used when we had a poor initialization.
-> Is yet again a product of inconsistencies between SURFEX and CANARI LSM

# ASSIM_TEB_n  (TEB is the only town model)

- CASSIM_TEB=NONE
  - No update
- CASSIM_TEB=ROADT
  - Update road layer 3 temperature based on T2m increment
  - Increment = T2m_increment/2*π
  - Legacy option. But has always been there….

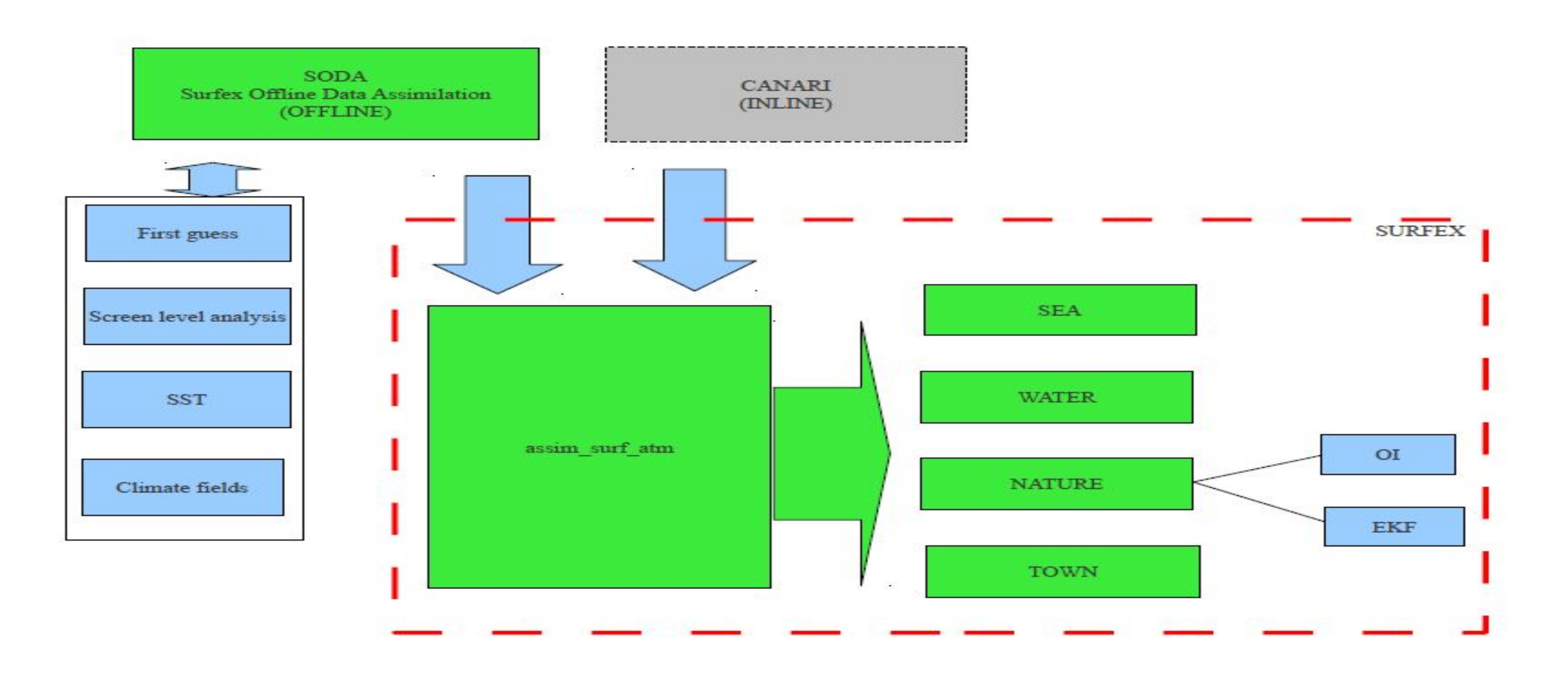
No more tiles....

# You are using Soda...

- If you are using Soda you are using the offline binary SODA, which always use the general assimilation entry point ASSIM_SURF_ATM_n
  - Remark: It exists a seldom used binary OI_main using the entry point OI_CONTROL
- From CANARI you can call both the general assimilation entry point ASSIM_SURF_ATM_n and the entry point OI_CONTROL
  - This is not using Soda but the namelist switch which is called LL_SODA will call the same entry point as the offline binary SODA

Crystal clear?

# One flowchart to explain it all...


SODA
Surfex Offline Data Assimilation
(OFFLINE)
CANARI
(INLINE)
First guess
Screen level analysis
SST
Climate fields
SURFEX
assim_surf_atm
SEA
WATER
NATURE
TOWN
OI
EKF