



anr[®] | SCInf



Realistic simulations for precision cosmology with Type Ia Supernovae for the LSST era

Luna Dellazzeri

Under the supervision of Mickaël Rigault and Florian Ruppin

DESC SNIa Cosmology Collab

Simulation pipeline

- *SNANA*: Kessler, Vincenzi et al.
- *LightCurveLynx*: Dai et al.
- *skysurvey*: Rigault, Ginolin, Dellazzeri et al.

Inference pipeline (validation teams)

- *FlowSN*: Boyd, Mandel et al.
- *CIGaRS*: Karchev, Giunta, Trotta et al.
- *Unity*: Rubin et al.
- *BBC*: Kessler et al.
- *Scinf*: Rigault, Trigui et al.



Key Project: DP2 SNIa pipeline validation

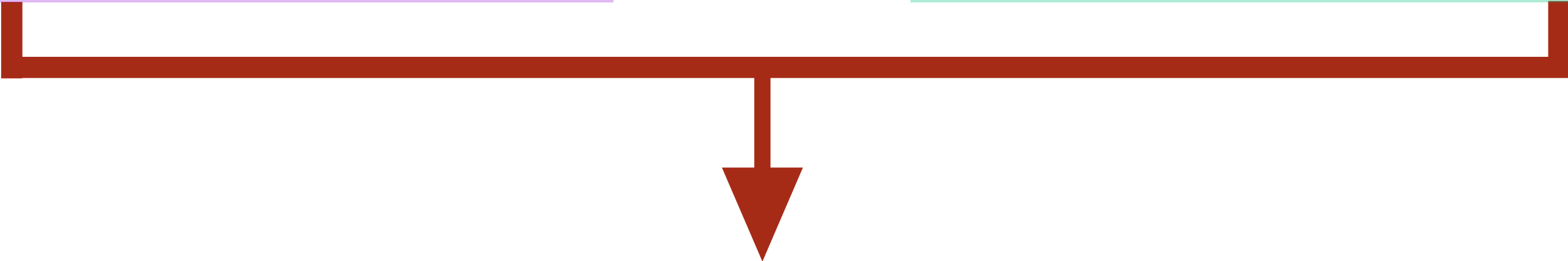
DESC SNIa Cosmology Collab

Simulation pipeline

- *SNANA*: Kessler, Vincenzi et al.
- *LightCurveLynx*: Dai et al.
- *skysurvey*: Rigault, Ginolin, Dellazzeri et al.

Inference pipeline (validation teams)

- *FlowSN*: Boyd, Mandel et al.
- *CIGaRS*: Karchev, Giunta, Trotta et al.
- *Unity*: Rubin et al.
- *BBC*: Kessler et al.
- *Scinf*: Rigault, Trigui et al.



Data challenges

Milestone 1 (ongoing): compare and validate the 3 simulator's outputs for the simplest simulations

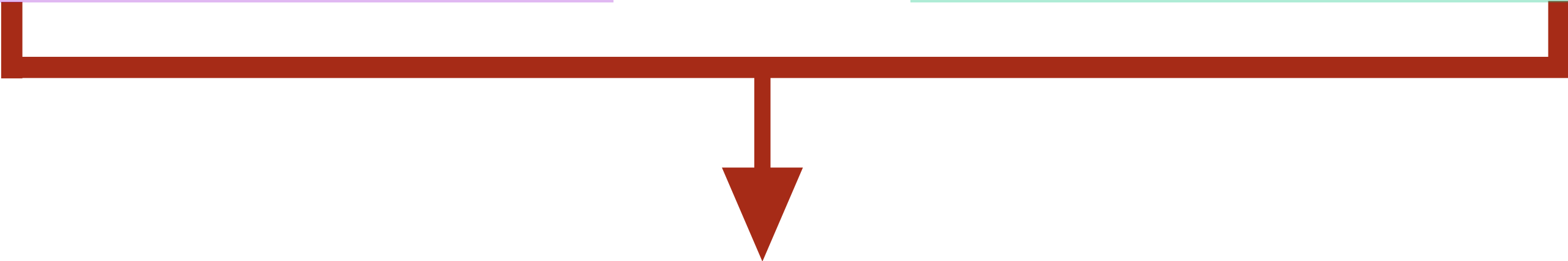
DESC SNIa Cosmology Collab

Simulation pipeline

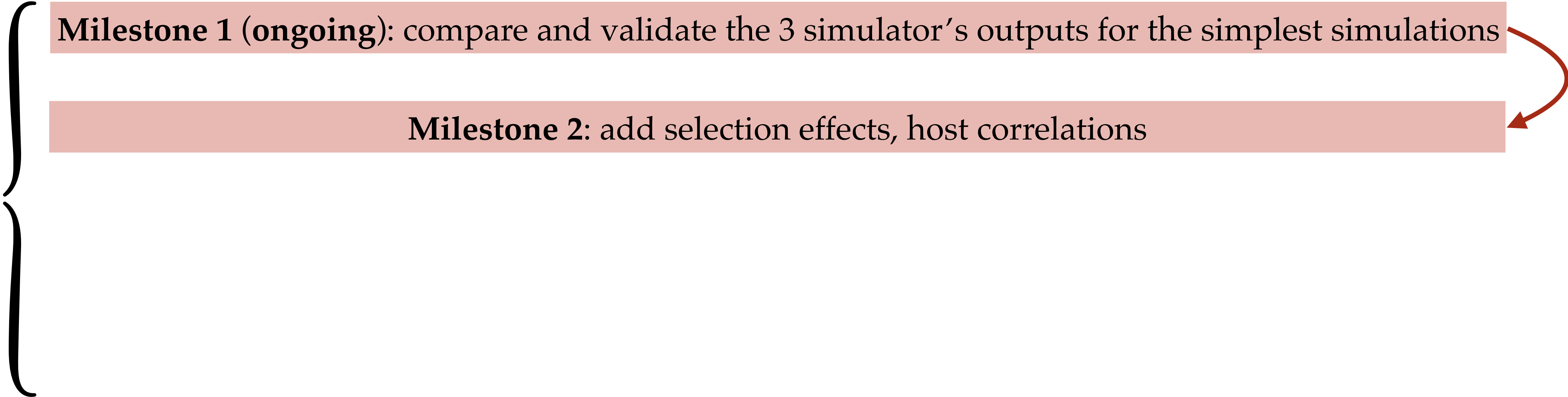
- *SNANA*: Kessler, Vincenzi et al.
- *LightCurveLynx*: Dai et al.
- *skysurvey*: Rigault, Ginolin, Dellazzeri et al.

Inference pipeline (validation teams)

- *FlowSN*: Boyd, Mandel et al.
- *CIGaRS*: Karchev, Giunta, Trotta et al.
- *Unity*: Rubin et al.
- *BBC*: Kessler et al.
- *Scinf*: Rigault, Trigui et al.



Data challenges



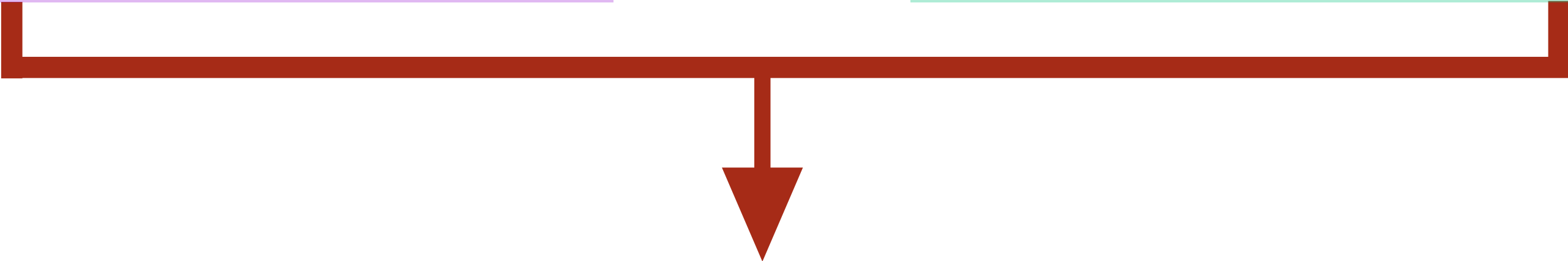
DESC SNIa Cosmology Collab

Simulation pipeline

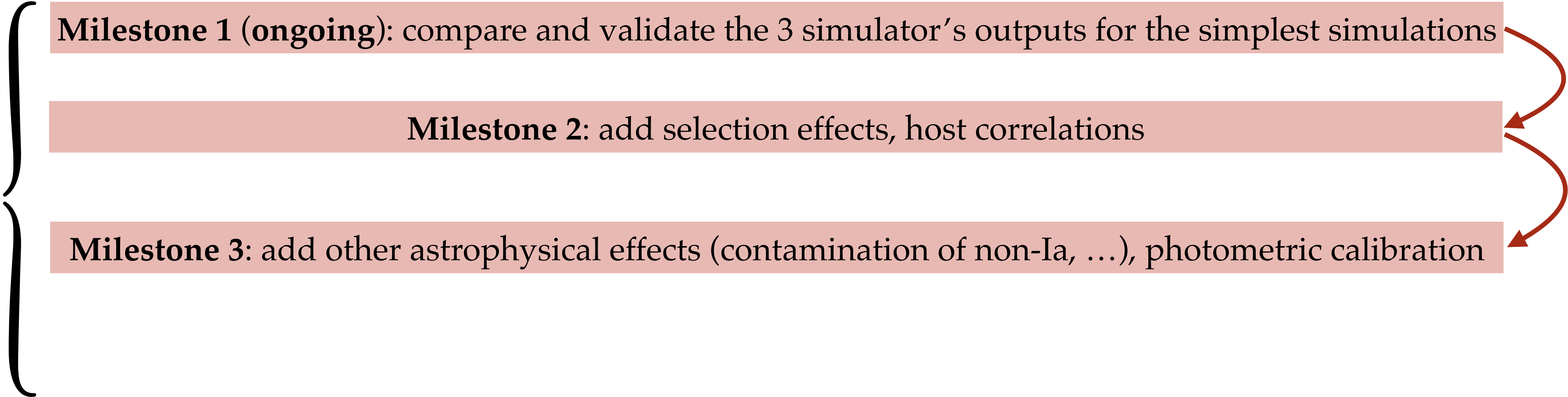
- *SNANA*: Kessler, Vincenzi et al.
- *LightCurveLynx*: Dai et al.
- *skysurvey*: Rigault, Ginolin, Dellazzeri et al.

Inference pipeline (validation teams)

- *FlowSN*: Boyd, Mandel et al.
- *CIGaRS*: Karchev, Giunta, Trotta et al.
- *Unity*: Rubin et al.
- *BBC*: Kessler et al.
- *Scinf*: Rigault, Trigui et al.



Data challenges



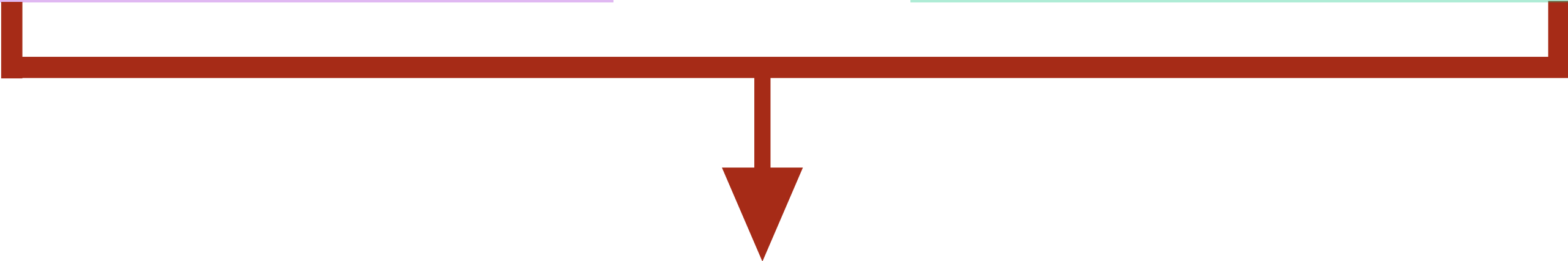
DESC SNIa Cosmology Collab

Simulation pipeline

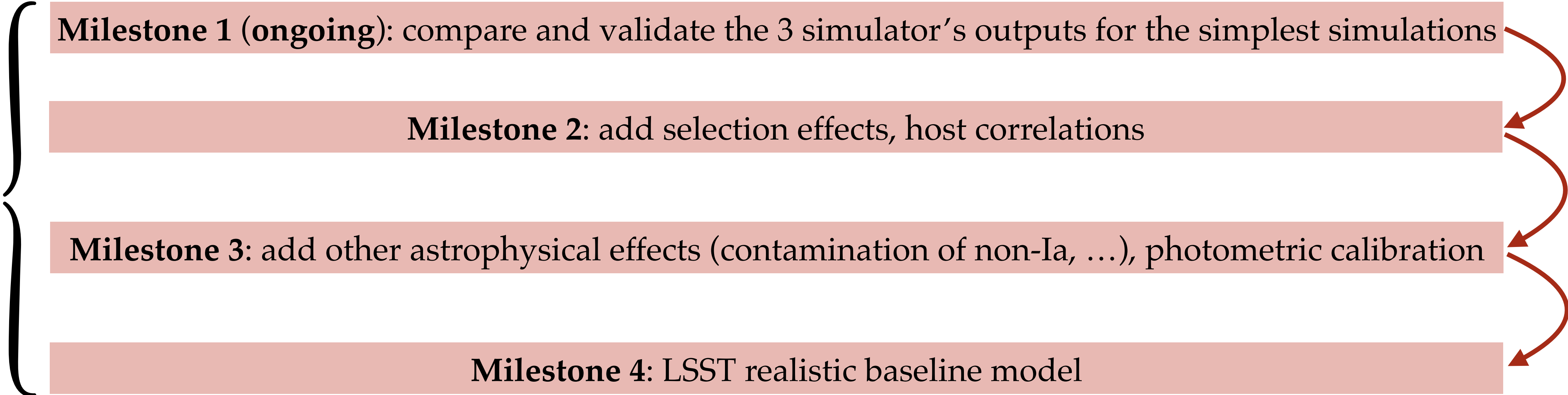
- *SNANA*: Kessler, Vincenzi et al.
- *LightCurveLynx*: Dai et al.
- *skysurvey*: Rigault, Ginolin, Dellazzeri et al.

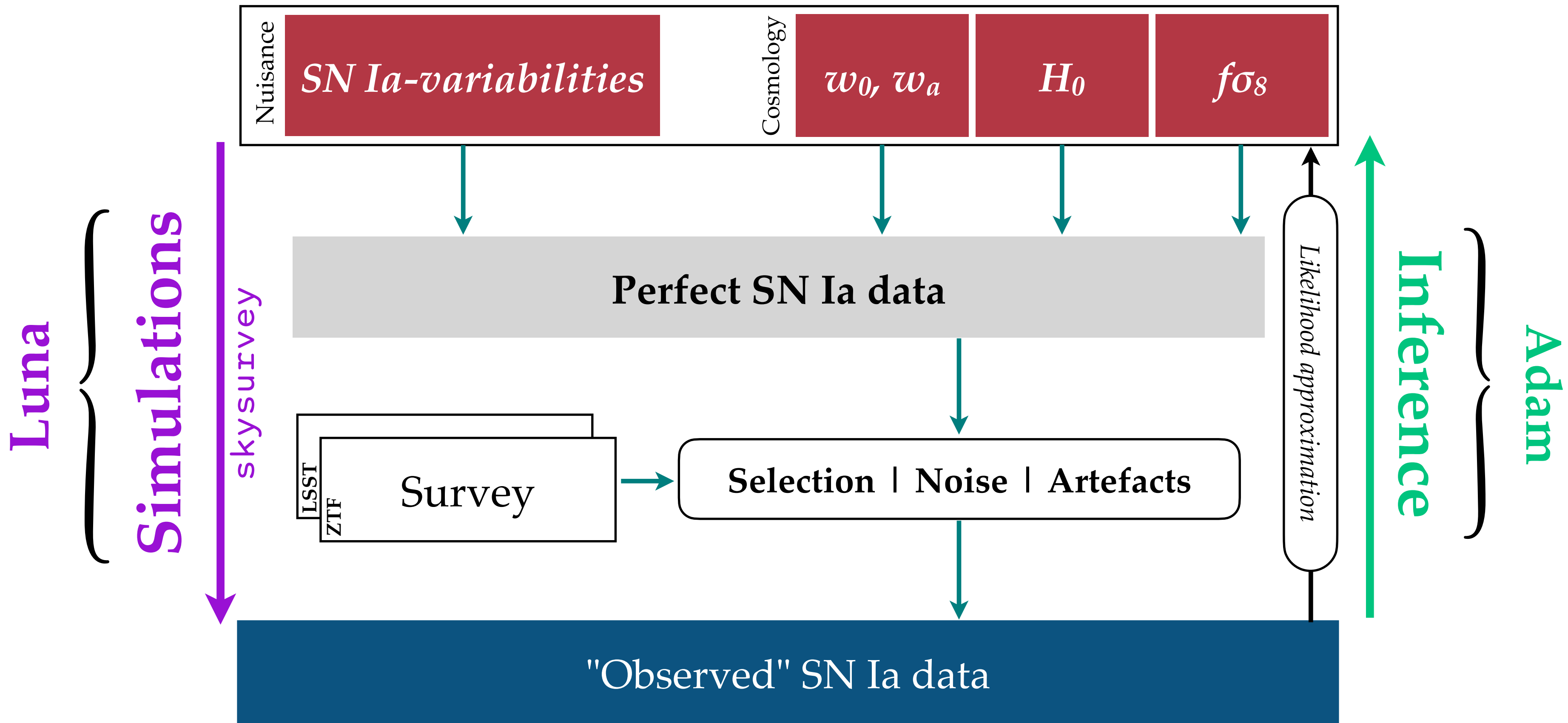
Inference pipeline (validation teams)

- *FlowSN*: Boyd, Mandel et al.
- *CIGaRS*: Karchev, Giunta, Trotta et al.
- *Unity*: Rubin et al.
- *BBC*: Kessler et al.
- *Scinf*: Rigault, Trigui et al.



Data challenges





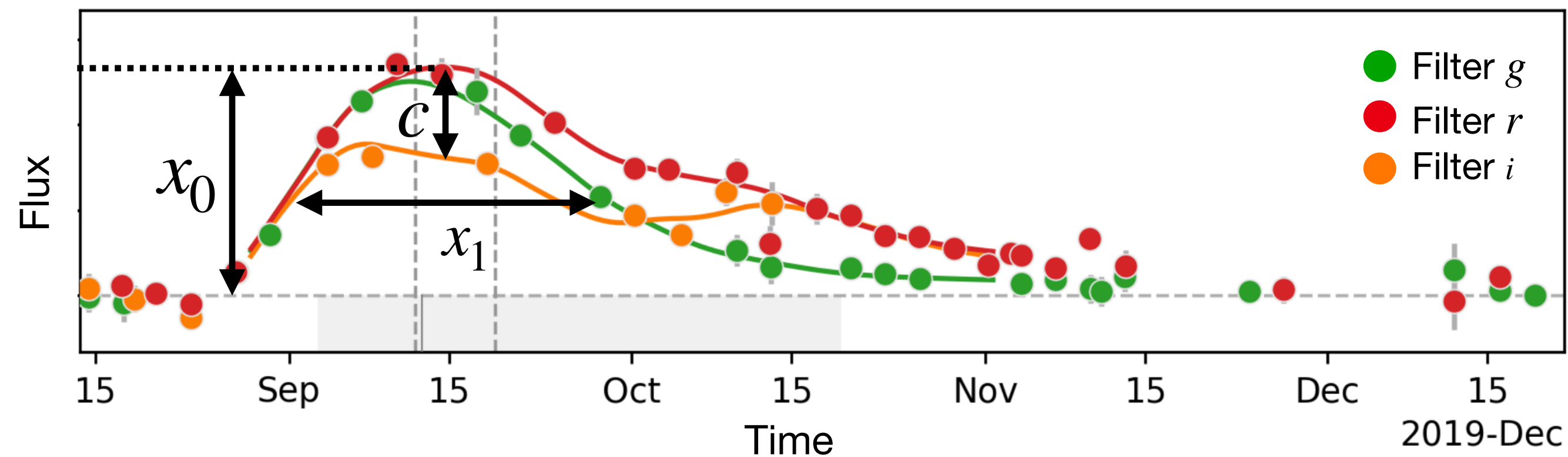
ZTF DR2

Zwicky Transient Facility :

- Located in Mount Palomar Observatory
- Large field of view
- High cadence (30s/ exposure)

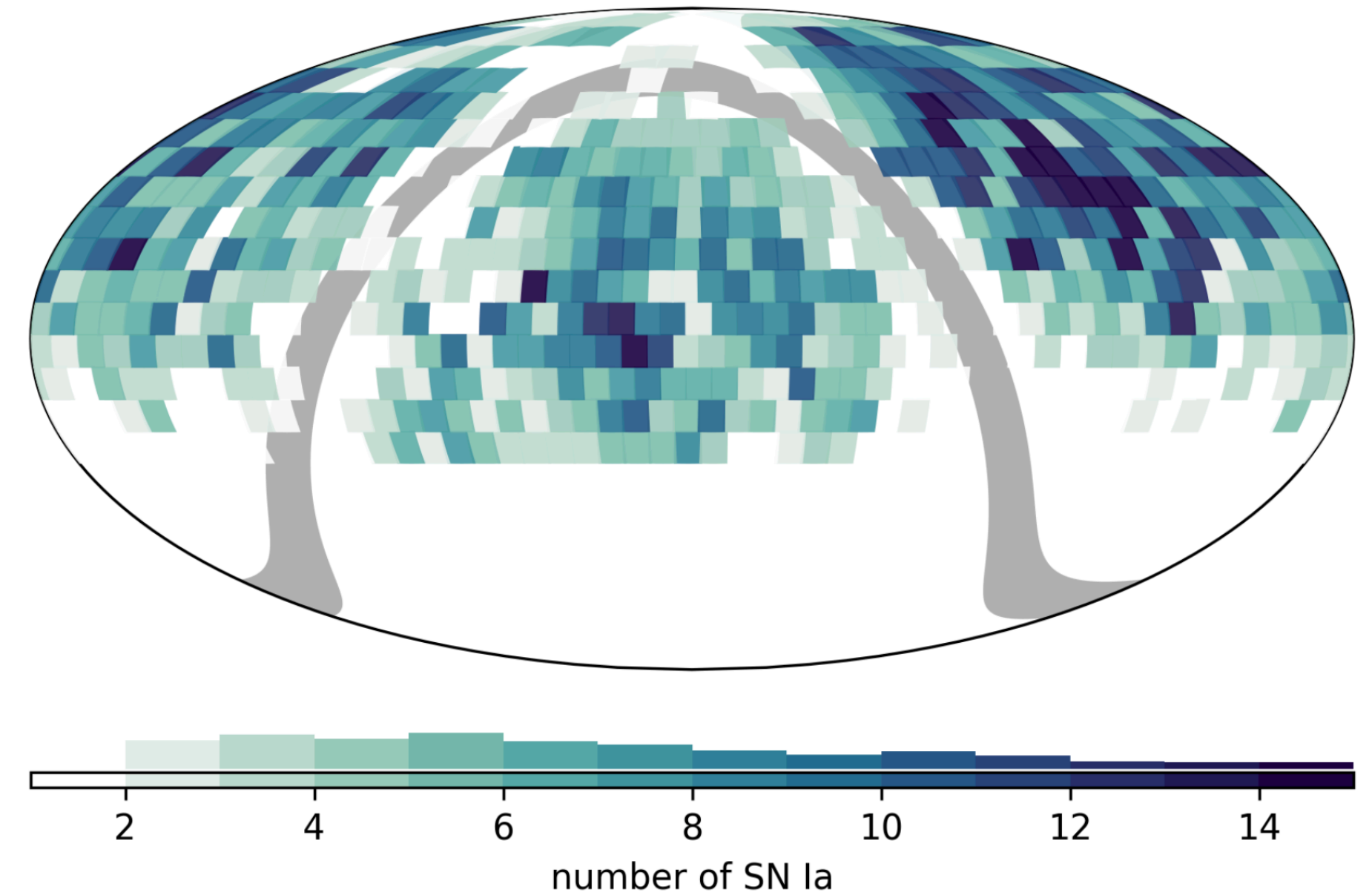
Data Release 2 :

- First and larger homogenous SNe Ia sample
- 3628 nearby events ($z < 0.3$)



ZTF DR2 : ZTF19abuonnx

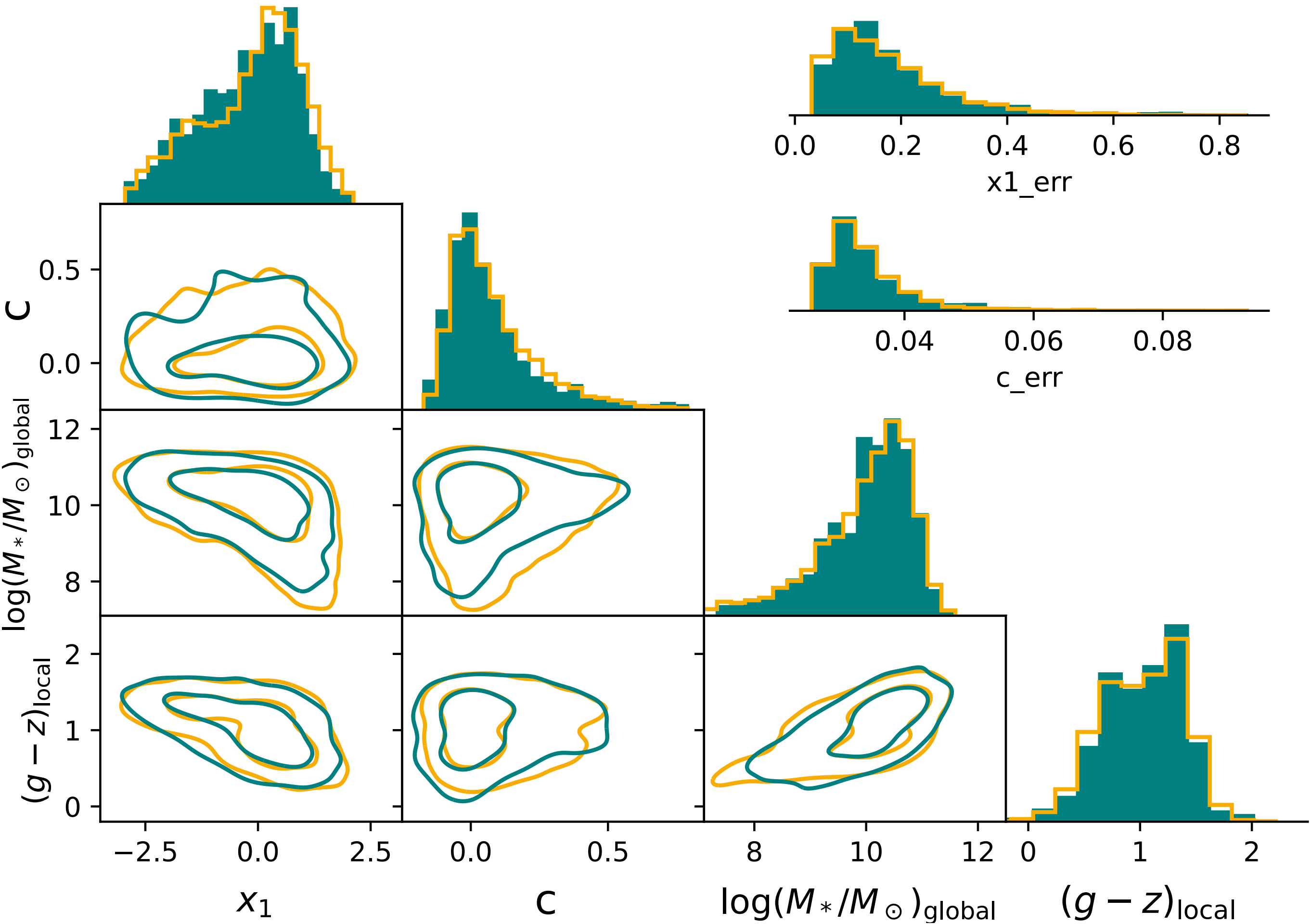
ZTF SN Ia DR2, data: ztfcosmo.in2p3.fr



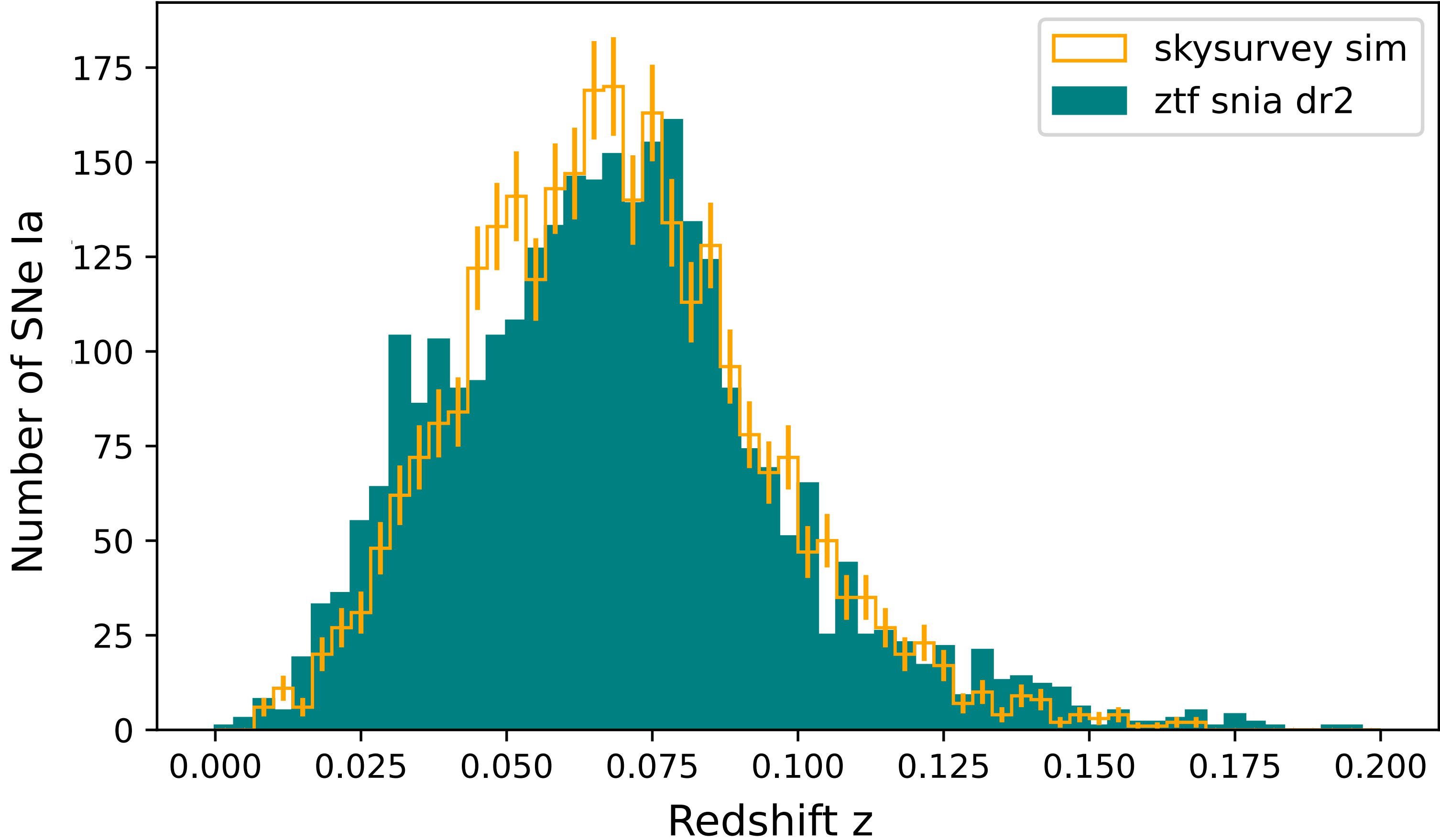
DR2 Overview : Rigault et al 2025

Skysurvey's state of the art

Ginolin et al. 2025a



Skysurvey's state of the art



- Assumptions :
- Homogeneous and isotropic Universe
 - Constant intrinsic SN Ia rate
 - Simple ZTF DR2 selection effect

Towards realistic simulations

Adding *systematics* to skysurvey

Selection effects

Astrophysical variabilities

Instrumental calibration

Galaxy related

20 %

Non Ia contamination

80 %

SNe Ia variabilities

80 %

SNe Ia properties evolution with redshift

30 %

Photometric calibration

20 %

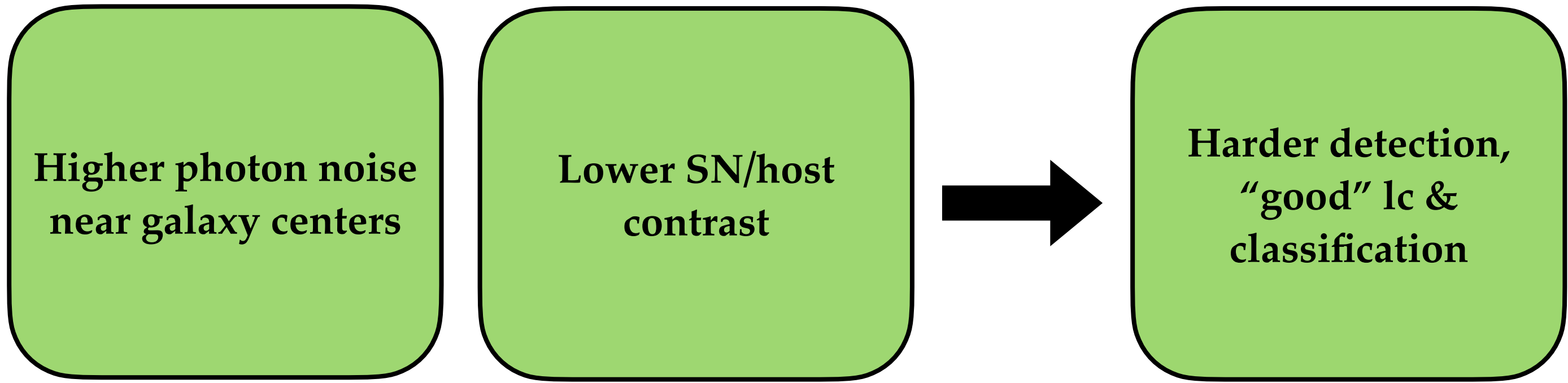
DESC Data Challenge: MS 2 and 3

DESC Data Challenge: MS 3

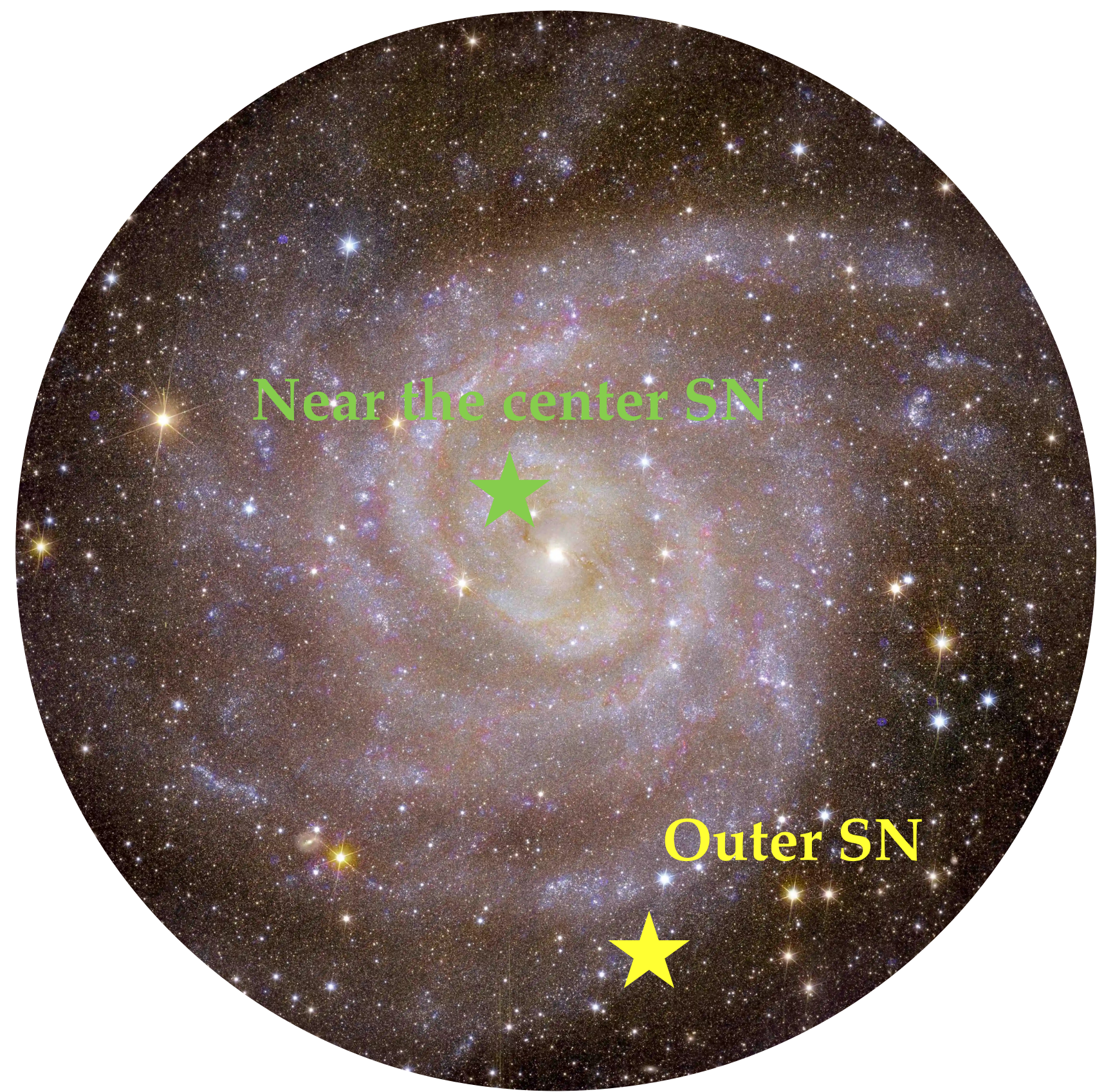
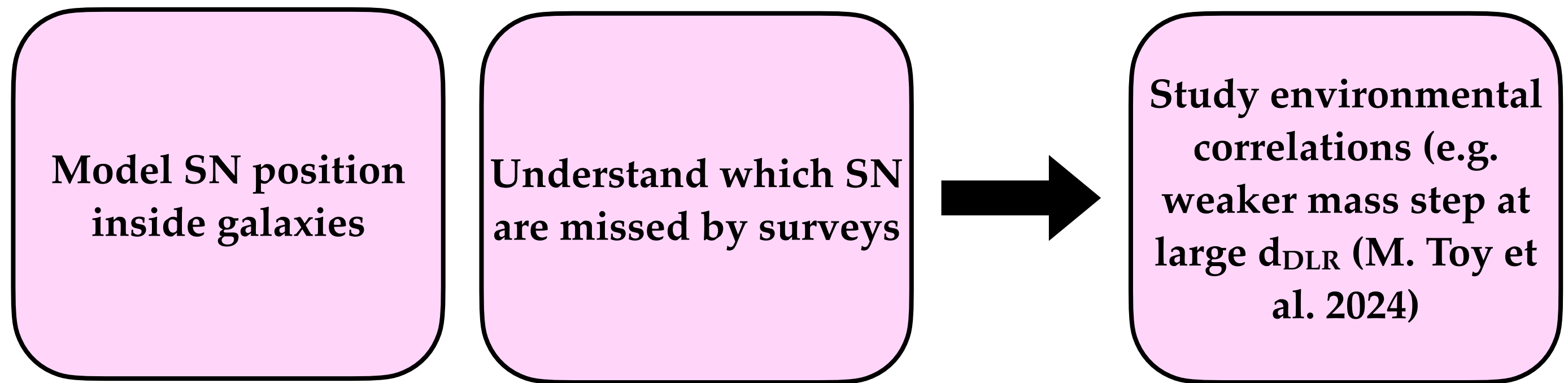
DESC Data Challenge: MS 3

Galaxy selection effects

Host-dependent selection effects:



Implementation in skysurvey:

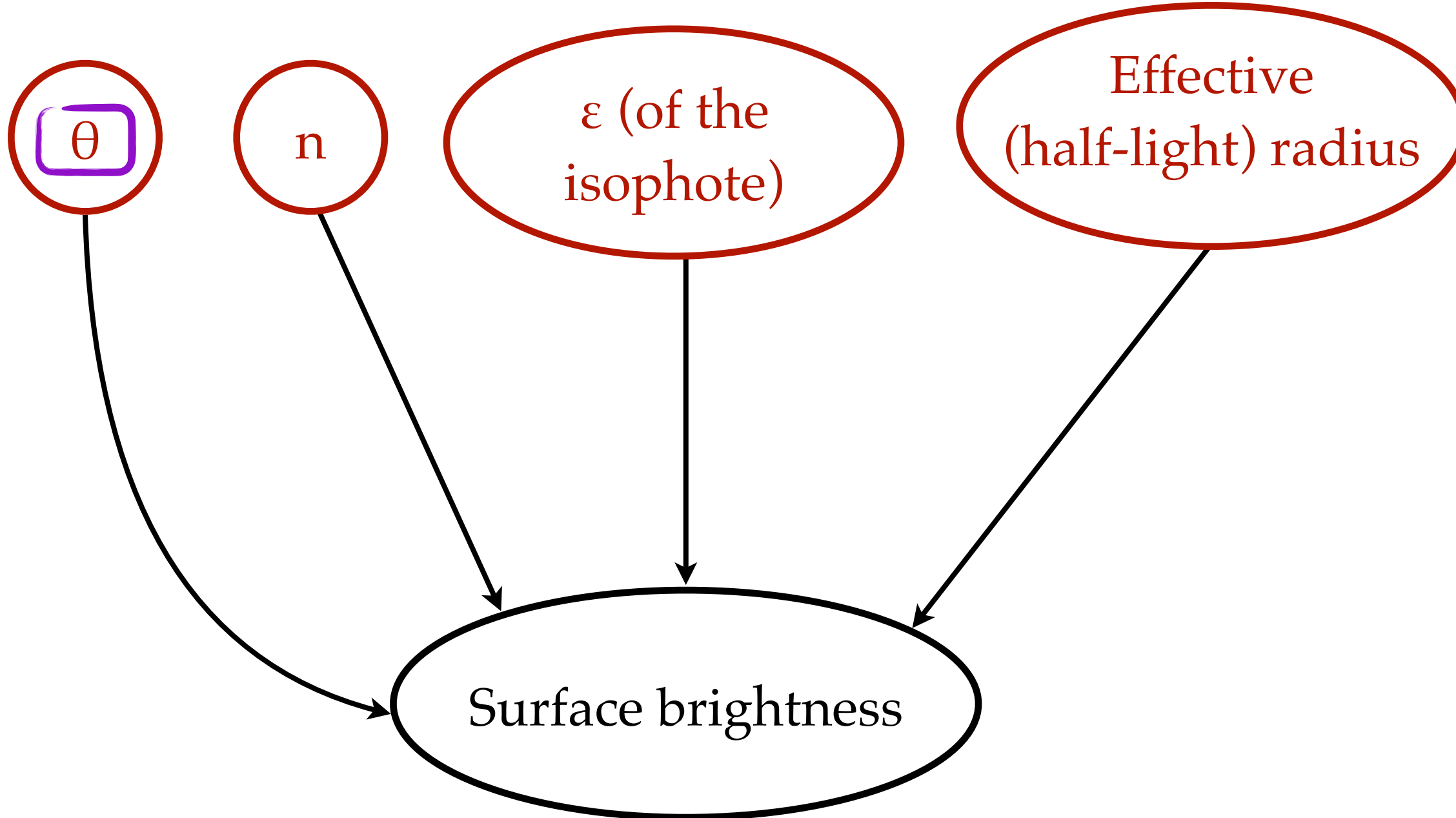


Galaxy modelling

First-approach :

simple Sérsic profile, independent parameters

DAG



```
sersic_profile = dict(  
    theta      = {"func": draw_position_angle, "kwargs": {}},  
    n          = {"func": draw_sersic_index, "kwargs": {}},  
    ellipticity = {"func": draw_ellipticity, "kwargs": {}},  
    r_eff      = {"func": draw_reff, "kwargs": {}},  
    surface_brightness = {"func": sersic_surface_brightness,  
                          "kwargs": {"theta": "@theta", "n": "@n",  
                                     "ellip": "@ellipticity", "r_eff": "@r_eff"}},  
)
```

```
mdag = modeldag.ModelDAG(sersic_profile)
```

see *modeldag* doc

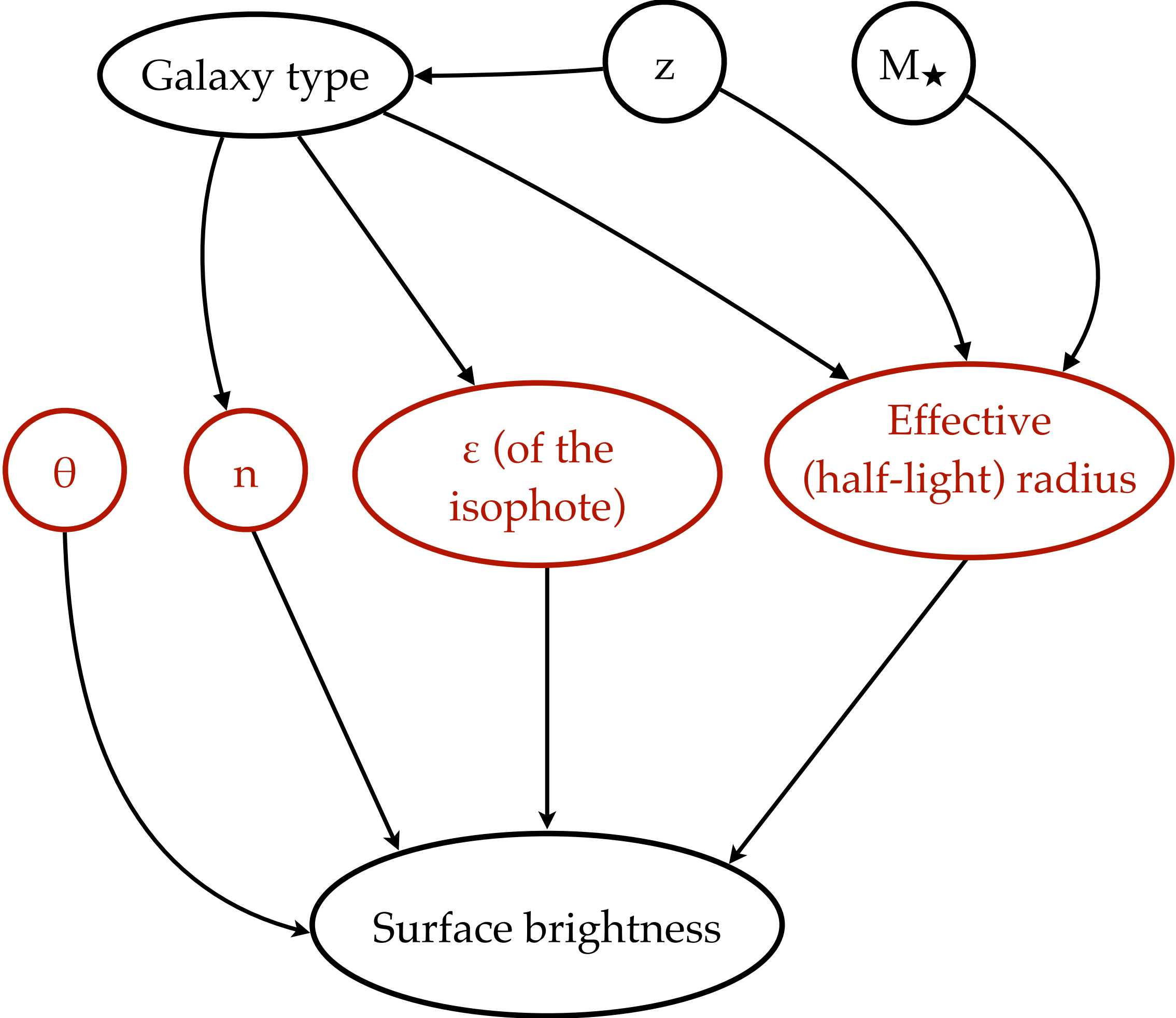
 Sérsic profil parameters

Galaxy modelling prospects



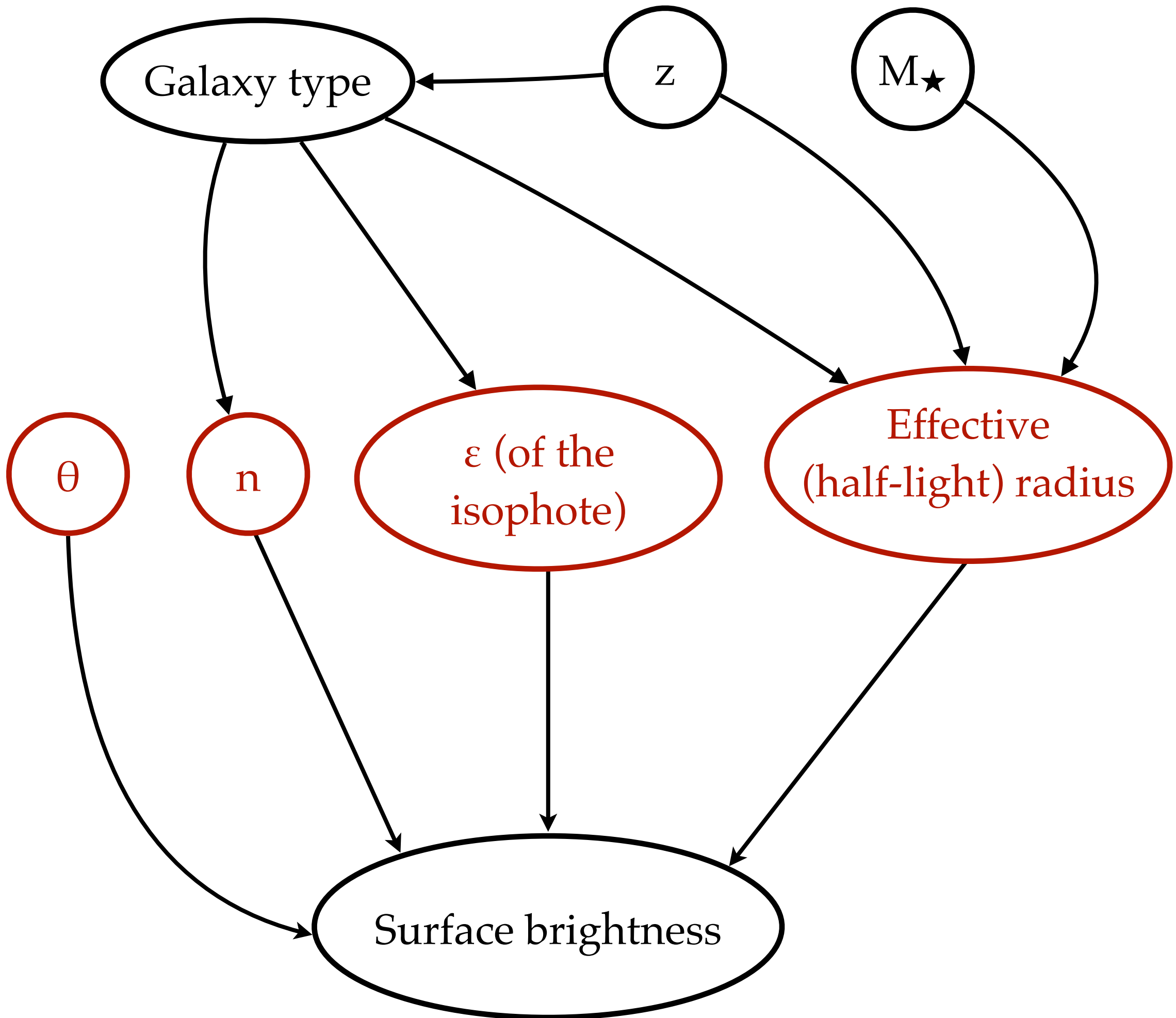
To more complex models :

- Galaxy properties (Galaxy type + M_{\star})
—> Metallicity, SFR, SFH
—> SN progenitor age, SN rate



 Sérsic profil parameters

Galaxy modelling prospects

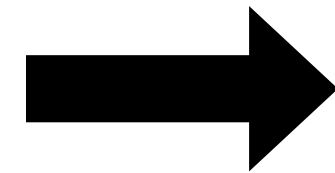
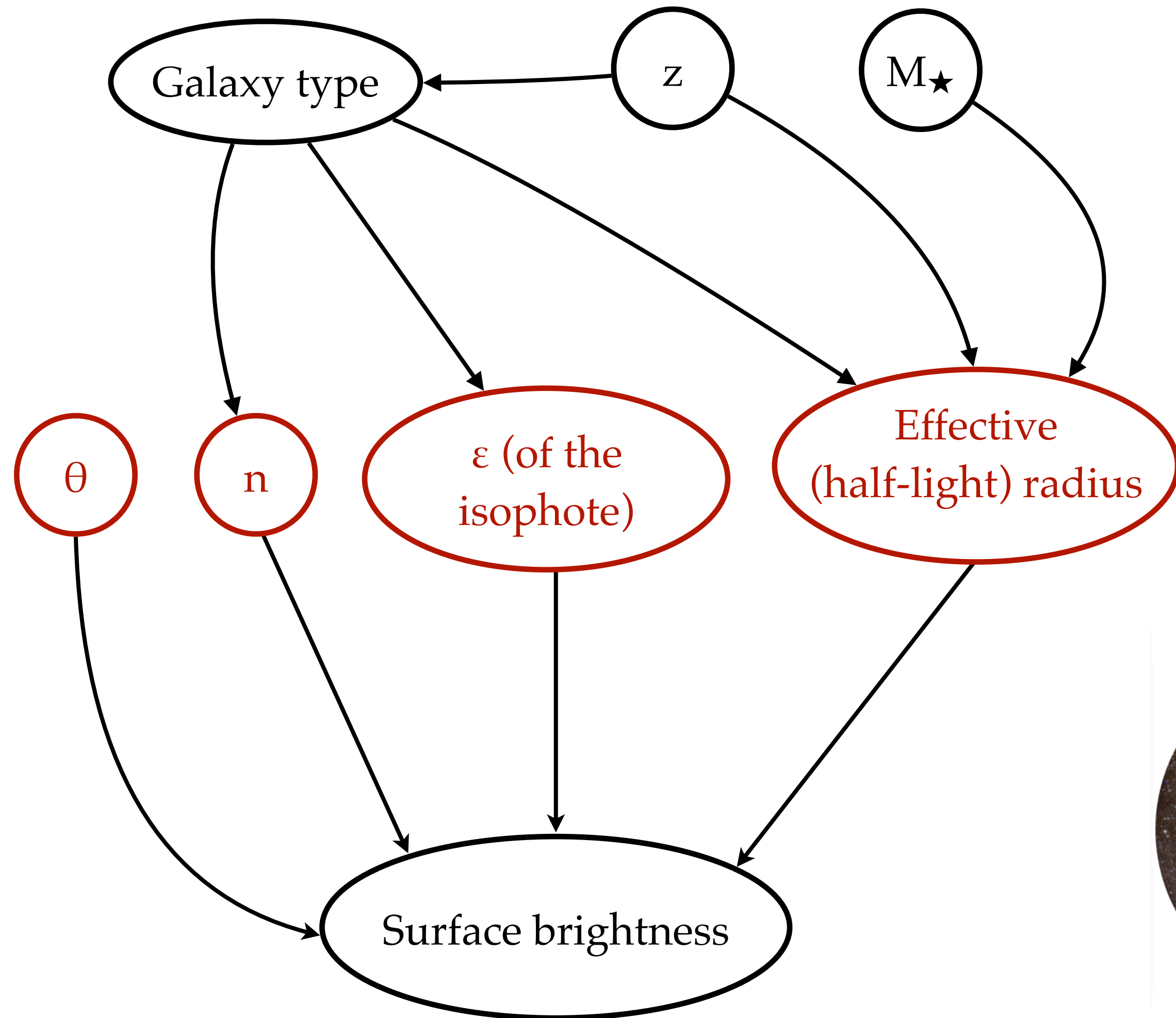


To more complex models :

- Galaxy properties (Galaxy type + M_{\star})
—> Metallicity, SFR, SFH
—> SN progenitor age, SN rate
- Double-Sérsic profile to describe region of a galaxy (spiral: $n_{\text{bulge}} \neq n_{\text{disc}}$) (Tang et al.25)

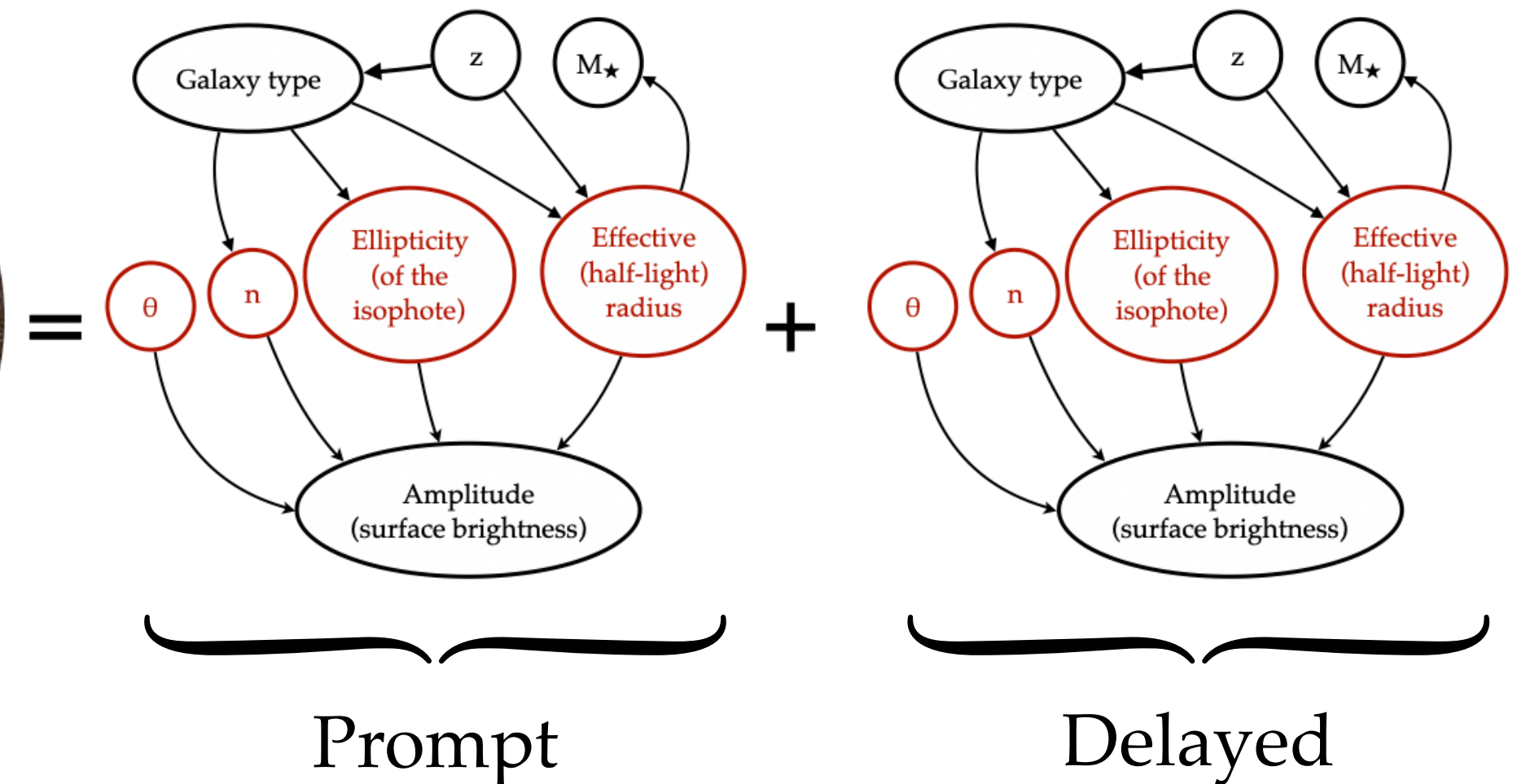
 Sérsic profil parameters

Galaxy modelling prospects



To more complex models :

- Galaxy properties (Galaxy type + M_{\star})
 —> Metallicity, SFR, SFH
 —> SN progenitor age, SN rate
- Double-Sérsic profile to describe region of a galaxy (spiral: $n_{\text{bulge}} \neq n_{\text{disc}}$) (Tang et al.25)
- Double-Sérsic profile for SN progenitors (prompt, delayed) (\neq SFR and regions of the galaxy)



Sérsic profil parameters

Conclusion

Adding *systematics* to skysurvey

Selection effects

Astrophysical variabilities

Instrumental calibration

Galaxy related

20 %

Non Ia contamination

80 %

SNe Ia variabilities

80 %

SNe Ia properties evolution with redshift

30 %

Photometric calibration

20 %

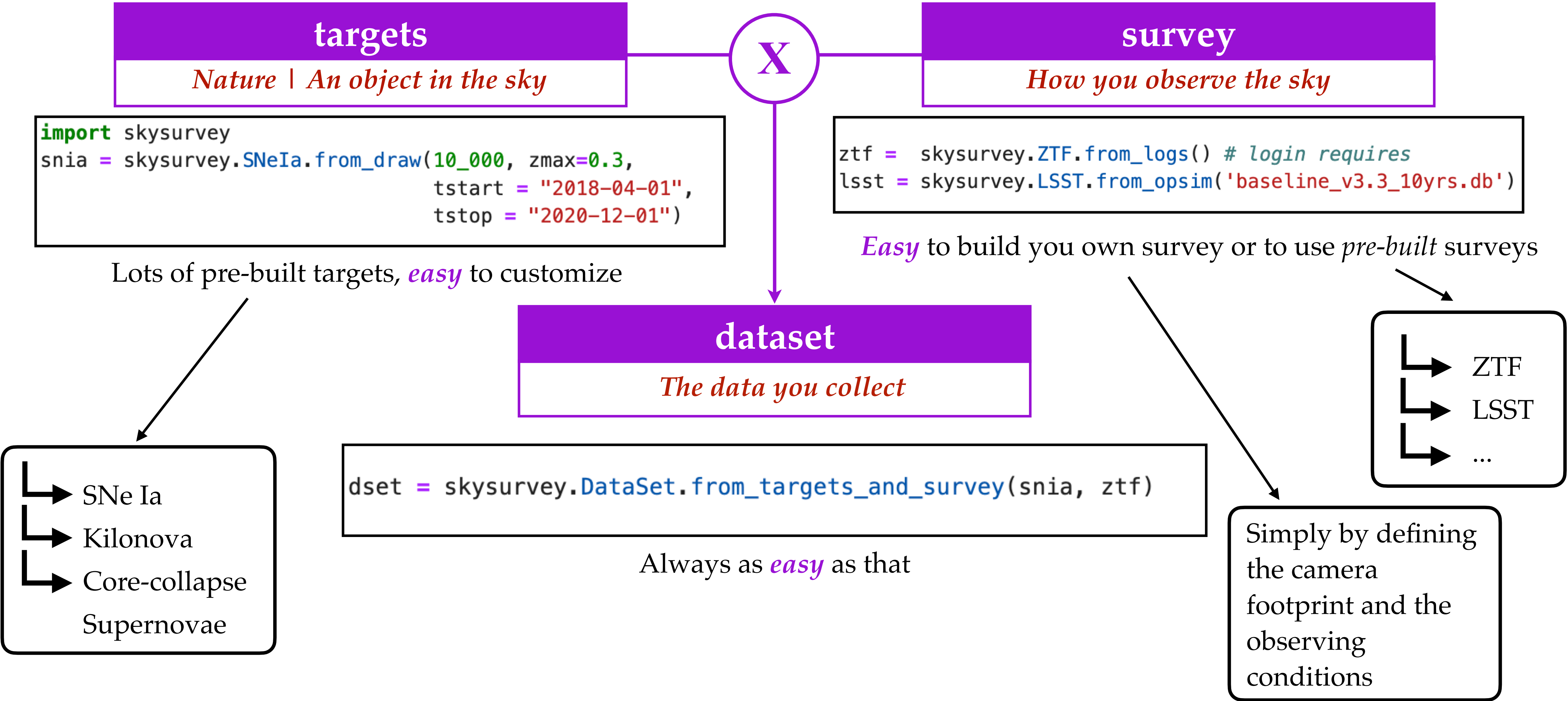
DESC Data Challenge: MS 2 and 3

DESC Data Challenge: MS 3

DESC Data Challenge: MS 3

Annexes

skysurvey : concept



Pre-Built Transients

Transient templates and sources are built on top of **SNCosmo** (Barbary et al. 2025)

Any **SNcosmo built-in source** is available:

Type Ia Supernovae

```
snia = skysurvey.SNeIa()
```

Transient Time Serie

```
sn_07pk = skysurvey.TSTransient("v19-2007pk-corr")
```

Kilonova

```
kilonova = skysurvey.Kilonova()
```

Supernova Types

```
snii = skysurvey.SNeII.from_draw(1000)
```

“Collection
Transient Time Series”

SNII | SNIib | SNIIn | SNIb | SNIc | SNIcBL
23 12 6 12 7 6

(Vincenzi et al. 2019)

Customization is *easy*
rate, redshift range, absolute
magnitude distribution,...

see skysurvey doc for how to change the rate of a target

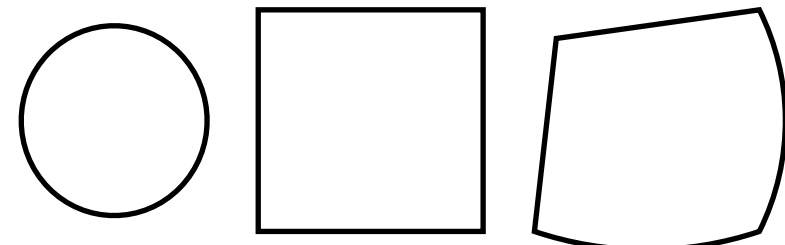
see tutorial "how to create a new transient class" on the skysurvey doc

Build your own Survey

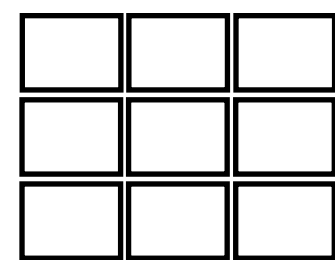
```
# Load the survey
survey = skysurvey.Survey.from_pointings(data, footprint=footprint)
```

footprint | camera footprint in the sky

shapely



Polygon



MultiPolygon

```
skysurvey.get_footprint("des")
```

```
import numpy as np
from shapely import geometry
from skysurvey.tools import utils

# footprint | 2-deg radius circle
footprint = geometry.Point(0,0).buffer(2)
```

data | what was observed when, under which conditions

pandas



ra, dec — Camera pointings

mjd — time in Modified Julian Date

band — name of the filter used

skynoise — level of the sky noise (unit of zp)

zp — depth of the image

gain — camera gain (e-/adu), used for noise

```
data["skynoise"] = np.random.normal(size=size, loc=200, scale=20)
```

Easy to build complex transient

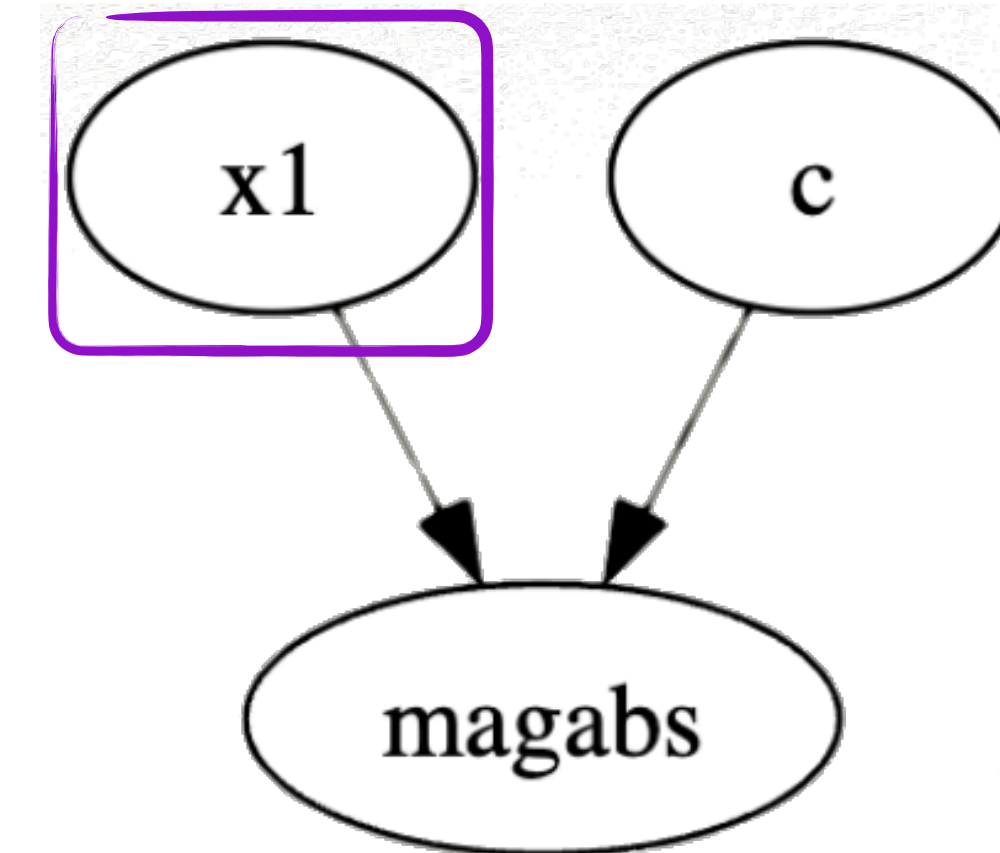
DAG

```
# ----- #
# Your functions #
# ----- #
def tripp98(x1, c, alpha=-0.14, beta=3.3, sigmaint=0.1, mabs=-19.3):
    """This is the Tripp 98 relation."""
    magabs = x1 * alpha + c * beta + mabs # natural SN Ia magnitude (non-standardized)
    magabs_scattered = np.random.normal(loc=magabs, scale=sigmaint) # add intrinsic scatter
    return magabs_scattered

# ----- #
# Create the dict #
# ----- #
modeldict = {
    "x1": {"func": np.random.normal,
          "kwargs": {"loc": 0, "scale": 1}},
    "c": {"func": stats.lognorm.rvs,
          "kwargs": {"s": 0.7, "loc": -0.2, "scale": 0.2}},
    "magabs": {"func": tripp98,
               "kwargs": {"x1": "@x1", "c": "@c"}}
}

# ----- #
# Simulate #
# ----- #
mdag = modeldag.ModelDAG(modeldict)
data_sim = mdag.draw(1_000)
```

```
mdag.visualize()
```



see modeldag doc

Easy to build complex transient

```
model_extension = {"prompt": {"func": ...,  
                             "kwargs": ...},  
                  "mass": {"func": ...,  
                           "kwargs": ...},  
                  ...  
                  }  
  
import skysurvey  
snia = skysurvey.SNeIa.from_draw(1_000,  
                                model=model_extension)
```

see modeldag doc

Current SNe Ia "Age model"

