Surface data assimilation

How we can update the SURFEX initial state based on observations and use it for NWP?

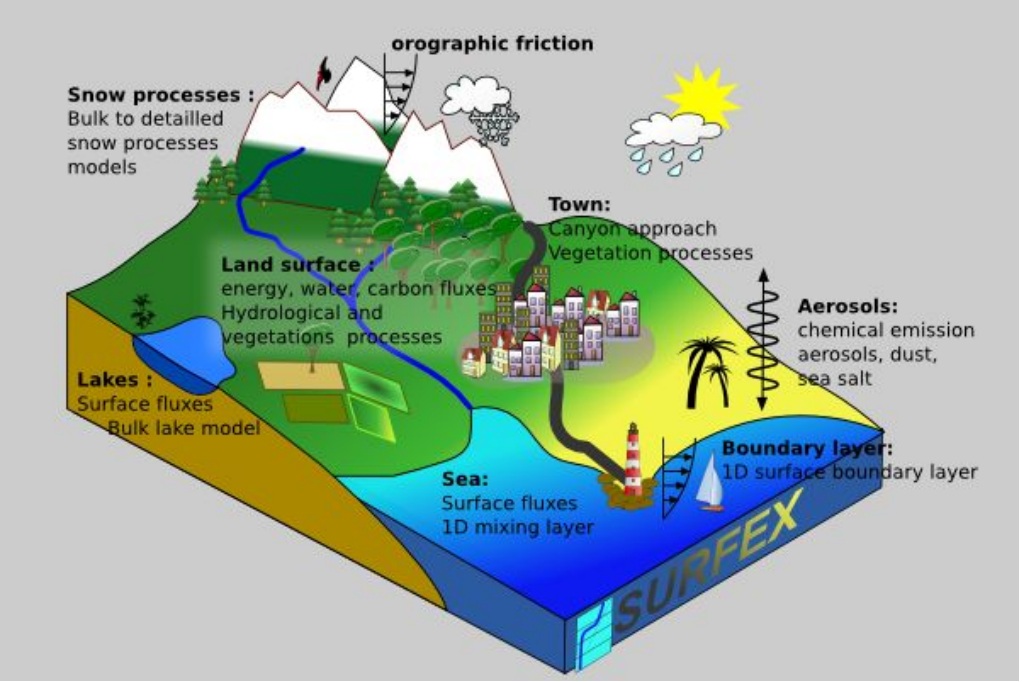
Trygve Aspelien

ALADIN/HIRLAM common data assimilation training week
Budapest, 10-15 February 2019

What is SURFEX?

- External surface model

  - 4 different tiles (SEA, (INLAND) WATER, NATURE, TOWN)
  - lot of options
  - 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



Input & output in NWP context

IN: SURFEX first guess from NWP run:

ICMSHHARM+0003.sfx

THIS PRESENTATION

OUT: SURFEX analysis

ICMSHANAL+0000.sfx

How do I turn this thing on?

sms/config_exp.h

```
ANASURF=CANARI_OI_MAIN                  # Surface analysis (CANARI|CANARI_OI_MAIN|OI|CANARI_EKF_SURFEX|EKF|fgcopy|none)
                                        # CANARI            : Old style CANARI
                                        # CANARI_OI_MAIN    : CANARI + SURFEX OI
                                        # CANARI_EKF_SURFEX : CANARI + SURFEX EKF ( experimental )
                                        # OI                : TITAN + gridPP + SODA
                                        # EKF               : TITAN + gridPP + SODA
                                        # fgcopy            : Copy initial from previous cycle
                                        # none              : No surface assimilation, cold start each cycle
```

scr/include.ass

```
#-----------------------------------------------------
# LIST OBSERVATIONS TO USE (0: NO; 1: YES)
# ODB base list updated accordingly
#-----------------------------------------------------
export SYNOP_OBS=1              # All SYNOP including SHIP
#-----------------------------------------------------
#  TITAN observation settings
#-----------------------------------------------------
export SYNOP_OBS_T2M=1              # SYNOP 2m air temperature from bufr files
export SYNOP_OBS_RH2M=1             # SYNOP 2m relative humidity from bufr files
export SYNOP_OBS_SD=1               # SYNOP snow depth from bufr files
export NETATMO_OBS_T2M=0            # 2m air temperature observations from Netatmo stations
export NETATMO_OBS_RH2M=0           # 2m relative humidity observations from Netatmo stations
```

Fine tuning: nam/surfex_namelists.pm

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

Cyclomania...

- I will focus on cy43h2 and SURFEX 8.1 which is used in HARMONIE
  - This is what I know
  - Hopefully the future…..

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

(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
- OFFLINE (Run an offline simulation)
  - Entry point: COUPLING_SURF_ATM_n
- SODA (Surfex Offline Data Assimilation)
  - Entry point: ASSIM_SURF_ATM_n

Outside the scope of this presentation

Where is this also used?

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

- 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

Input to SURFEX assimilation (1)

	ASCII filename	FA filename
SEA:		
- *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)
```

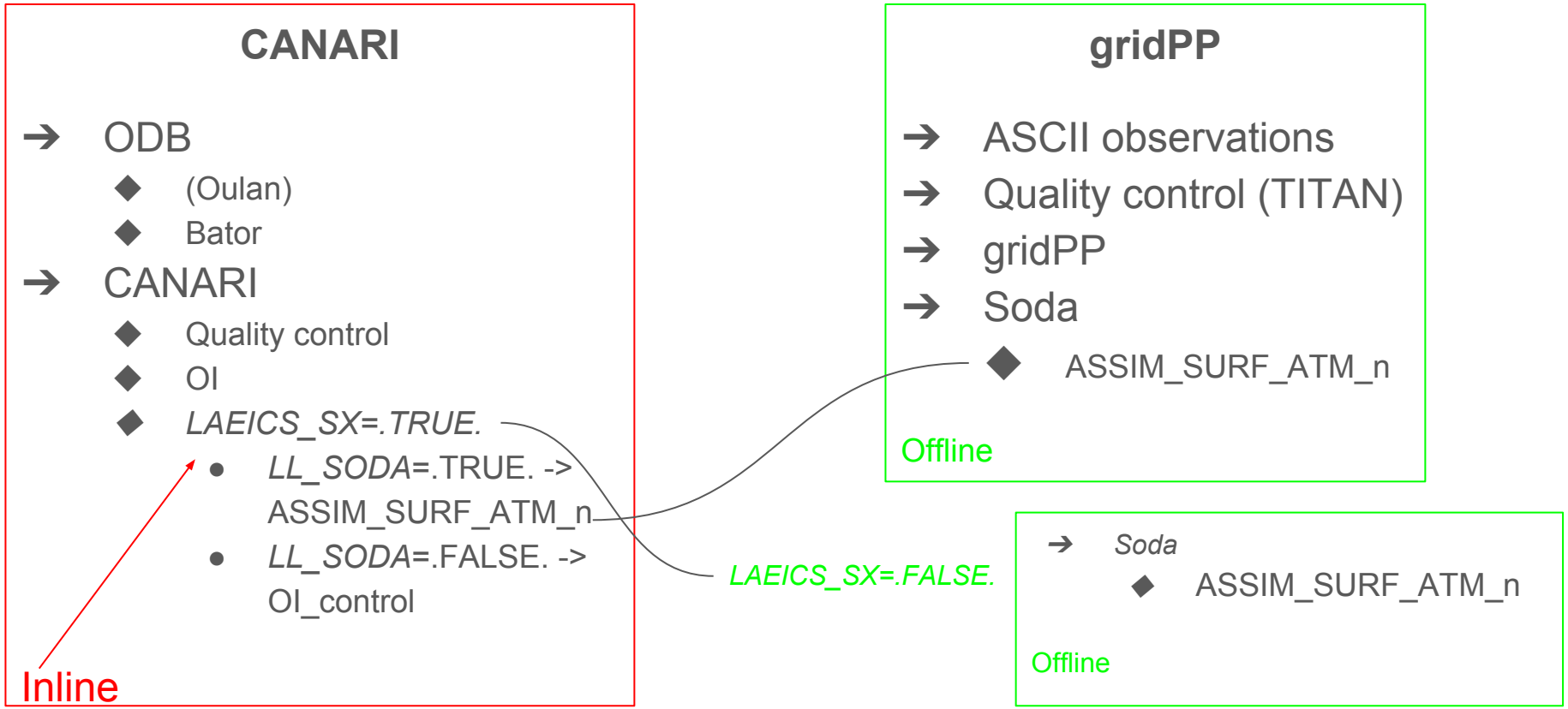
Input to SURFEX assimilation (2): Observations

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)



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

ALT: gridPP

Gridded post-processor

build passing coverage 62%

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


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)

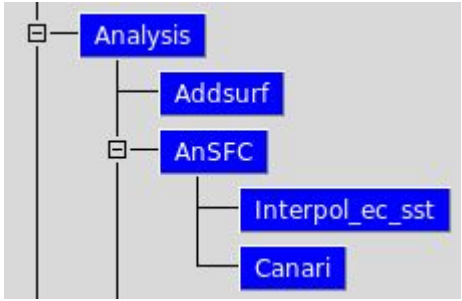
At the moment only implemented for D95. It will be adapted for ISBA-ES.

- 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

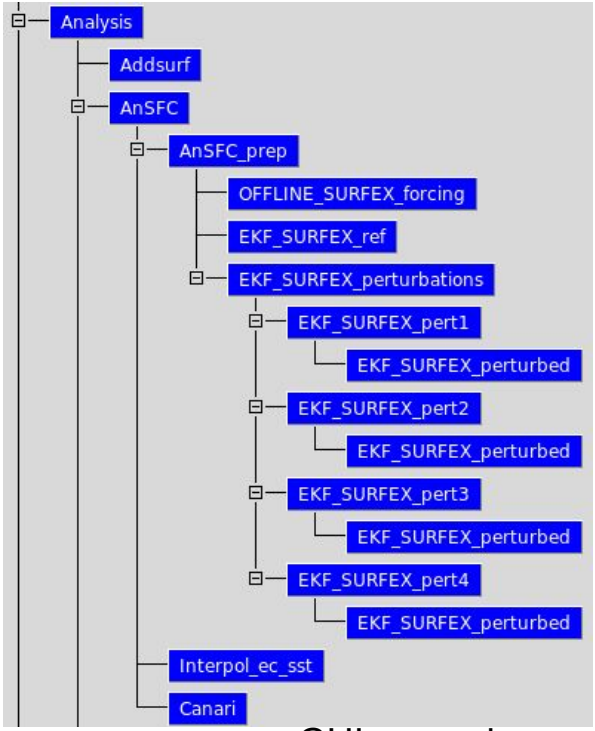
```
NAM_OBS=>{
  'LSWE'=>'.TRUE.,',
  'NOBSTYPE'=>'3',
  'COBS_M(1)'=>'"T2M",',
  'COBS_M(2)'=>'"HU2M",',
  'COBS_M(3)'=>'"SWE",',
  NNCO => $ENV{NNCO},
},
```

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},
 },
```



GUI example

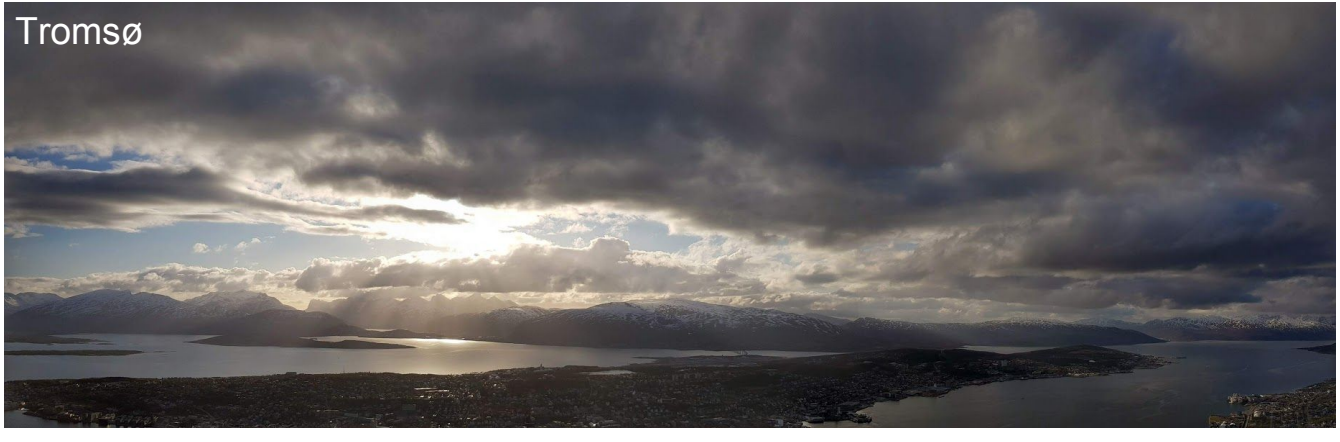
Extrapolation

- *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….



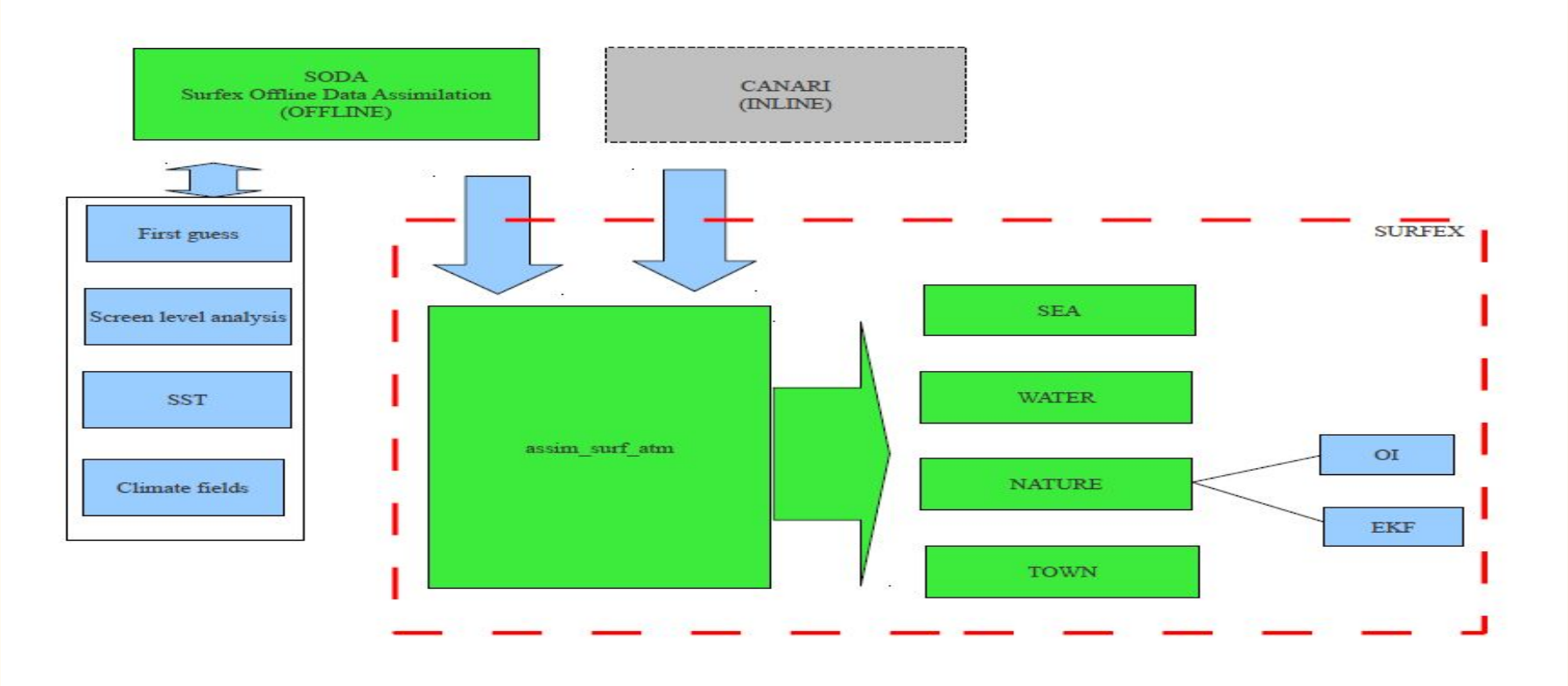
No more tiles....

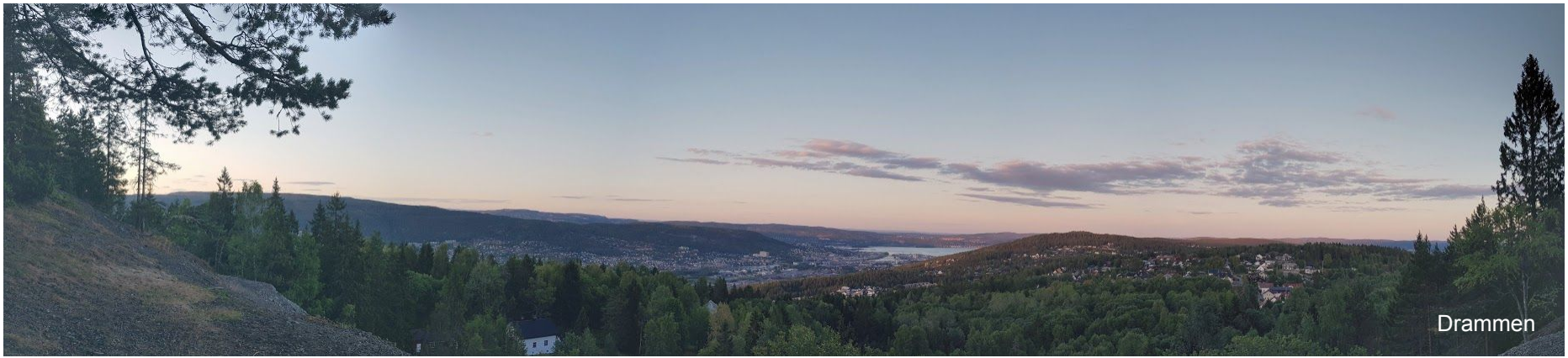
You are using Soda...

- 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...





Köszönöm