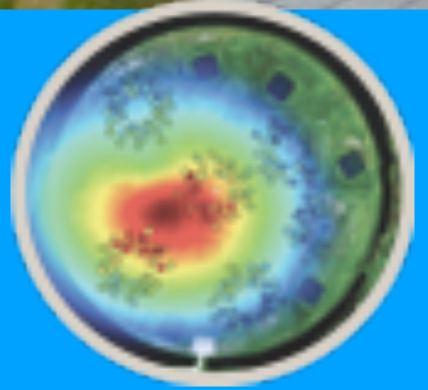
LoSiTo

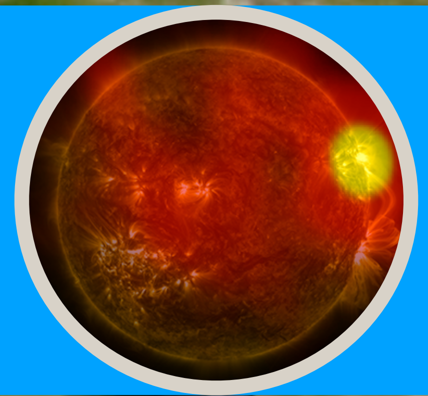
The LOFAR Simulation Tool

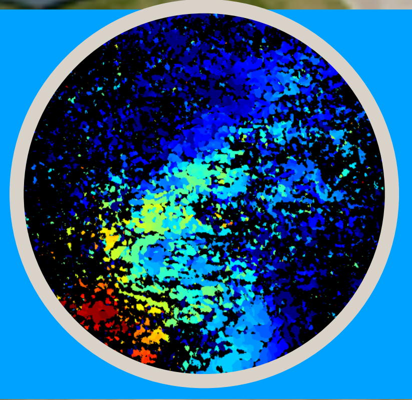
Henrik Edler

29/03/2021



6th LOFAR Data School





LoSiTo - the LOFAR Simulation Tool

Software to simulate LOFAR observations, including realistic systematic effects and noise.

- Developed together with David Rafferty and Francesco de Gasperin
- Documentation & code: github.com/darafferty/losito
- Written in Python -> easy to contribute and adapt
- Makes use of LOFAR software (DP3, LSMTool, RMExtract, LoSoTo)

Motivation

Why do we need a simulation code?

- Low frequency calibration is **highly complex**
  - Only in simulations, we have a ground-truth to evaluate our calibration solutions
  - Test and cross-check data-reduction procedures
- Development of calibration strategies for future development versions of LOFAR (e.g. **LOFAR 2.0**)

Illustrations for some LOFAR calibration pipelines (van Weeren 2016, de Gasperin 2019, de Gasperin 2020)

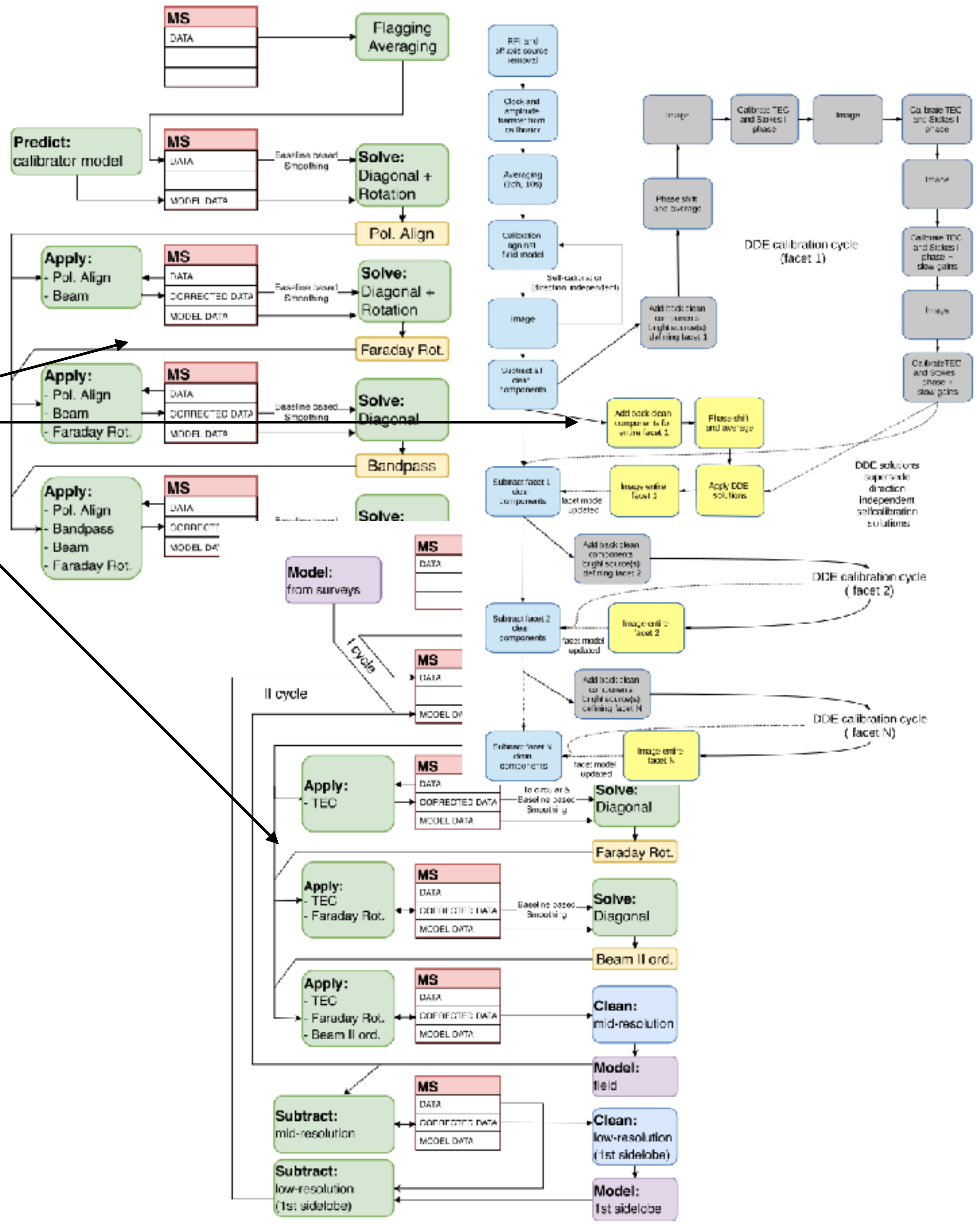


Image credits: A. Corstanje, F. Sweijen, C. Van Eck, P. Zucca

Scheme of LoSiTo

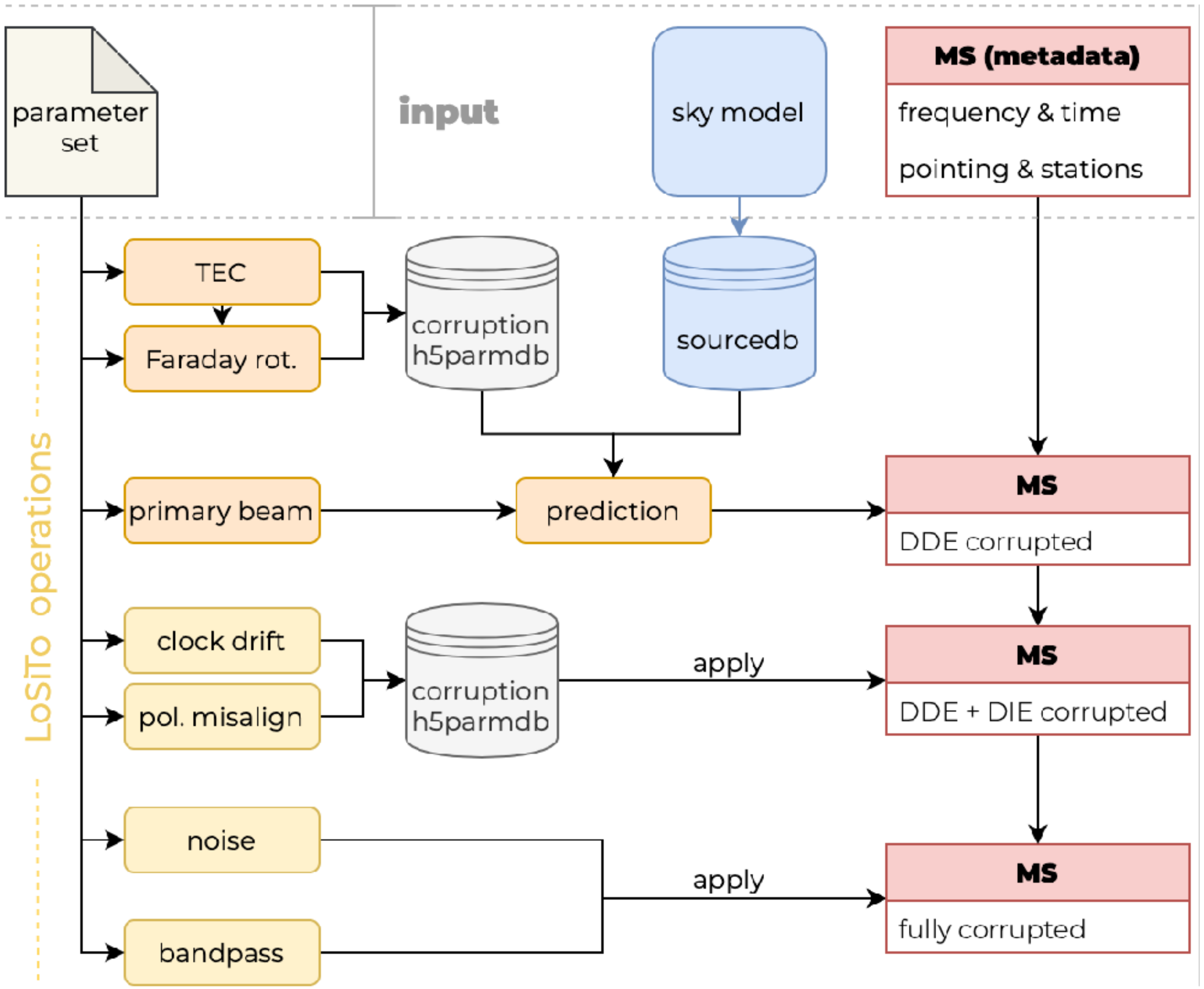
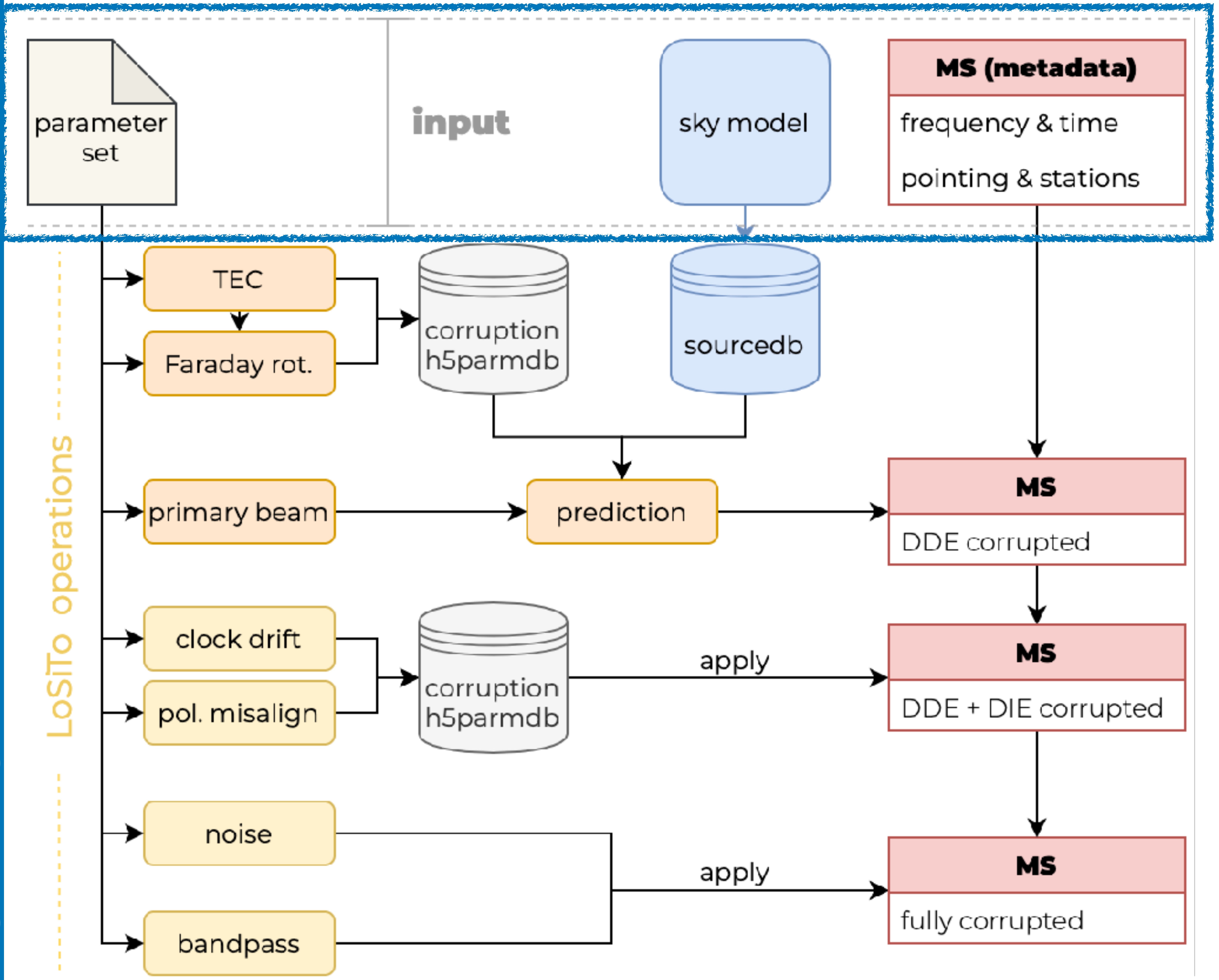


Image credits: A. Corstanje, F. Sweijen, C. Van Eck, P. Zucca

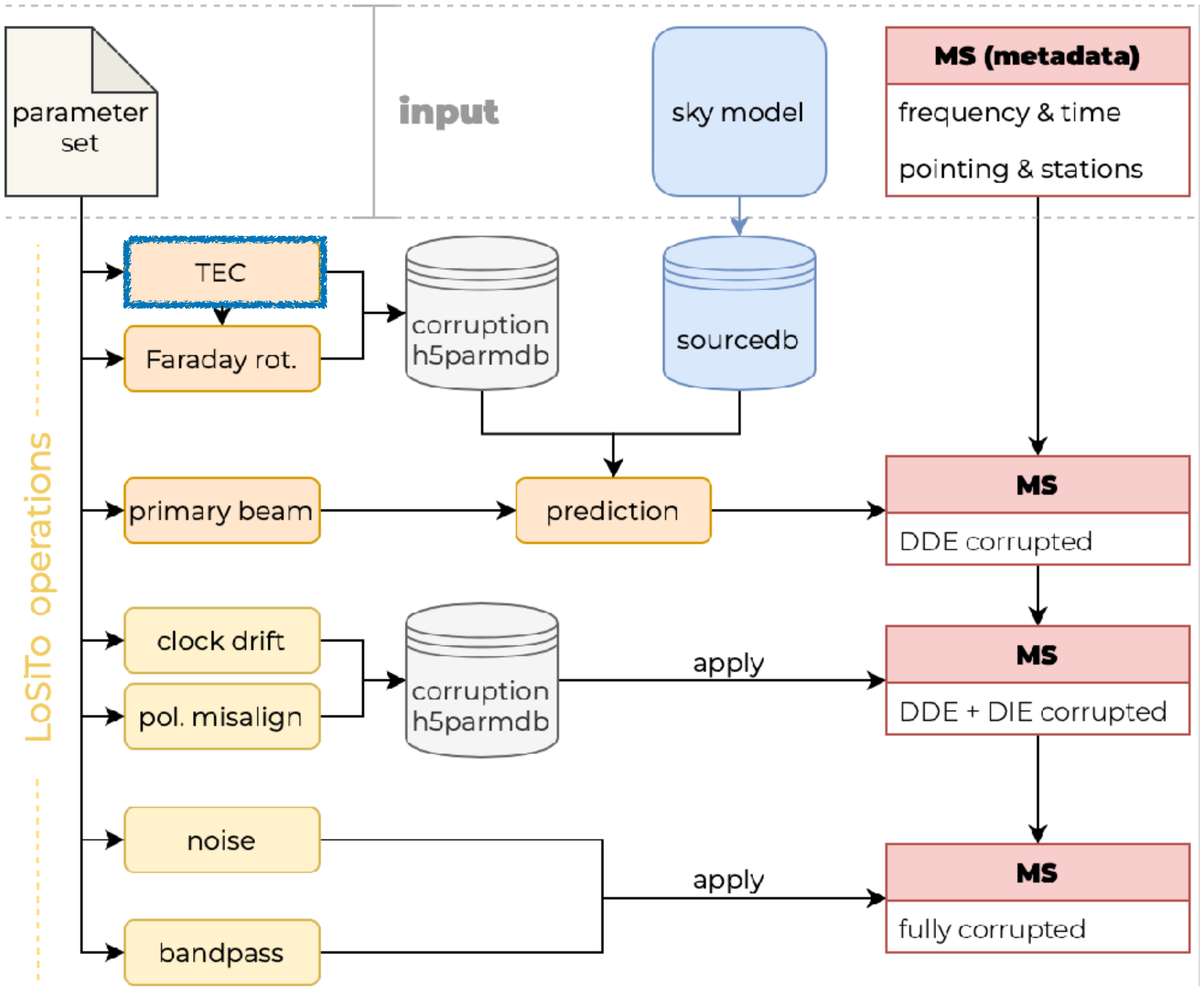
Scheme of LoSiTo



6th LOFAR Data School

Image credits: A. Corstanje, F. Sweijen, C. Van Eck, P. Zucca

Scheme of LoSiTo



- Ionospheric corruptions: Need model of total electron content (TEC)

$$\phi_{\text{disp. delay}} = -\,84.48 \left[\frac{\mathrm{d}TEC}{1\,\mathrm{TECU}}\right]\left[\frac{100\,\mathrm{MHz}}{\nu}\right]\,\mathrm{rad}$$

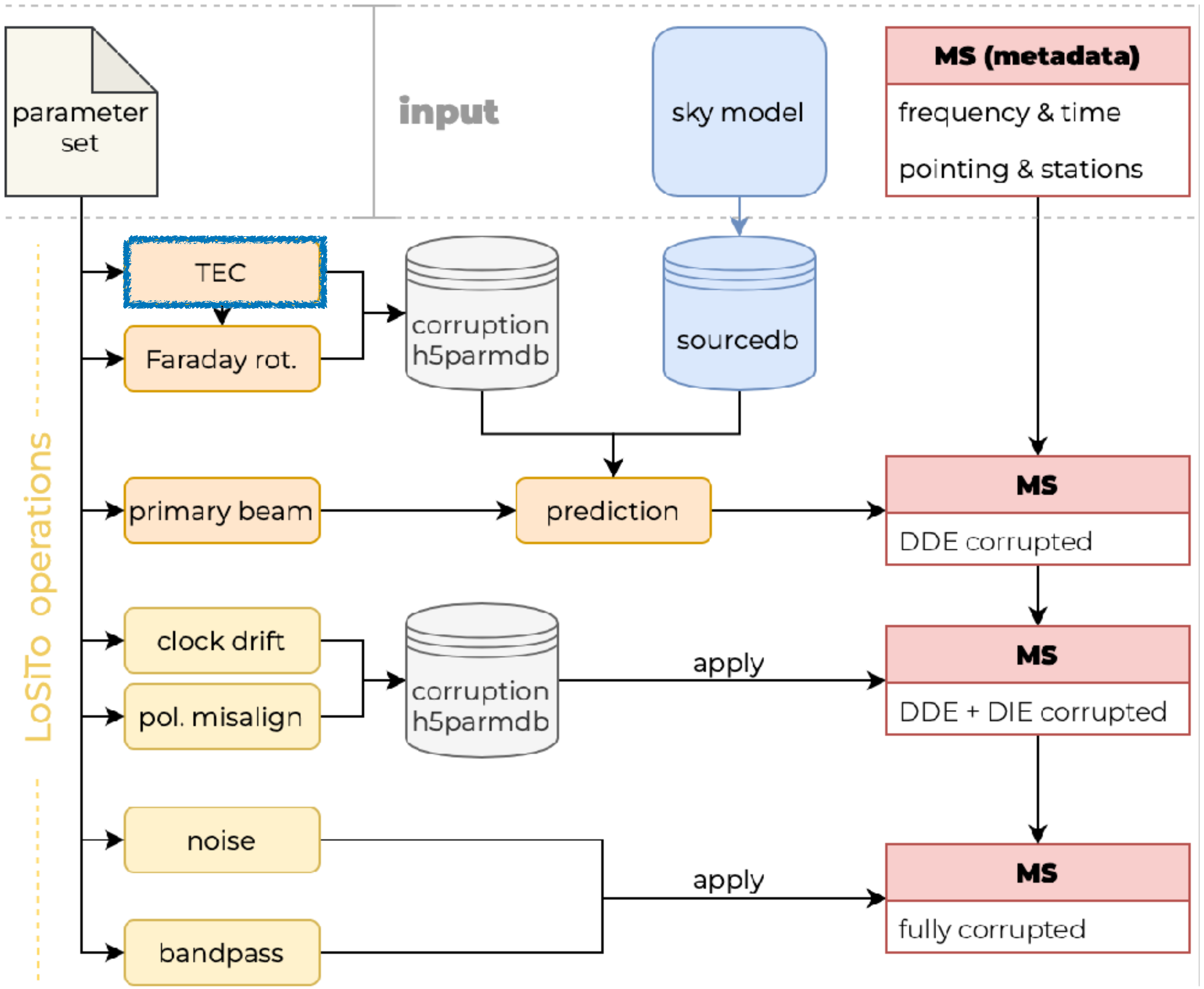
- Thin layer approximation

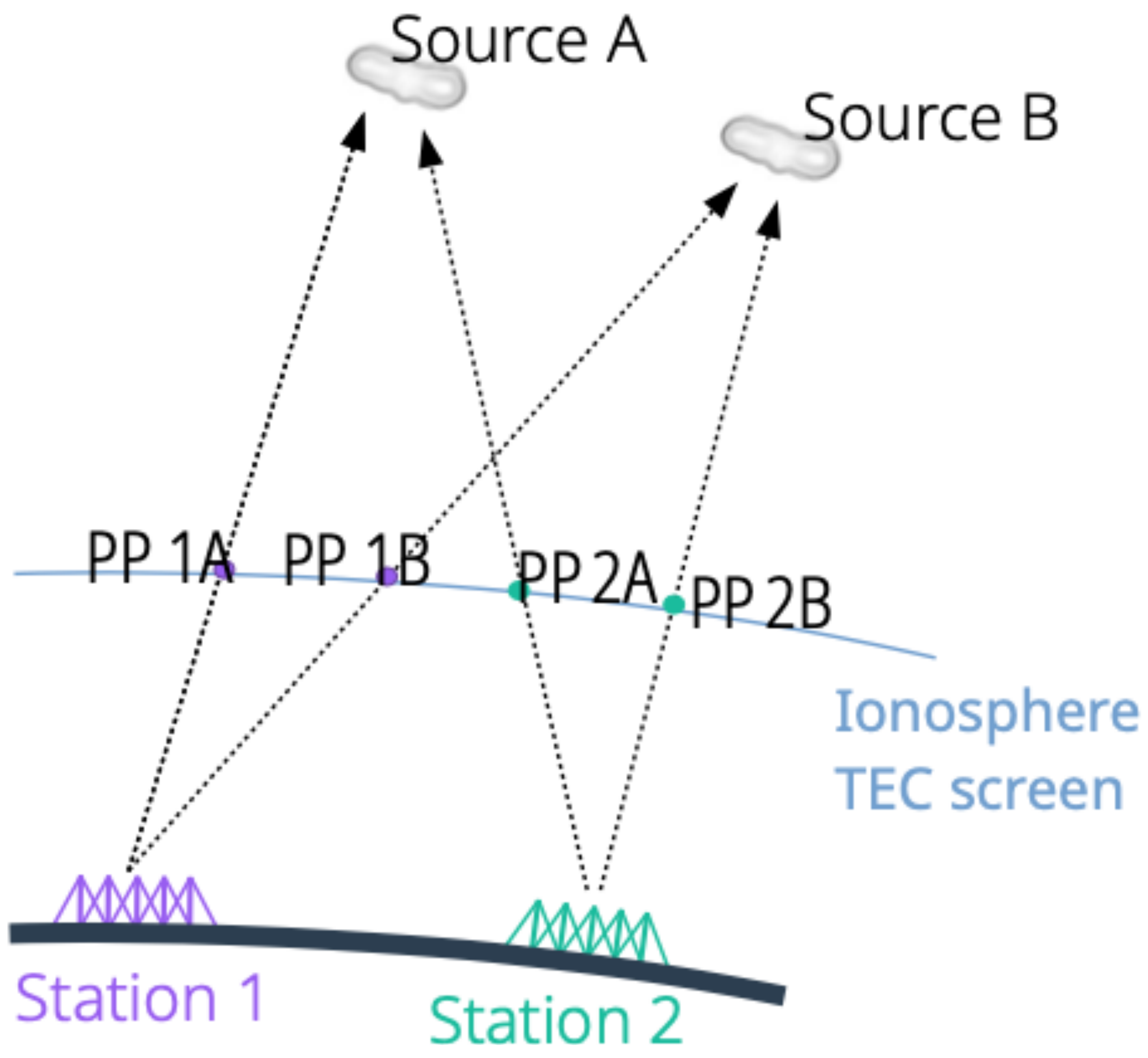
  ➡ TEC-Screen

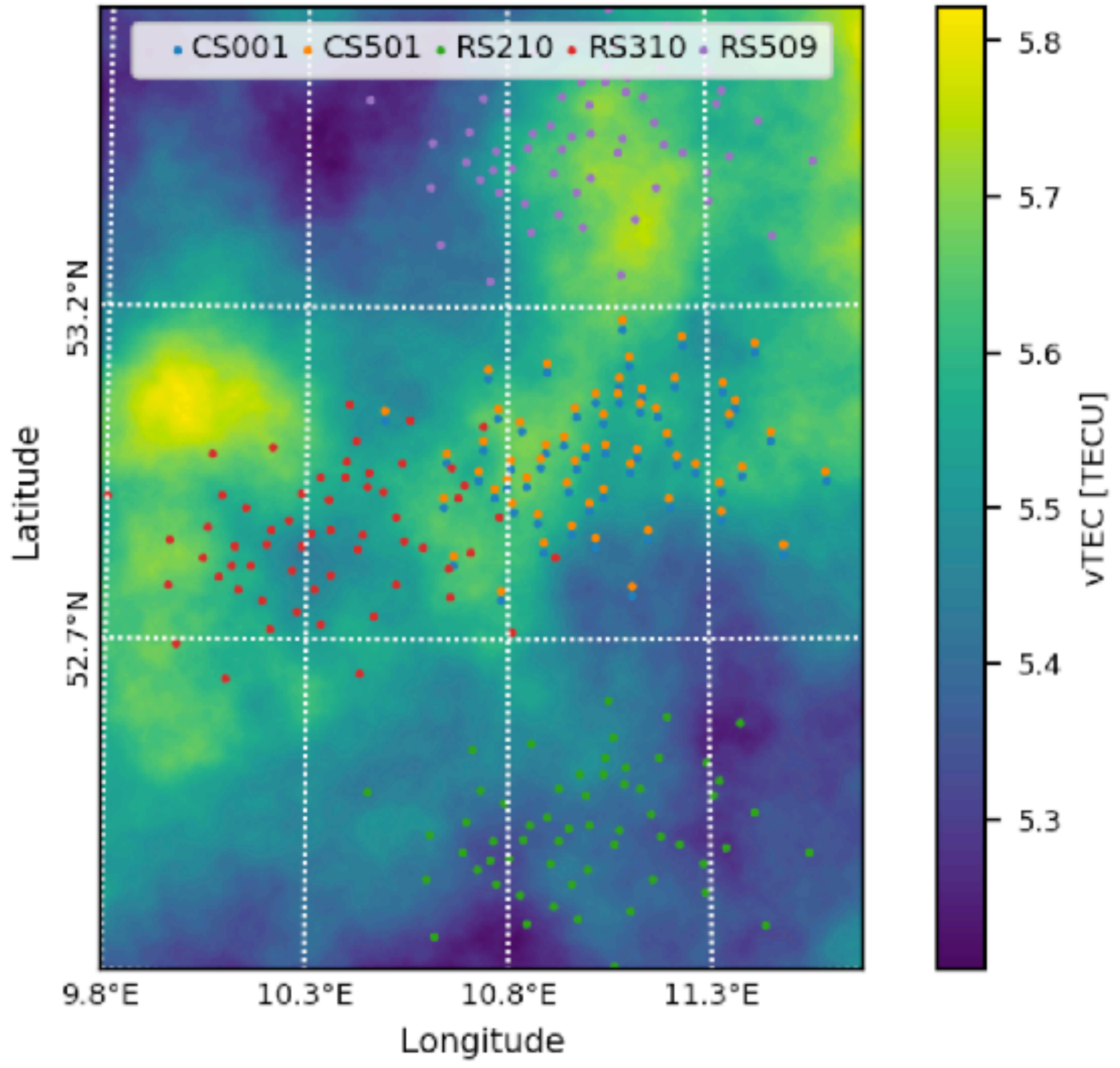
- Frozen turbulence

Image credits: A. Corstanje, F. Sweijen, C. Van Eck, P. Zucca

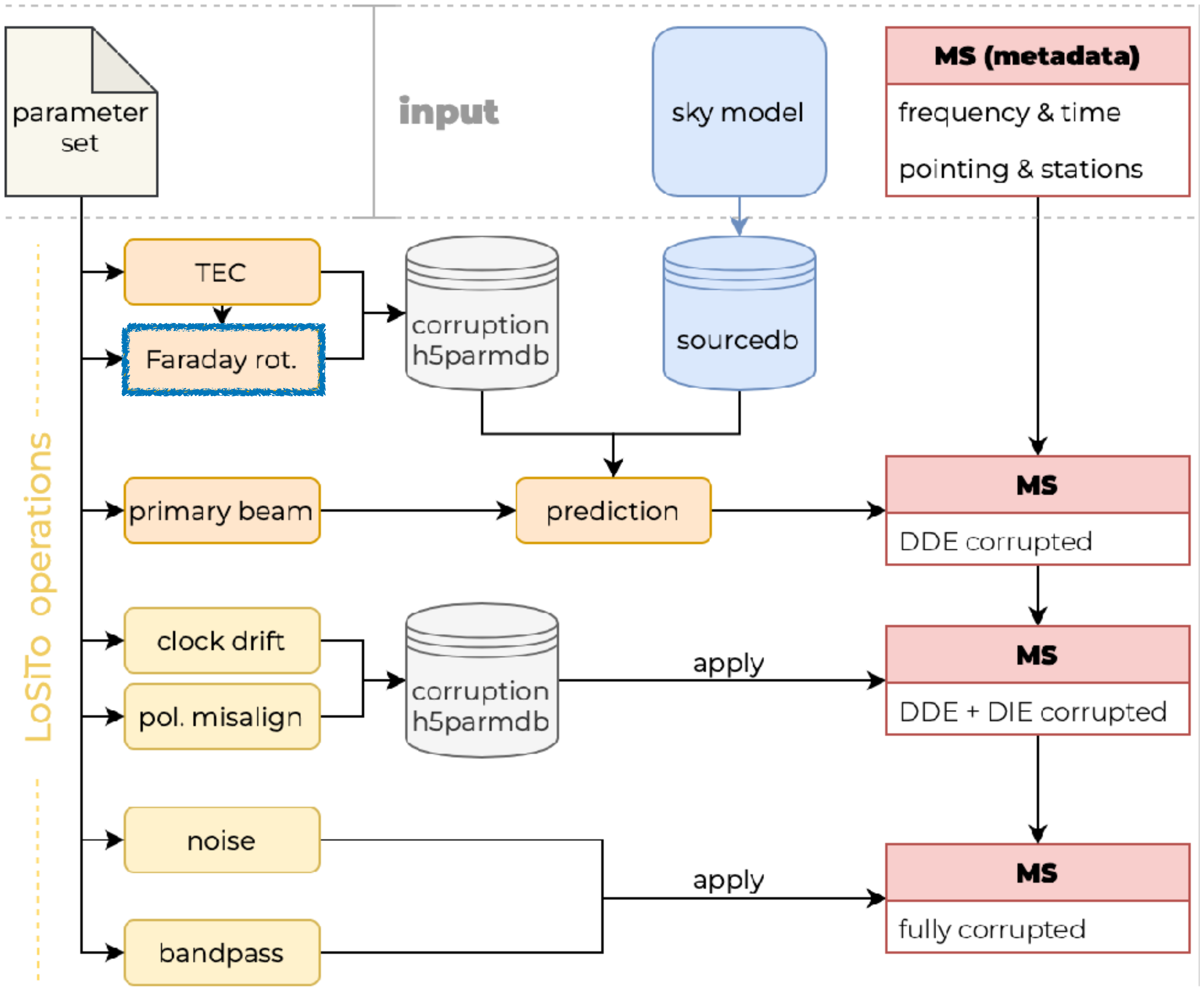
Scheme of LoSiTo







Scheme of LoSiTo



- Rotation by angle β in plane of linear polarization

$$\beta_{\mathrm{rot}} = RM \cdot \left(\frac{c}{\nu}\right)^2$$

- Need RM values

$$RM = 2.62 \cdot 10^{-13}\mathrm{T}^{-1} \cdot TEC \cdot \hat{d} \cdot \overrightarrow{B}$$

- Use RMExtract implementation of World Magnetic Model

Scheme of LoSiTo

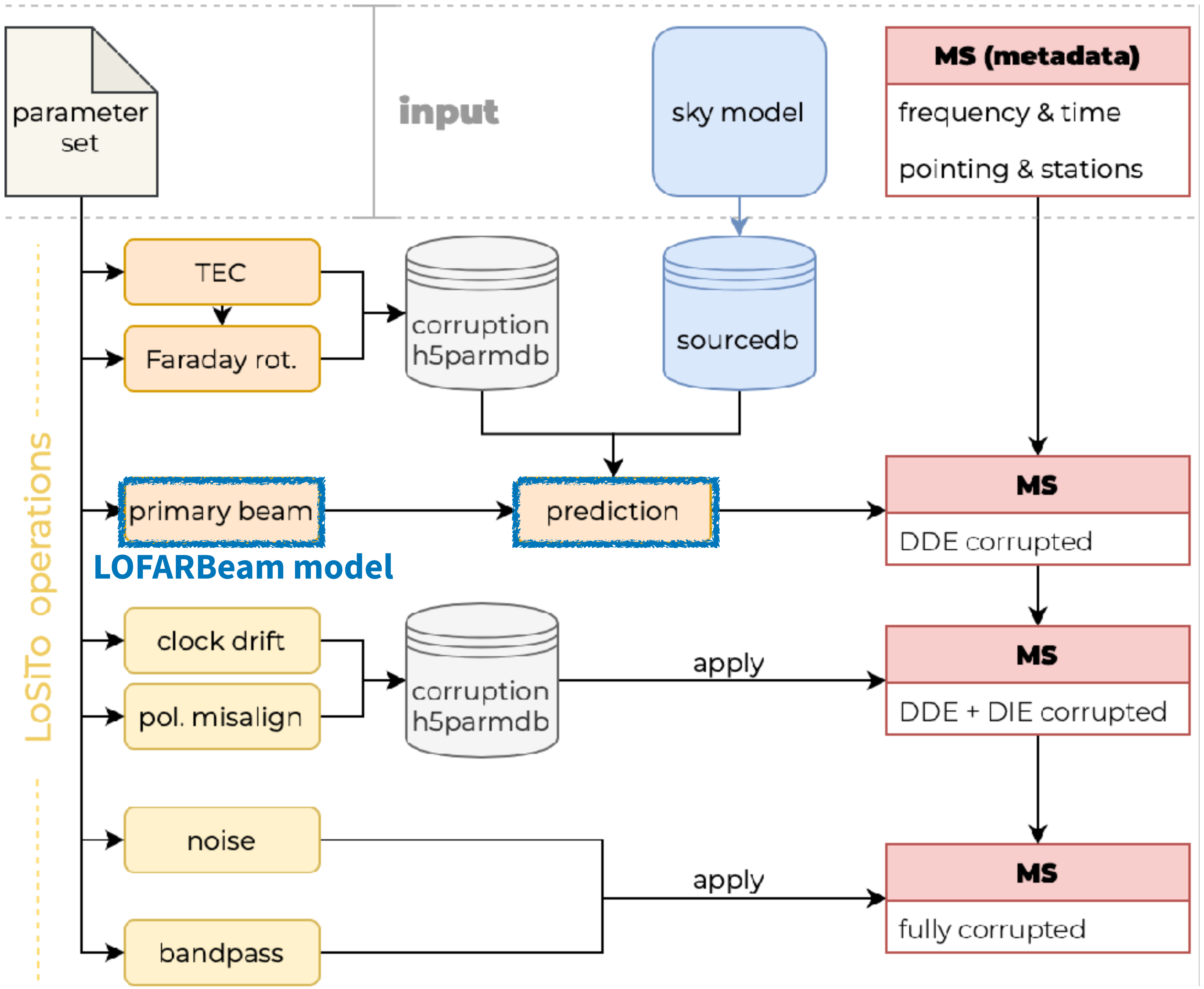
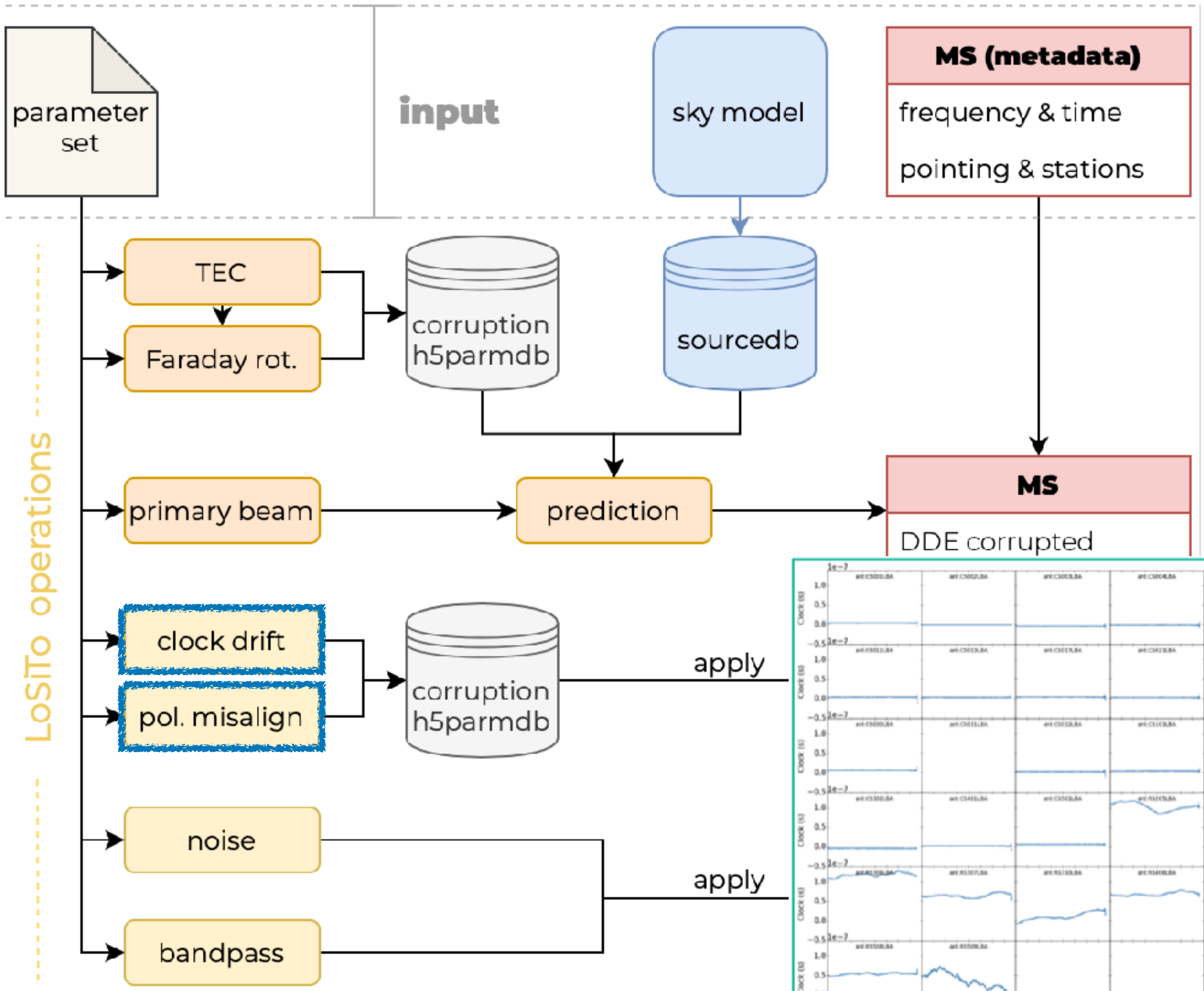
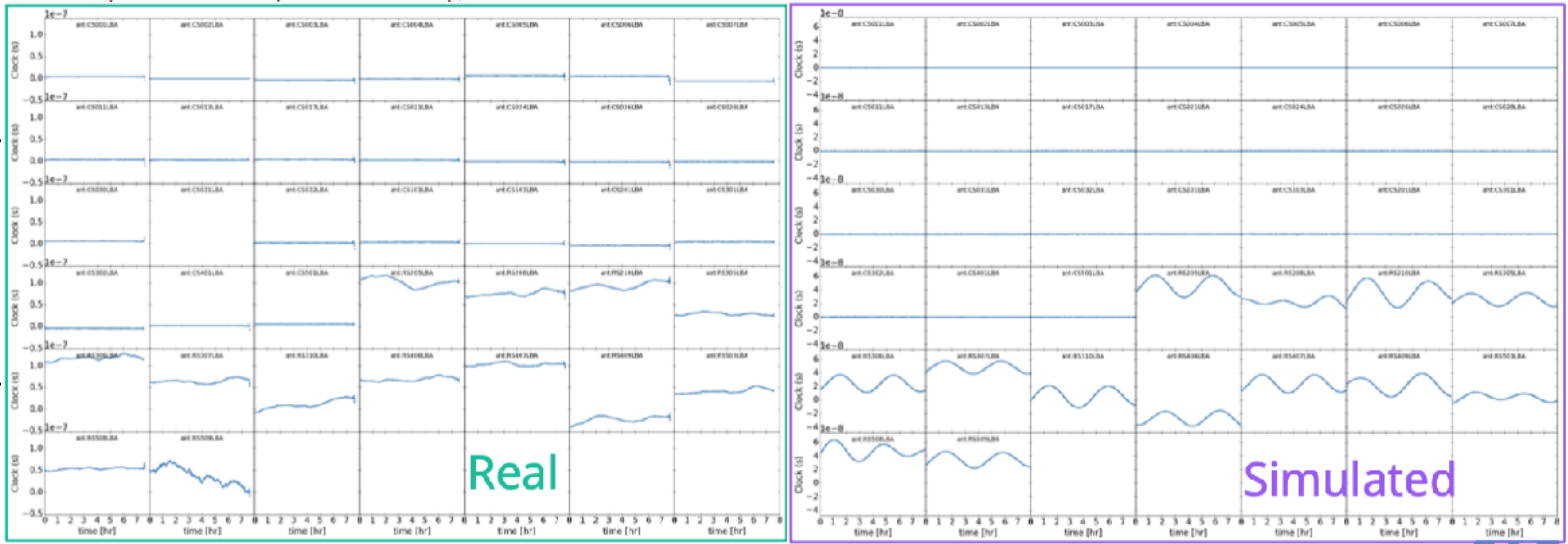


Image credits: A. Corstanje, F. Sweijen, C. Van Eck, P. Zucca

Scheme of LoSiTo



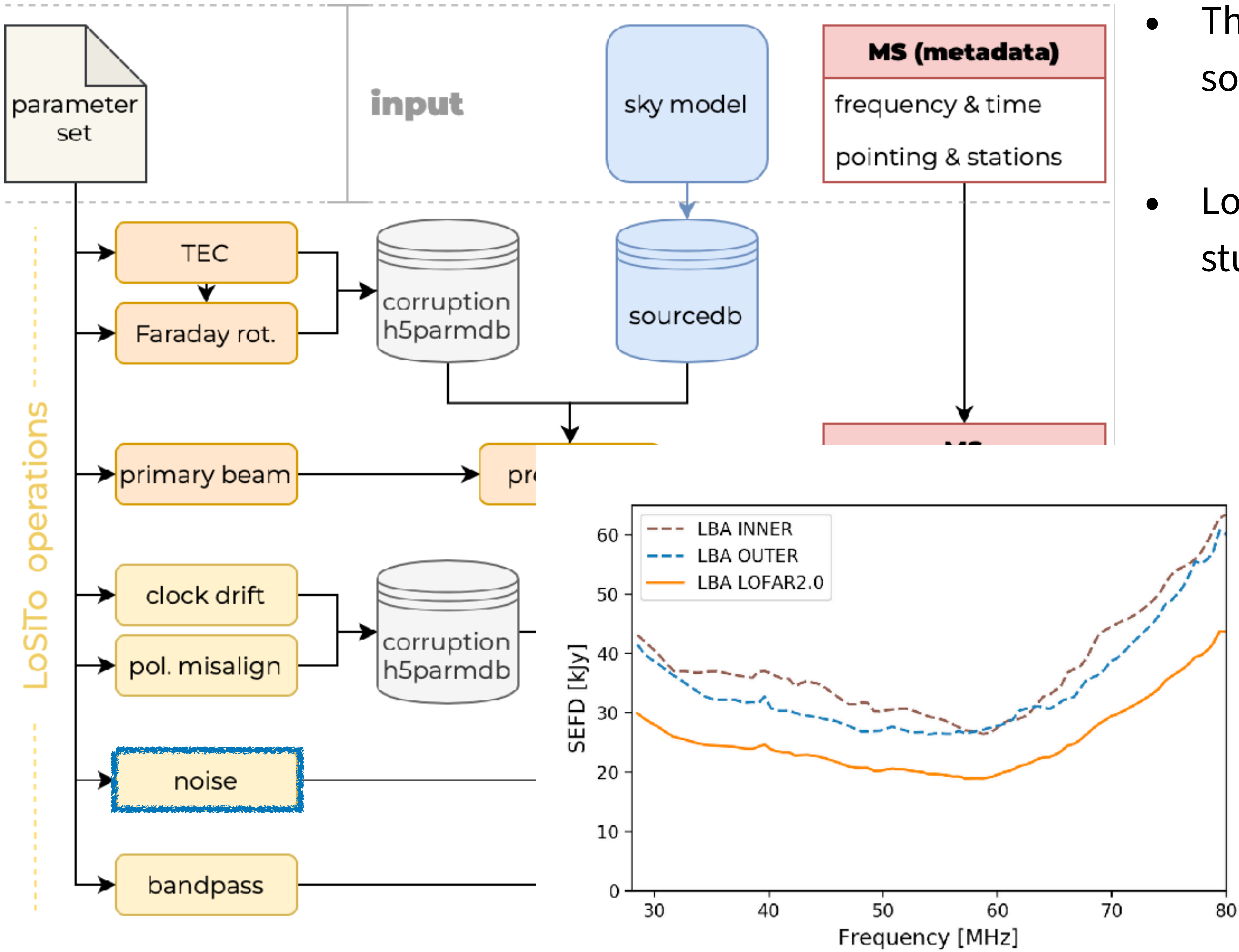
- Clock drift: simple random-seeded sinusoidal model
- Polarization misalingment: random O(ns) delays between polarizations

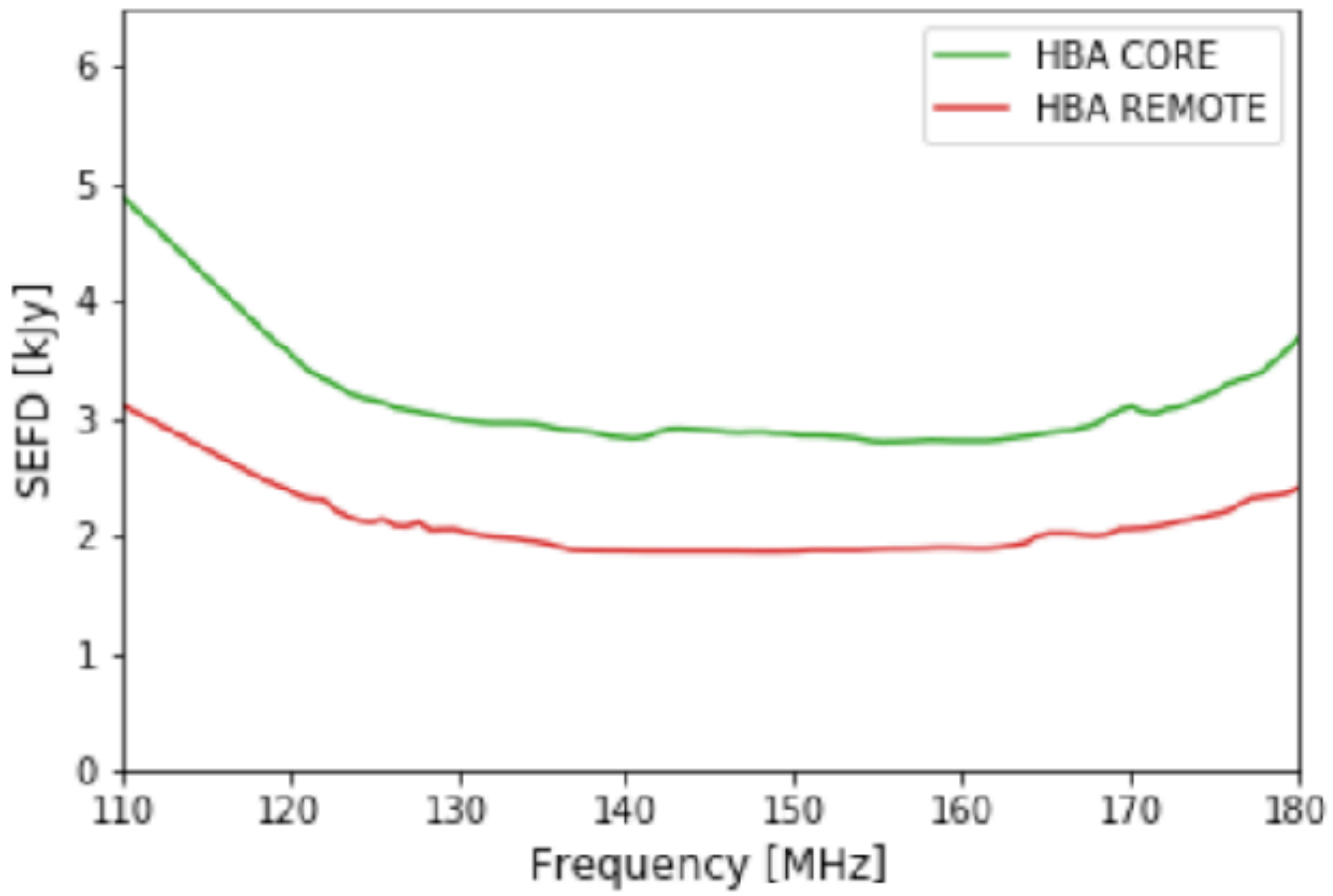


Scheme of LoSiTo

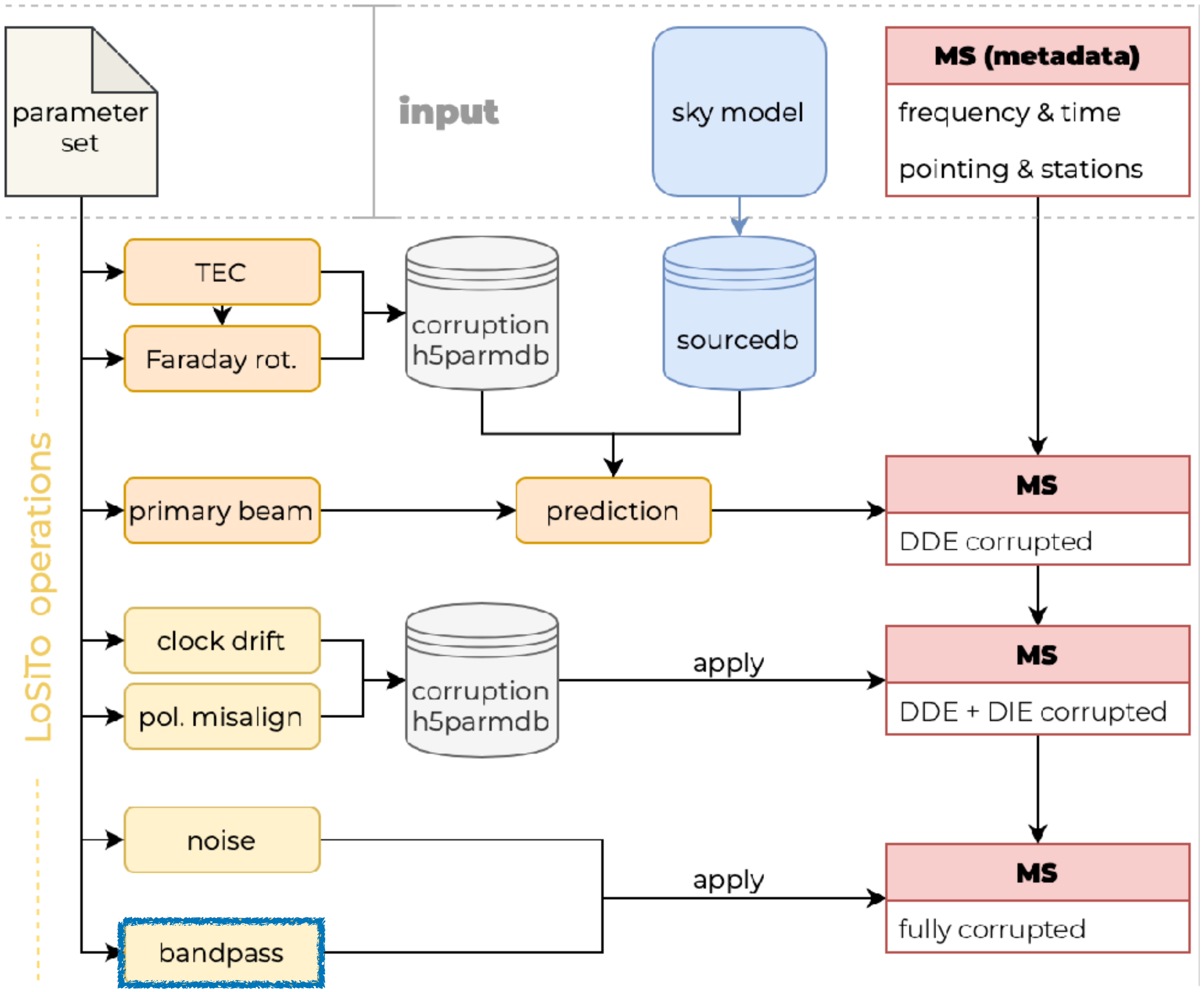
- Thermal noise level can be characterised by source equivalent flux density (SEFD)
- LoSiTo uses SEFD determined in an empirical study of the instrument in van Haarlem+ 2013

$$\Delta S_{ij} = \frac{1}{\eta}\sqrt{\frac{SEFD_i SEFD_j}{2\Delta\nu\tau}}$$

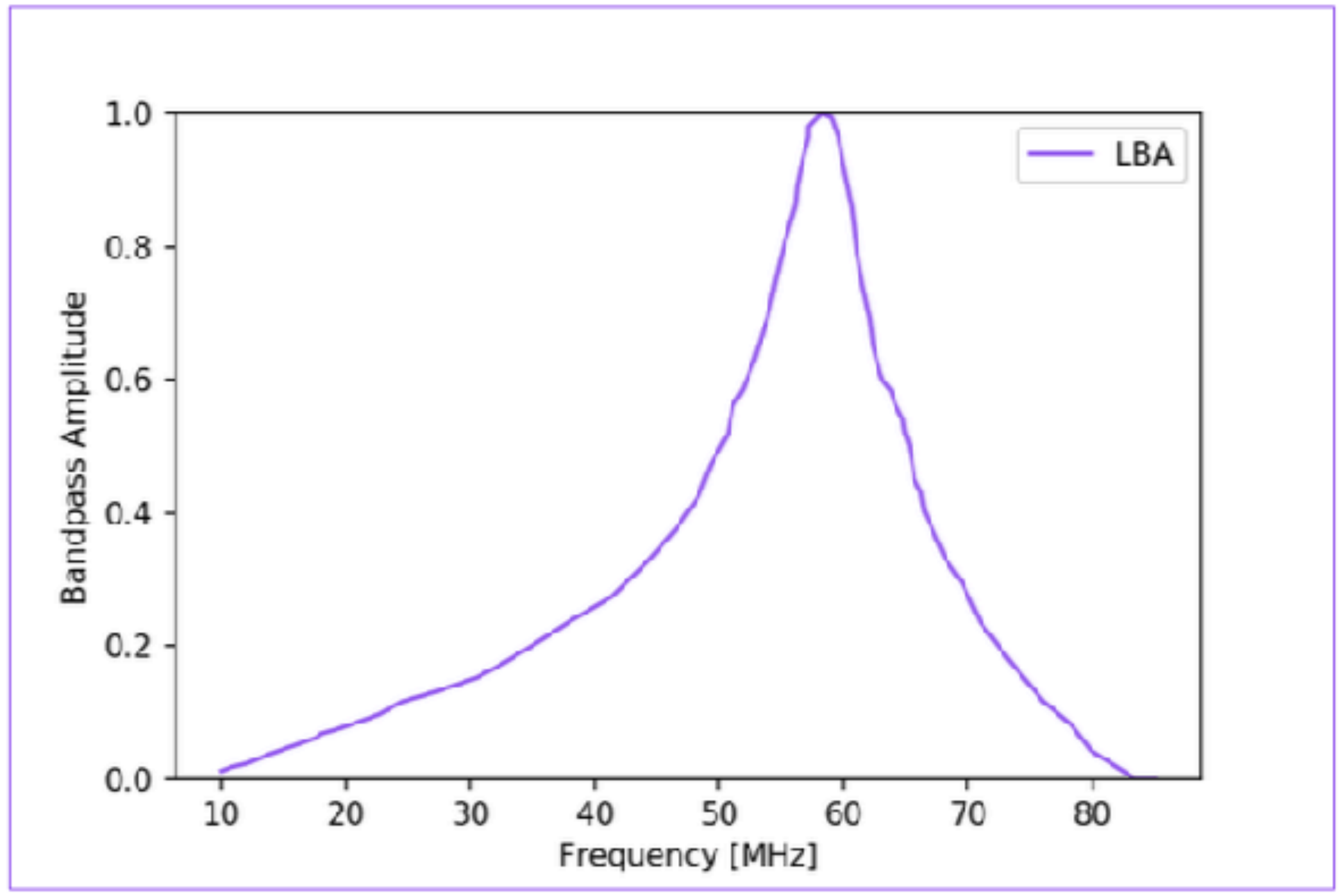


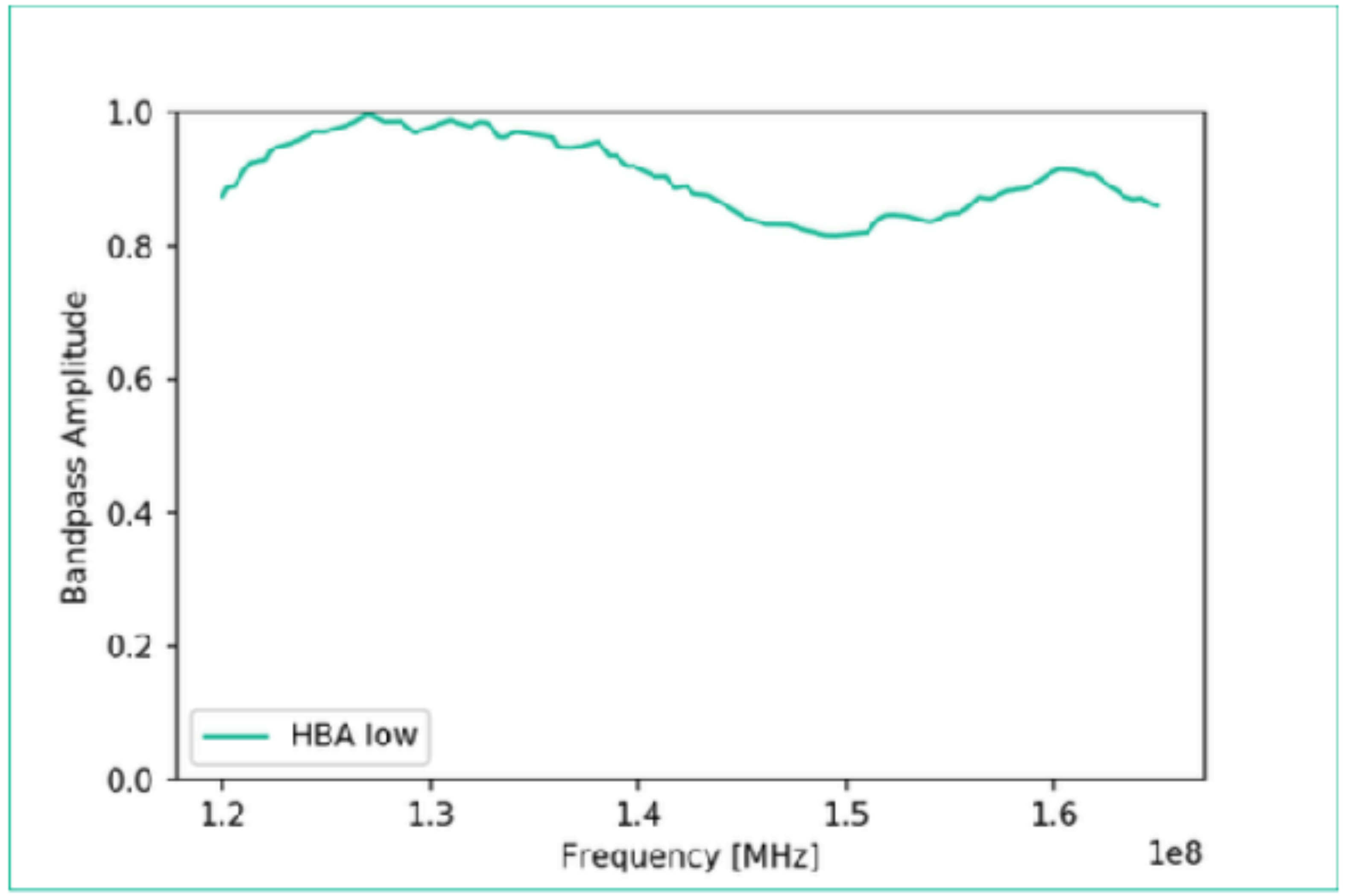


Scheme of LoSiTo



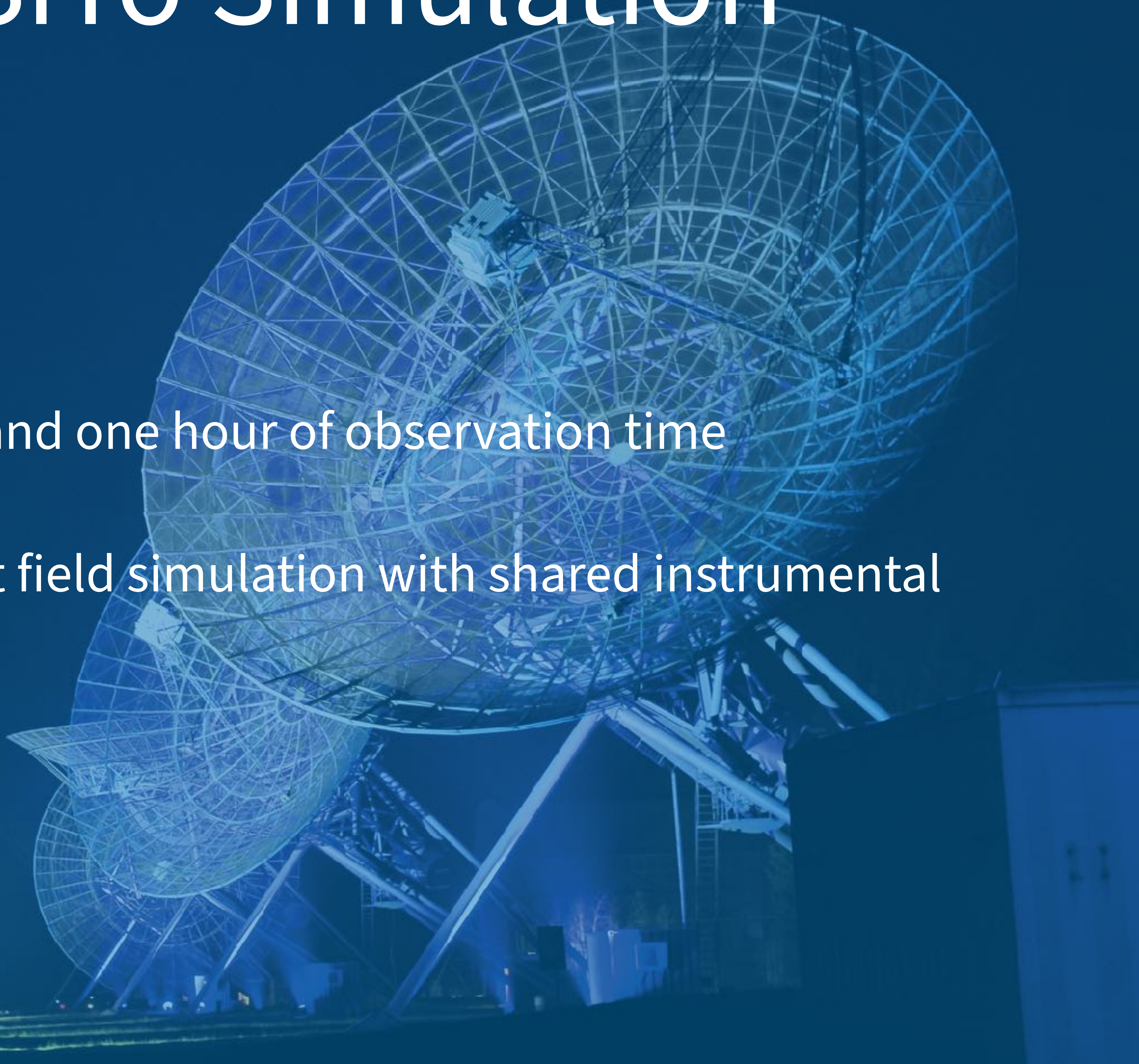
Use average bandpass from van Haarlem et al. 2013





How to run a LoSiTo Simulation

- Simulation of one single Sub-band and one hour of observation time
- Parallel calibrator (3c196) and target field simulation with shared instrumental effects



Prepare the data

- If you find time during this LOFAR data school (or afterwards), you can follow the LoSiTo steps on the next slides.
- You will need the singularity image LDS21_systematics+simulations.simg
    - If you do not work on CEP3 but have trouble with the above, you can download an alternative one here
    - If you work on CEP3, then use `/data018/scratch/LDS2021/soft_images/lds-losito-cep3.sif`
    - Or maybe you rebuilt it already for yourself for the LoSoTo session
- Get and unpack the losito.tar from the slack channel
- Enter the singularity and navigate to `losito/cal`

Input: Measurement Set

- We can use the `synthms` script to create empty measurement sets for a single sub-band at 140 MHz.

- This should take around a minute, however it seems to be much slower from inside the singularity. If you don't want to wait, you can also find the files here.

  - `synthms --name 3c196 --tobs 1.0 --ra 2.75570763 --dec 0.61219363 --start 5037073202 --lofarversion 1 --minsb 250 --maxsb 250 --station HBA`

`3c196_t201806301100_SBH250.MS`

- Any LOFAR MS can be used as input, it is also possible to use multiple MS in one simulation

- We use a sky model with four Gaussian components for the calibrator 3c196

Input: Parameter Set

1. Options for direction-dependent corruptions

2. Options for instrumental delays

3. Predict the visibilities - this is done for each source independently since they all have different corruptions!

4. Add noise and scale by average bandpass

```
#LoSiTo parset
##### global #######
msin = 3c196*.MS
skymodel = 3c196.sky

######### DIRECTION DEPENDENT STEPS  #######
## get simulated total electron content
[tec]
operation = TEC
method = turbulence
hIon = 250e3 # height of ionosphere layer
maxdtec = 0.1
maxvtec = 7.0
alphaIon = 3.89 # slope of structure function
seed = 26
angRes = 240 # arcseconds

# get corresponding Faraday rotation
[faraday]
operation=FARADAY
hIon = 250e3

# Add beam effects (array_factor+element)
[beam]
operation = BEAM
mode = default

######### DIRECTION INDEPENDENT STEPS #######
# clock drift
[clock]
operation=CLOCK
mode = lofar1
seed = 25

# X-Y delay
[polmisalign]
operation=POLMISALIGN
seed = 25

######### PREDICTION #######
# Do the predict
[predict]
operation = PREDICT
outputColumn = DATA
resetWeights = False
predictType = h5parmpredict

# Add noise to the predicted visibilities
[noise]
operation = NOISE
outputColumn = DATA

# Apply the bandpass
[bandpass]
operation = BANDPASS
method = ms
```

Run the simulation

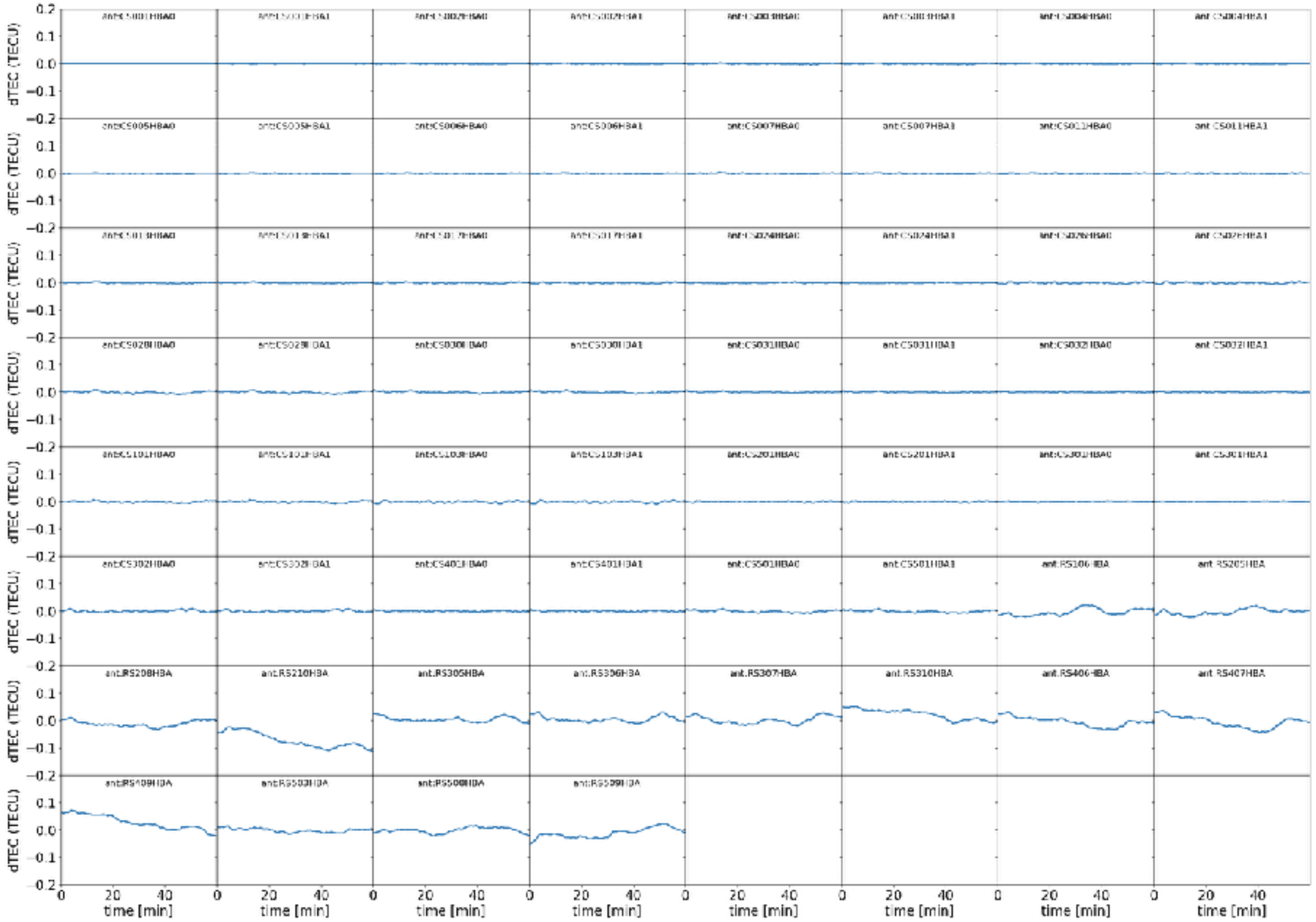
- Execute losito using the parset in the tgts folder
  - `losito sim_cal.parset`
- This might take a few minutes (for large sky models, this can become very time-expensive).
- The simulation will store the corruptions into a h5parm called ‘`corruptions.h5`’

6th LOFAR Data School

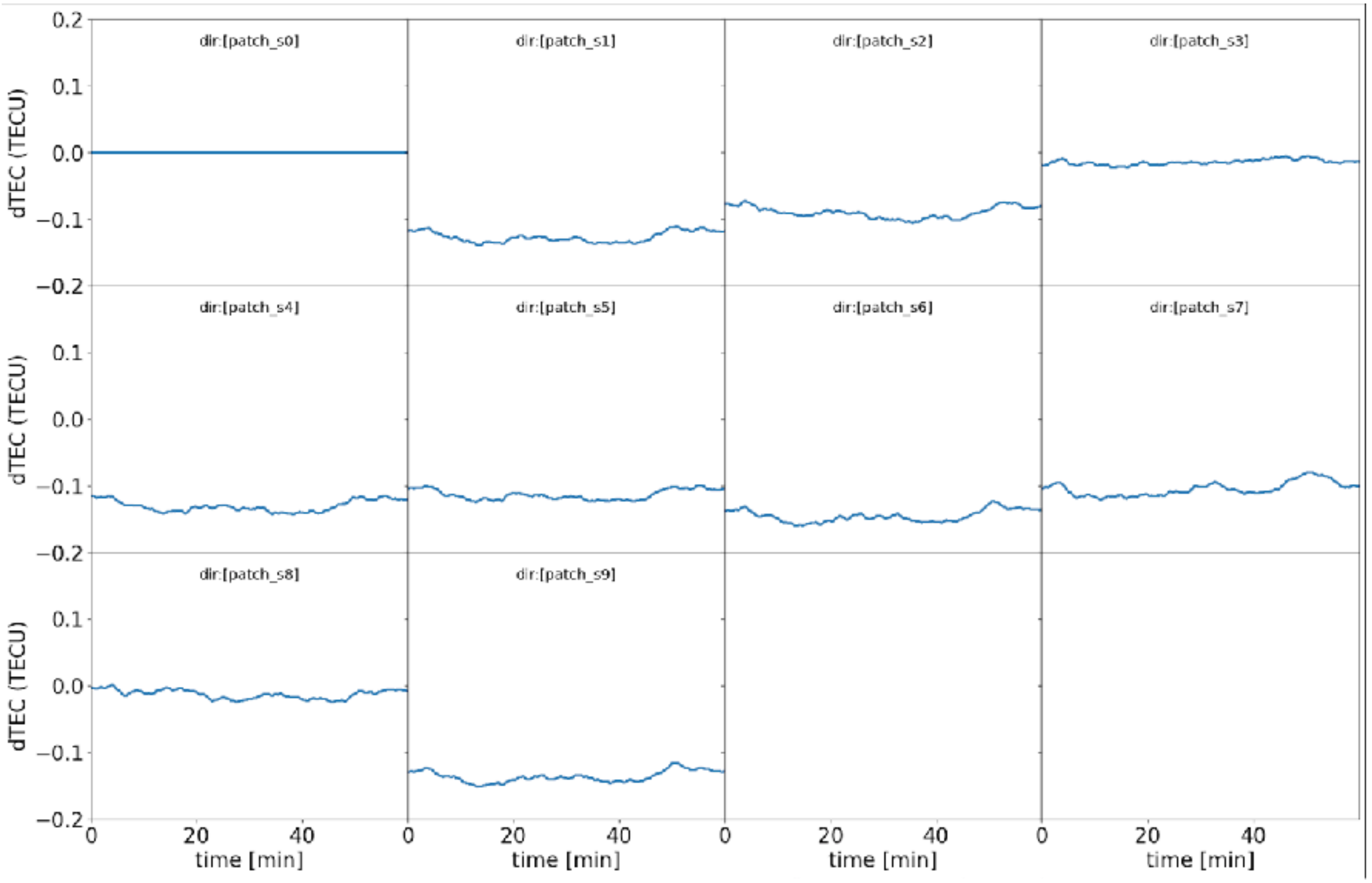
Plot the simulated corruptions

- Use losoto to see what corruptions we added to our simulation (the results will be in a new directory ‘plots’)
  - `losoto corruptions.h5 plot.parset`

`tec_dir[patch_s0].png`



`tec_antCS001HBA0.png`



Next: Simulate the Target Observation

- We want the target observation to have the same instrumental delays, but a different ionosphere!
  - Possible to share the same instrumental systematics either via same random seed or by using the same corruptions h5parm
- Enter the tgts directory
- Again, create the empty MS (now with a different pointing direction!)
  - `synthms --name target --tobs 1.0 --ra 2.75570763 --dec 0.61219363 --start 5037073202 --lofarversion 1 --minsb 250 --maxsb 250 --station HBA`
- Now we need a model for our target sky. We can use the sky model script in LoSiTo:
  - `skymodel --output random.sky --kind random --radius 1 --maxflux 10 --minflux 0.4  --nptsrc 4 --ngauss 6 target_t201806301100_SBH250.MS`

```
skymodel - generate simple sky models

positional arguments:
  msfile                Input MS filename.

optional arguments:
  -h, --help            show this help message and exit
  --output OUTPUT       Ouput sky model filename.
  --kind KIND           Kind of sky model: cross, spiral, tree, random or cat.
  --radius RADIUS       Radius in degrees within which sources are added.
  --nptsrc NPTSRC       Number of point sources to generate.
  --ngauss NGAUSS       Number of Gaussians to generate (kind=random only).
  --minflux MINFLUX     Minimum peak flux density in Jy (kind=random only).
  --maxflux MAXFLUX     Maximum peak flux density in Jy (kind=random only).
  --minmaj MINMAJ       Minimum major axis in arcsec of Gaussian sources.
  --maxmaj MAXMAJ       Maximum major axis in arcsec of Gaussian sources.
  --maxell MAXELL       Maximum ellipticity (1 - maj/min) of Gaussian sources.
  --minspidx MINSPIDX   Minimum spectral index of sources (kind=random only).
  --maxspidx MAXSPIDX   Maximum spectral index of sources (kind=random only).
  --logsi LOGSI         Value for LogarithmicSI: true or false (kind=random
```

Next: Simulate the Target Observation

- We want the target observation to have the same instrumental delays, but a different ionosphere!
  - Possible to share the same instrumental systematics either via same random seed or by using the same corruptions h5parm
- Enter the tgts directory
- Again, create the empty MS (now with a different pointing direction!)
  - `synthms --name target --tobs 1.0 --ra 2.75570763 --dec 0.61219363 --start 5037073202 --lofarversion 1 --minsb 250 --maxsb 250 --station HBA`
- Now we need a model for our target sky. We can use the sky model script in LoSiTo:
  - `skymodel --output random.sky --kind random --radius 1 --maxflux 10 --minflux 0.4  --nptsrc 4 --ngauss 6 target_t201806301100_SBH250.MS`

```
format = Name, Type, Patch, Ra, Dec, I, Q, U, V, MajorAxis, MinorAxis, Orientation, ReferenceFrequency, SpectralIndex, LogarithmicSI , , patch_s0, 10:32:4.470, +34.19.42.198
s0, POINT, patch_s0, 10:32:4.470, +34.19.42.198, 3.660690555510574, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 139842224.12109375, [-0.719924749485694], true
 , , patch_s1, 10:30:27.211, +35.26.42.508
s1, POINT, patch_s1, 10:30:27.211, +35.26.42.508, 0.536333497929252, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 139842224.12109375, [-0.9927144207444613], true
 , , patch_s2, 10:32:51.891, +35.30.12.908
s2, POINT, patch_s2, 10:32:51.891, +35.30.12.908, 1.1866836637889666, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 139842224.12109375, [-0.7403354973719278], true
 , , patch_s3, 10:30:53.500, +34.13.16.683
s3, POINT, patch_s3, 10:30:53.500, +34.13.16.683, 5.592552285786197, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 139842224.12109375, [-0.927459157120474], true
 , , patch_s4, 10:30:45.221, +35.33.7.442
s4, GAUSSIAN, patch_s4, 10:30:45.221, +35.33.7.442, 1.364124239861554, 0.0, 0.0, 0.0, 5.340964958065337, 4.279113952555738, 119.73487817386221, 139842224.12109375, [-0.9868282279437893], true
 , , patch_s5, 10:30:54.334, +35.22.59.793
s5, GAUSSIAN, patch_s5, 10:30:54.334, +35.22.59.793, 0.4134047336000961, 0.0, 0.0, 0.0, 7.455803174337176, 6.906028826612832, 150.54376159032932, 139842224.12109375, [-1.0698657380331091], true
 , , patch_s6, 10:29:35.537, +35.29.8.063
s6, GAUSSIAN, patch_s6, 10:29:35.537, +35.29.8.063, 7.563825266823199, 0.0, 0.0, 0.0, 29.457967101990675, 24.91153017619323, 65.8094013363831, 139842224.12109375, [-0.6031383504255526], true
 , , patch_s7, 10:28:41.165, +34.47.20.160
s7, GAUSSIAN, patch_s7, 10:28:41.165, +34.47.20.160, 1.0003143196814492, 0.0, 0.0, 0.0, 16.452706618184813, 15.475174397088347, 125.5408935943402, 139842224.12109375, [-0.9007004367549055], true
 , , patch_s8, 10:33:29.328, +34.46.23.258
s8, GAUSSIAN, patch_s8, 10:33:29.328, +34.46.23.258, 0.46248683496391374, 0.0, 0.0, 0.0, 25.725546304151223, 20.911627151881593, 35.57113424201169, 139842224.12109375, [-0.8697182066458602], true
 , , patch_s9, 10:29:42.079, +35.23.49.583
s9, GAUSSIAN, patch_s9, 10:29:42.079, +35.23.49.583, 0.72427972413862, 0.0, 0.0, 0.0, 20.952852295539117, 17.969124888972086, 44.658295132185046, 139842224.12109375, [-0.6957328830502183], true
```

Run the simulation

- Execute losito using the parset in the tgts folder
    - `losito sim_target.parset`
- This parset is very similar, but uses a different seed for the ionosphere!
- Again, plot the corruptions and compare them to the calibrator simulation
    - `losoto corruptions.h5 plot.parset`

A few more things to try…

- If you want to play a bit more with the simulated data, you can look at the simulation in image space (see `run_wsclean.sh`)
- You can play with the different corruptions by changing the parset. How does the image change if the ionospheric effects are left out? What is the impact of the instrumental delays?
- **Code**: github.com/darafferty/losito
- **Documentation**: https://www.astron.nl/citt/losito/
- (If you want to leave out both the ionosphere and the instrumental delays, you have to change the predictType to 'predict' since no corruptions will be applied during prediction)

LoSiTo: Limitations

1. RFI
2. Assumes perfect beam model
3. No 3rd order ionospheric terms (relevant <40 MHz)
4. Bright sources in side-lobes
5. Very extended Galactic emission

Sneak preview: Simulating LOFAR 2.0

- LoSiTo features simulations of LOFAR 2.0!
  - Use 96 instead of 48 LBA dipoles

    ➡Less noise

    ➡Different primary beam
  - Simultaneously observe with LBA and HBA

    ➡Features simultaneous observations with shared ionosphere
  - New distributed clock signal