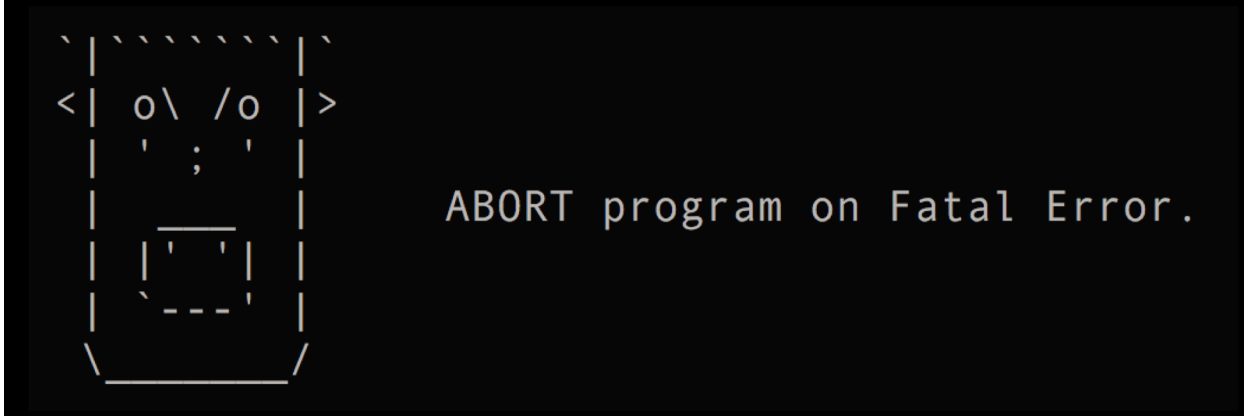


SNANA Tutorial

R. Kessler

Duke Seminar

June 18, 2020

From the SNANA Legal Team:

- **SIDE EFFECTS** include, but are not limited to: confusion, frustration, watery eyes, head-aches, weight loss, weight gain, systematics-limited results, incorrect results, denial of tenure.
- Do not drive or operate heavy machinery while using SNANA.

GOOGLE Search: No, not this SNANA

SNANA

Welcome to the SuperNova ANAlysis software homepage

Install Guide | SNANA Manual | Overview Paper | Legal Notice

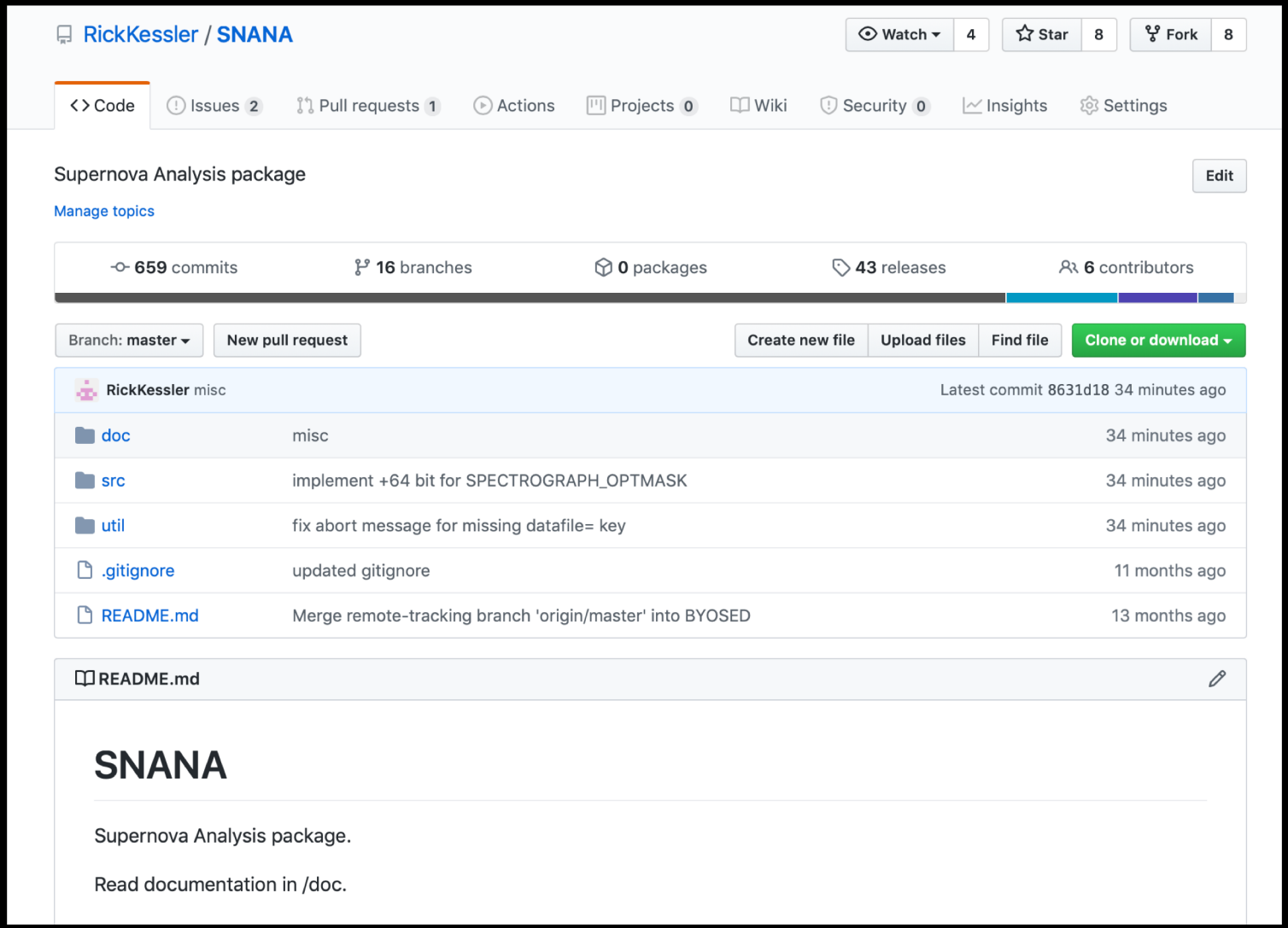
SNANA Download | Mailing List | Contact Form

SNANA contains a light curve fitter and simulation that can be applied to any supernova (SN) model and to any data set. This website provides installation instructions, a user manual, and a software package download area.

And not this obsolete link

It's here on Github:

https://github.com/RickKessler/SNANA



Outline

- Intro
- Architecture
- Main programs and features
- Systematics & Multi-Core Jobs
- Pippin Analysis Pipeline (Super Glue)
- Output
- Documentation

Why SNANA

- ***SNIa-Cosmology Analysis (SDSS, JLA, PS1-Pantheon, DES)***
- Forecast future surveys (DES,LSST,WFIRST …)
- Simulate wide range of transients (e.g., PLAsTiCC classification challenge)
- Quantify sensitivity for rare searches (e.g., KN, ICECUBE-v source)
- Efficiency correction for rate measurements
- Validate diff-imaging pipeline (using fakes)

Few Principles for SNANA

- ***Science needs drive development***
- Stay connected to real data, even if designing a future survey
- Always look for opportunities to try new things, even if it seems like a distraction.
- Minimize memory, process time, file size.

Summary of Ready-to-Run Programs in $SNANA_DIR

- Simulation package (catalog, not pixels)
- Light Curve Fitting & Template Fitting
- Bias Corrected Hubble diagram using BBC
- Create stat+syst covariance matrix
- Cosmology Fitting (old, simple, fast: better codes elsewhere)
- Utilities for systematics & multi-core processing
- *NO Image-Processing Tools*

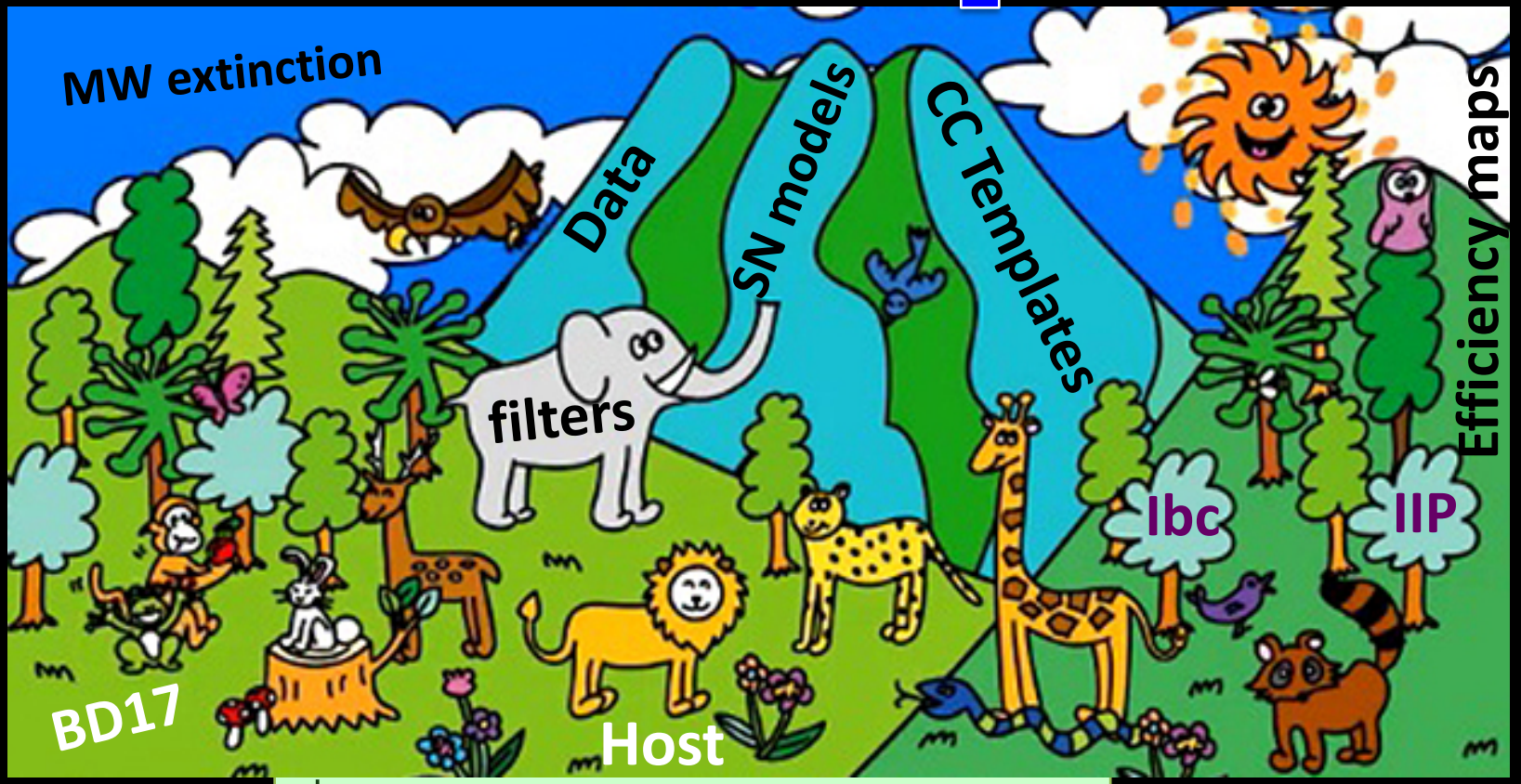
Code Languages

- sim, BBC, wFit: **C**
- LC fit: **C + Fortran** (unintended consequence)
- Batch-submit scripts: **Perl** (naive mistake)
- createCov + plot-util + misc: **Python**

Architecture: Environment

- Simulation package (catalog, not pixels)
- Light Curve Fitting & Template Fitting
- Hubble Diagram
- Utilities for systematics & multi-core processing

$SNANA_DIR



$SNDATA_ROOT

SNANA Architecture: File Sharing

SNANA was implicitly designed to run on a cluster with many users sharing files.

SNANA Architecture: File Sharing

SNANA was implicitly designed to run on a cluster with many users sharing files.

$SNDATA_ROOT contains

- Data
- Simulated output
- Filter transmissions
- Primary SEDs
- Ia & CC spectral templates
- MW extinction map
- SN model parameters
- SIMLIB files
- HOSTLIB files
- Efficiency maps
- Etc . . .

Can run tests with file(s) in your private directory, but goal is to share files with ***community*** via $SNDATA_ROOT

SNANA Architecture: Sharing Proprietary Files

During analysis it is useful to share proprietary SNANA files,

$DES_ROOT
$LSST_ROOT
$WFIRST_ROOT

INTERNAL directories are *NOT* public.

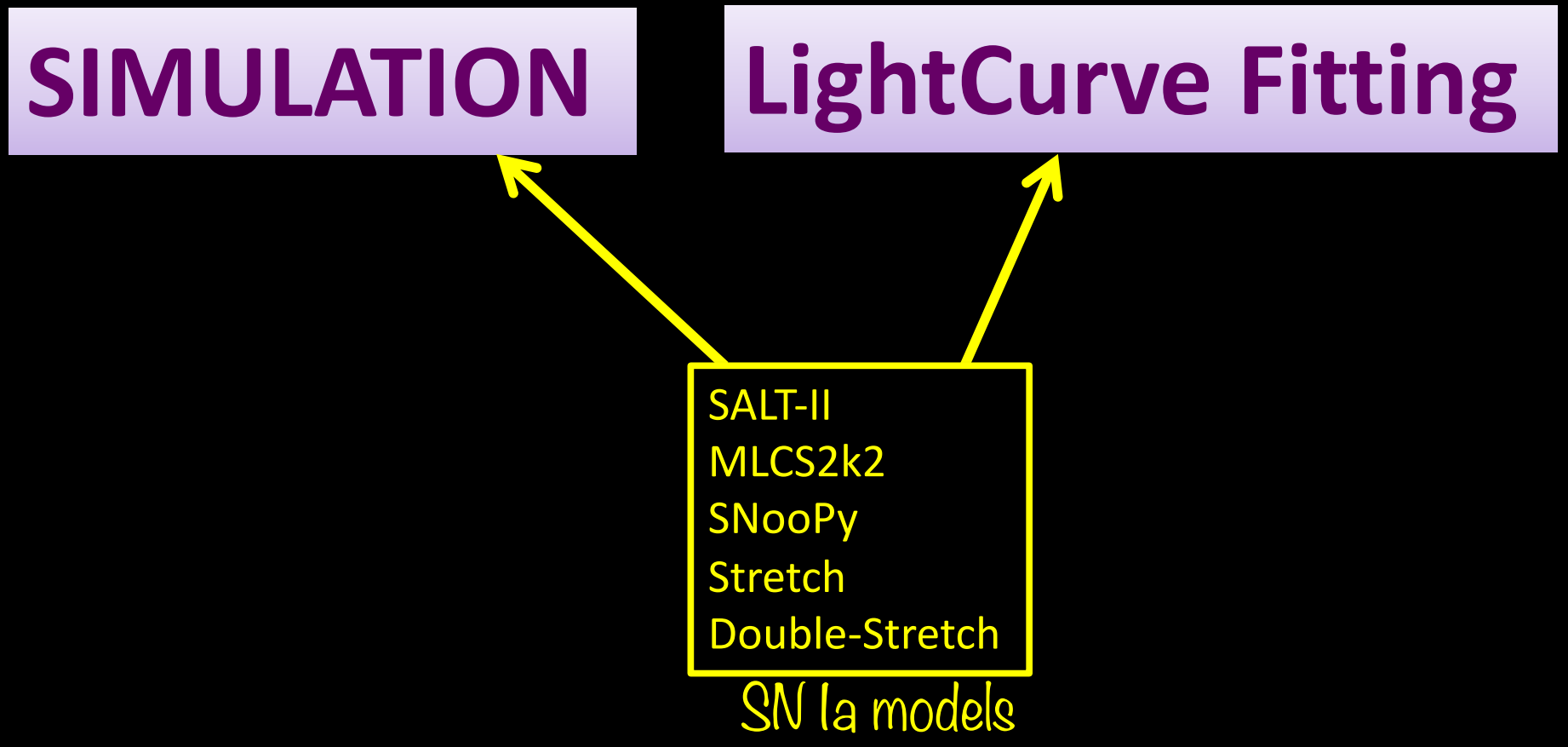
ENV can be used as part of any input fileName, e.g.,
SIMLIB_FILE: $DES_ROOT/simlibs/DES_DIFFIMG.SIMLIB
(ENVs simplify transfer to another compute cluster)

Surveys Investigated on U.Chicago's "Midway" Cluster

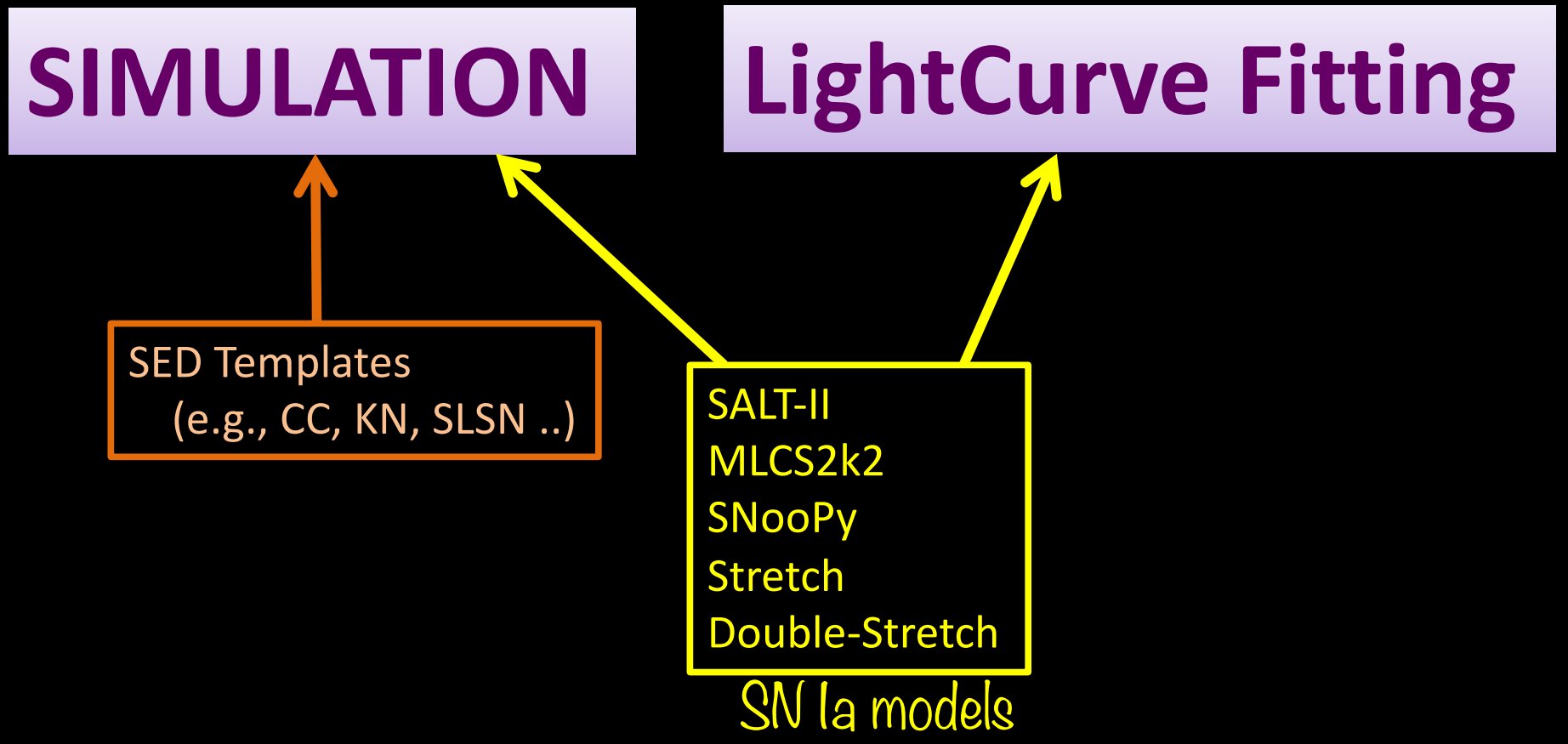
```
DES         JLA    PS1MD   SMT       WFIRST   ZTF
FOUNDATION  LSST   SDSS    SNTRAIN   YSE
```

- No barriers between project areas
- Easy to switch between projects (no new login)
- Easy to collaborate with others
- Relies honor system to respect proprietary files

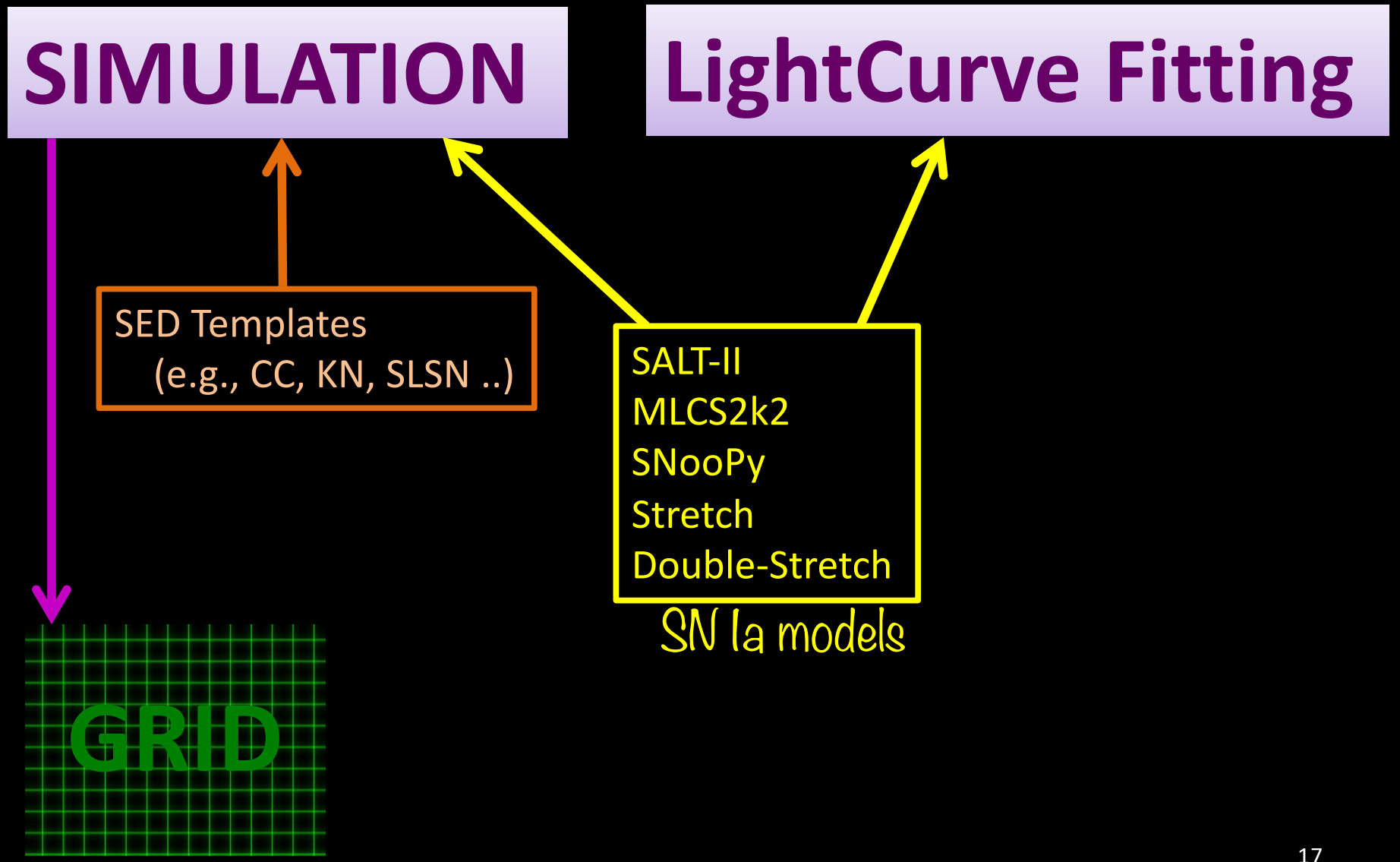
SNANA Architecture



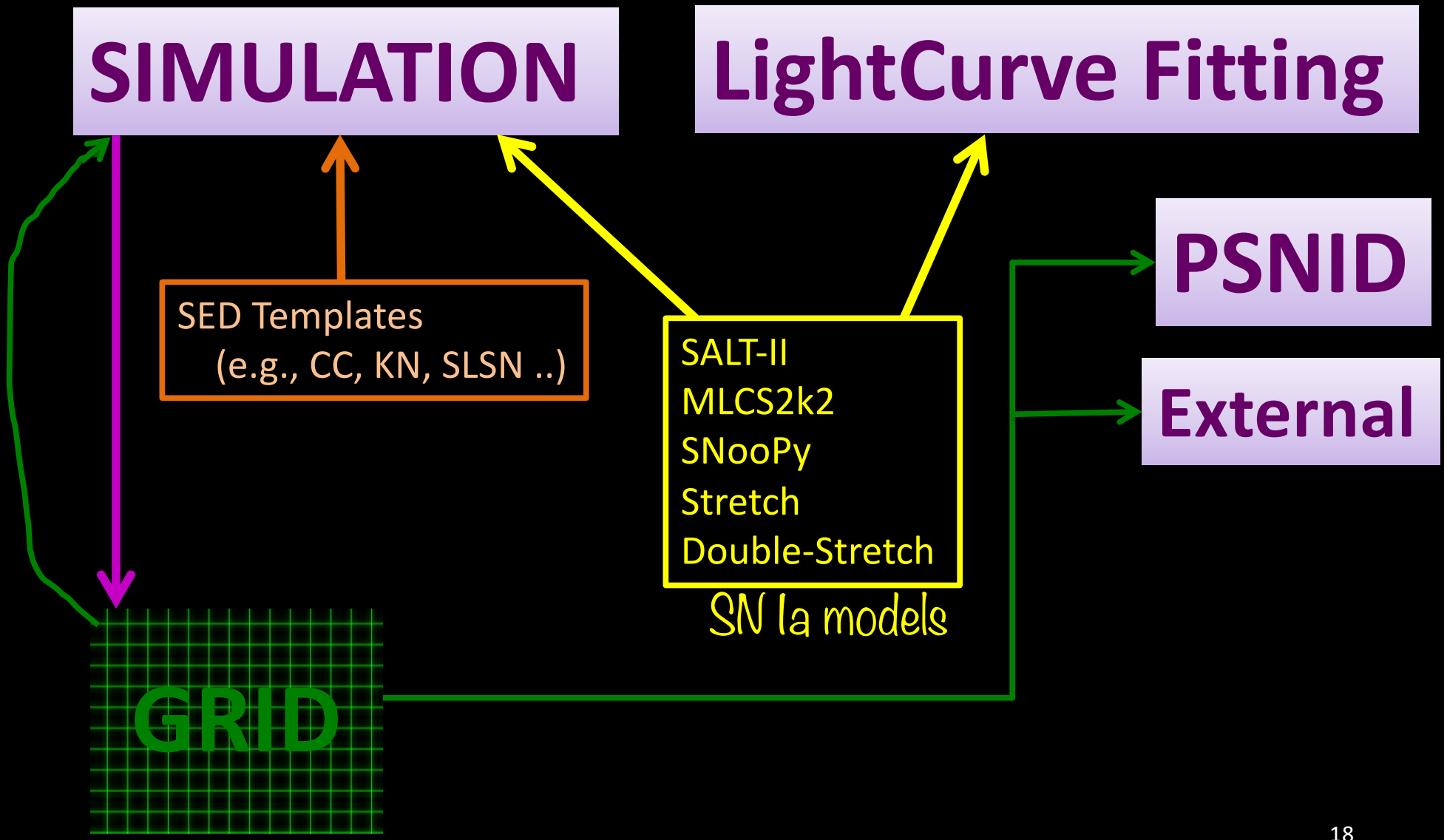
SNANA Architecture



SNANA Architecture



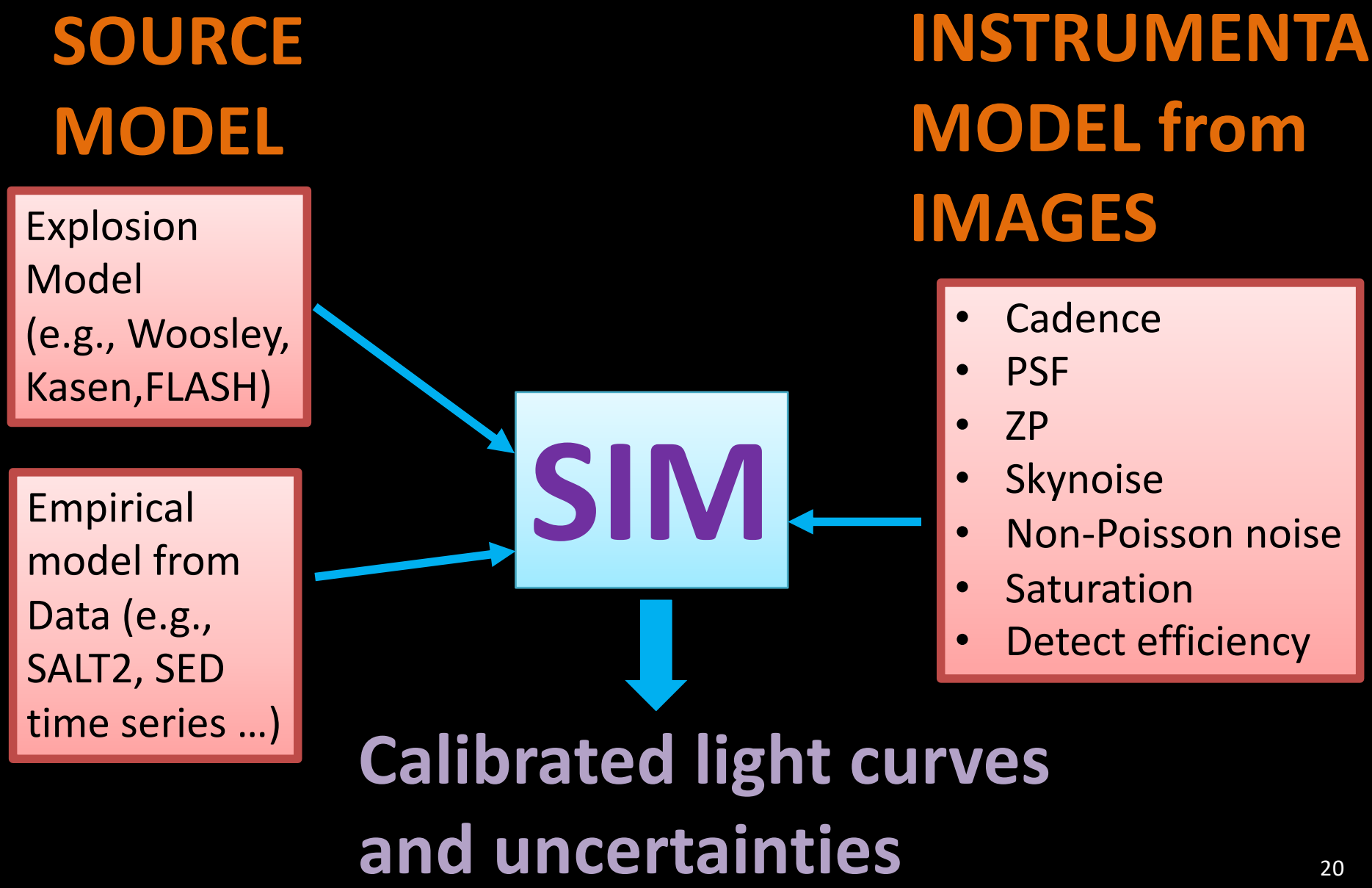
SNANA Architecture



Why Use GRID ?
(versus model params & redshift)

- Allows using templates constructed from non-SNANA programs
- Any SN model → standard model format for template-fitting programs such as PSNID
- Can be faster (e.g., huge speed-up for SNooPy)

SNANA Simulation



Source Model

Generate source SED (each epoch)

↓

Cosmological dimming, lensing, pec. velocity, redshift SED

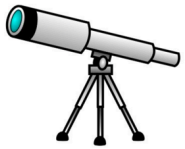
↓

Galactic extinction

↓

Integrate redshifted SED for each filter

Noise Model



Use cadence info (PSF, SKY, Zeropoint) to convert true mag to true flux (ADU) and true uncertainty

↓

Apply Poisson noise to get measured flux.

Trigger Model

Check for detection

↓

Apply candidate logic: e.g., 2 detections

↓

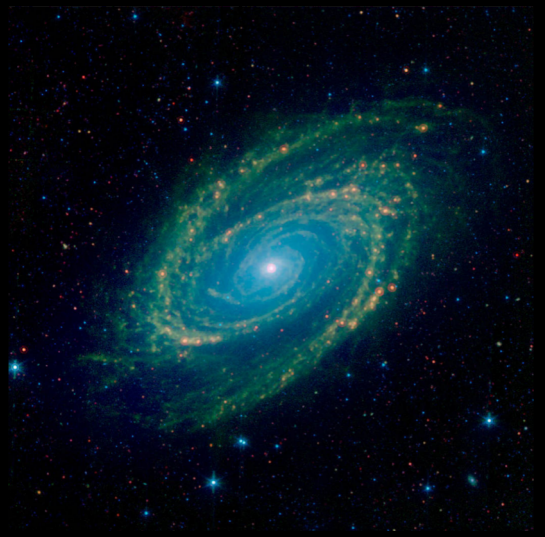
Apply spectroscopic selection function

↓

Write selected events to data files.

Almost 20 different ``maps'' are input to the simulation

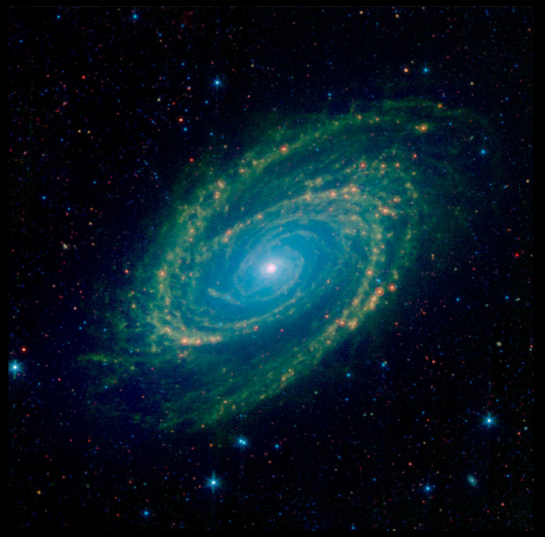
Host Galaxy Library (HOSTLIB)



- mis-matched host based on min DLR
- SN-host correlations (e.g., brightness vs. mass)
- SN population (color, stretch, dust) dependence on HOST properties (e.g, mass, SFR …)
- Host photo-z (e.g., from Graham 2018)

  use as prior in fit for SN photo-z
- Local surface brightness → Poisson & non-Poisson (anomalous) noise.
- Efficiency map of spec-zHOST vs. HOST properties

Host Galaxy Library (SN-Host correlations)



- Given SN properties (color, stretch), choose distribution of host properties

  or

- Given host properties, choose distributions of SN properties. (color, stretch, RV, AV)

Host Galaxy Library

(Example HOSTLIB using MICECAT)

```
VARNAMES: GALID RA_GAL DEC_GAL ZTRUE
  g_obs r_obs i_obs z_obs
  ABS_MAG_g ABS_MAG_r ABS_MAG_i ABS_MAG_z
  a0_Sersic b0_Sersic n0_Sersic a_rot
  LOGMASS LOG_SFR

GAL: 28581888 6.32295 25.82068 0.30917
   22.47973 21.19873 20.77373 20.56673
  -18.71227 -19.31427 -19.55727 -19.76027
   0.91923 0.36473 2.0 -162.357 9.89 -10.2

etc ...
```

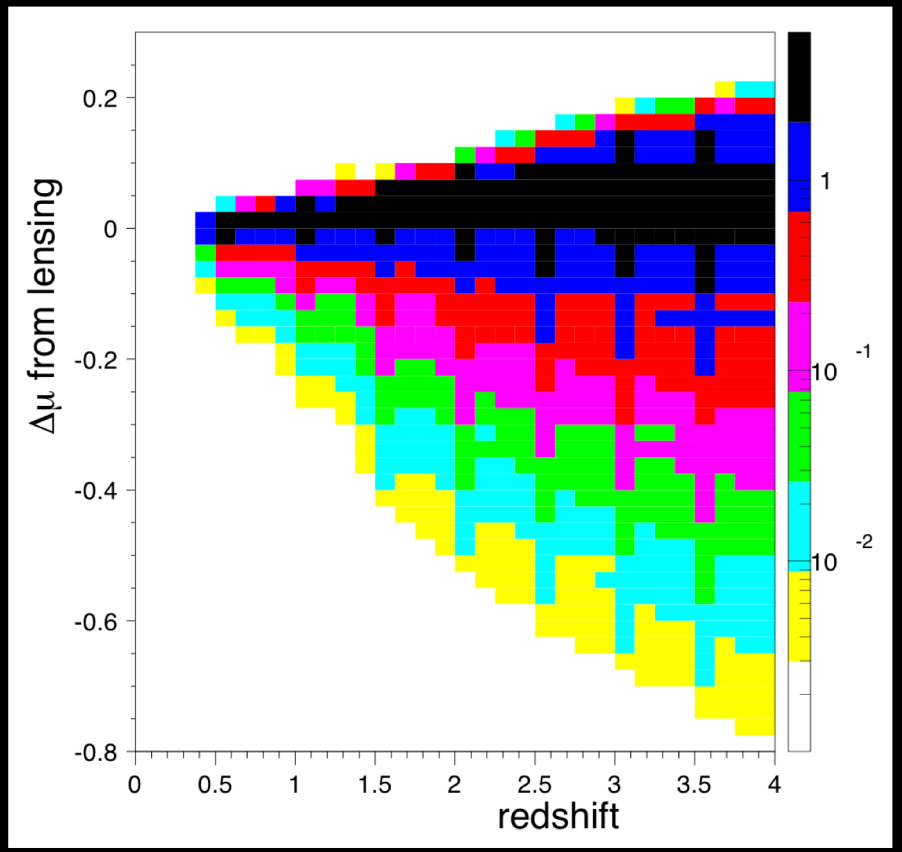
Host Galaxy Library (HOSTLIB)

WARNING

SNANA code implementation is the easy part … main burden is on the user to construct suitable HOSTLIB

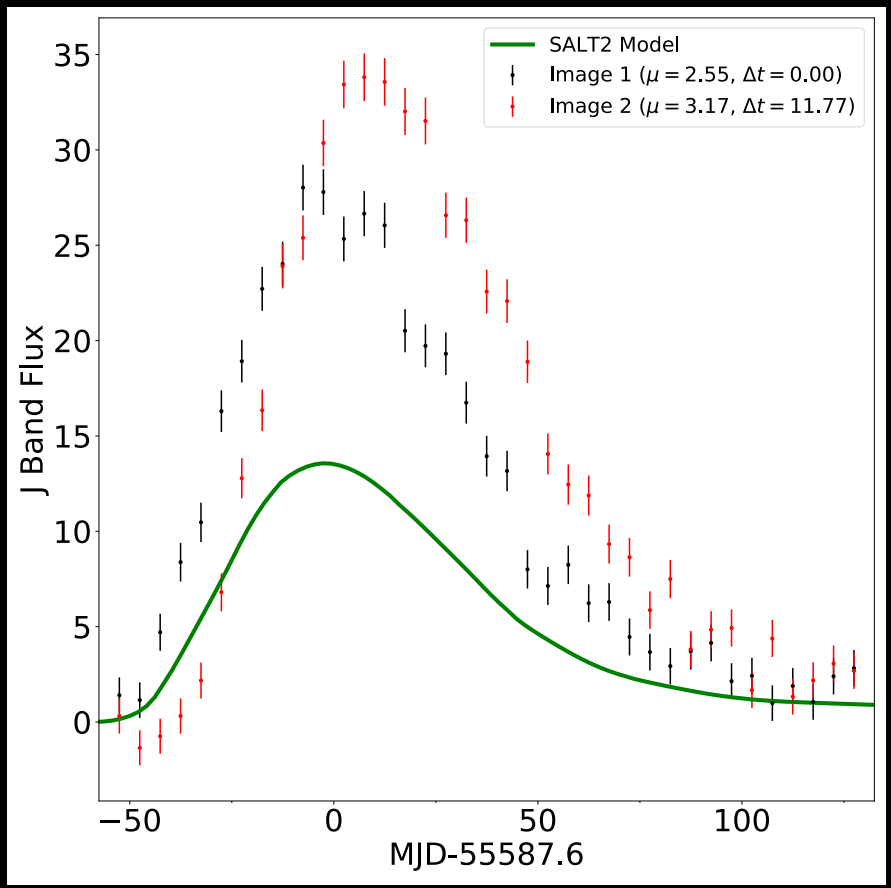
Simulation of Weak & Strong Lensing

Weak lens magnification map from
J.Barreiro, T. Davis (MICECATv1)



Nancy Grace Roman Space Telescope

J. Pierel et al, in prep



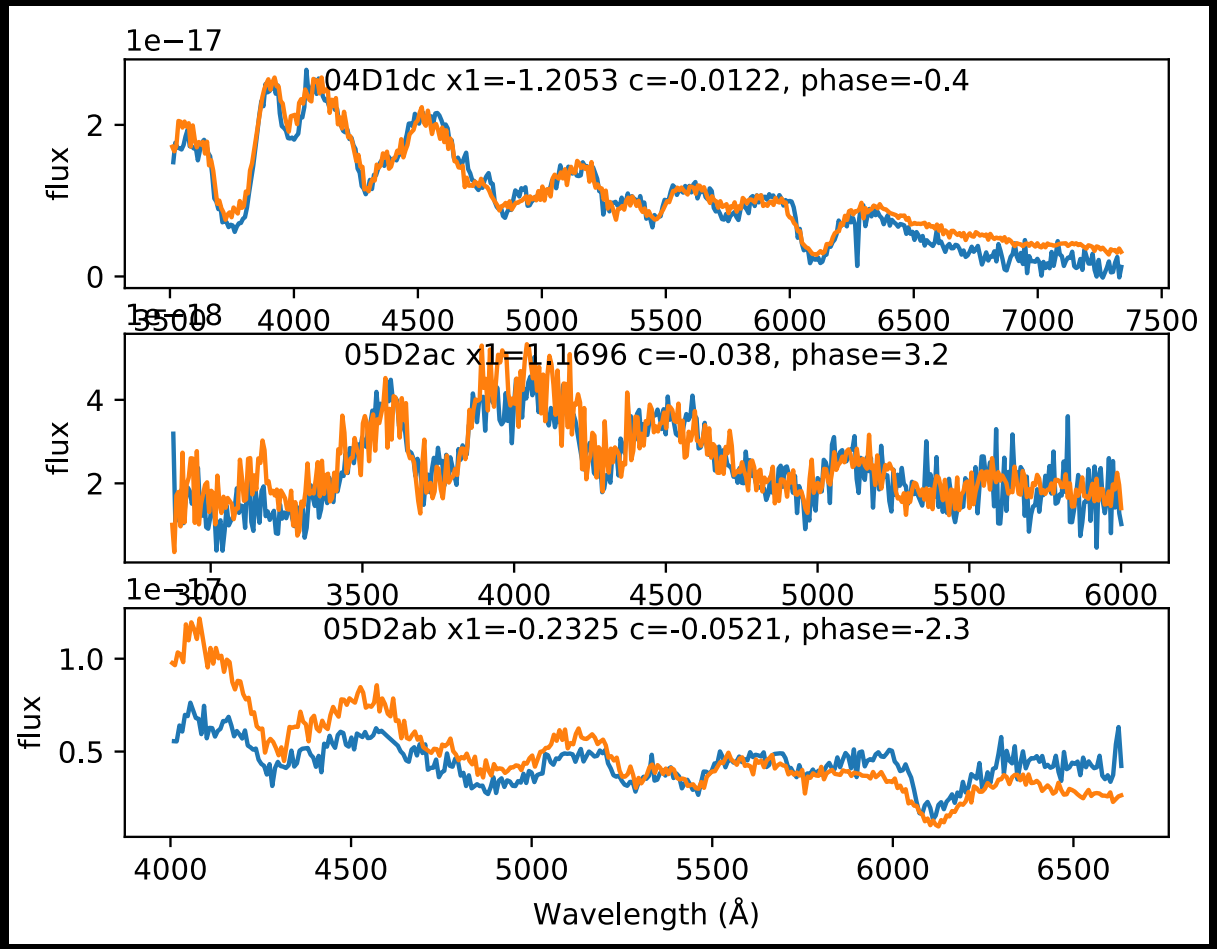
Simulating Spectra: SN & Host

- Test SALT2 training with simulated photometry & spectra
  - Impact on spectral calibration
  - Impact of host contamination
  - Impact of spectral wave & phase coverage
- NGRST (WFIRST): forecast host redshift capabilities

Simulating Spectra: SN

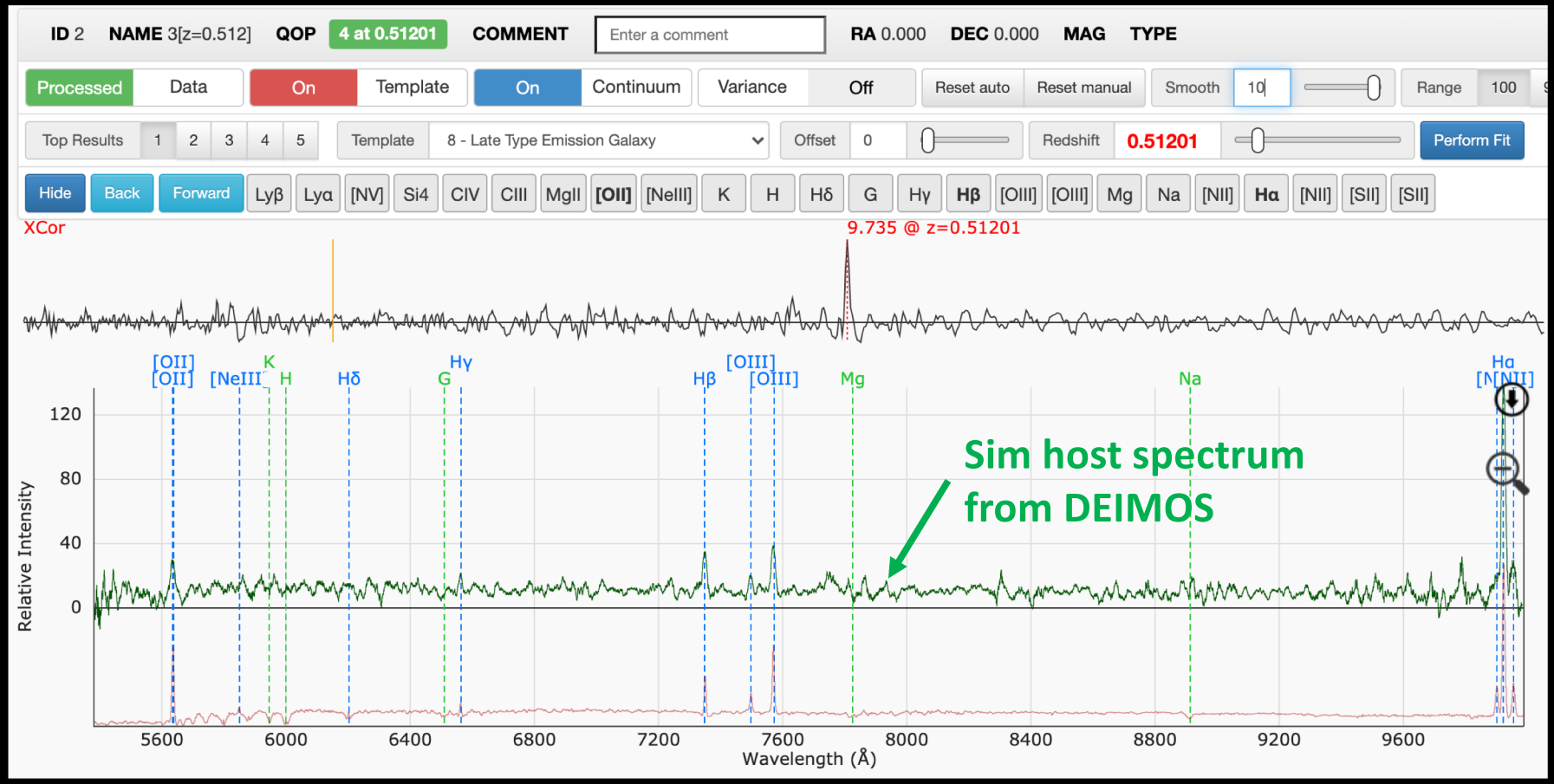
--- Real SNIa, SNLS

--- Sim SNIa, SNANA



Plot from D.Jones

Simulating Spectra: Host
(and reformat option for Marz redshift utility)



- Spectrograph Sim from S.Rodney
- http://samreay.github.io/Marz (by S.Hinton)

SIM CPU Proc-Time

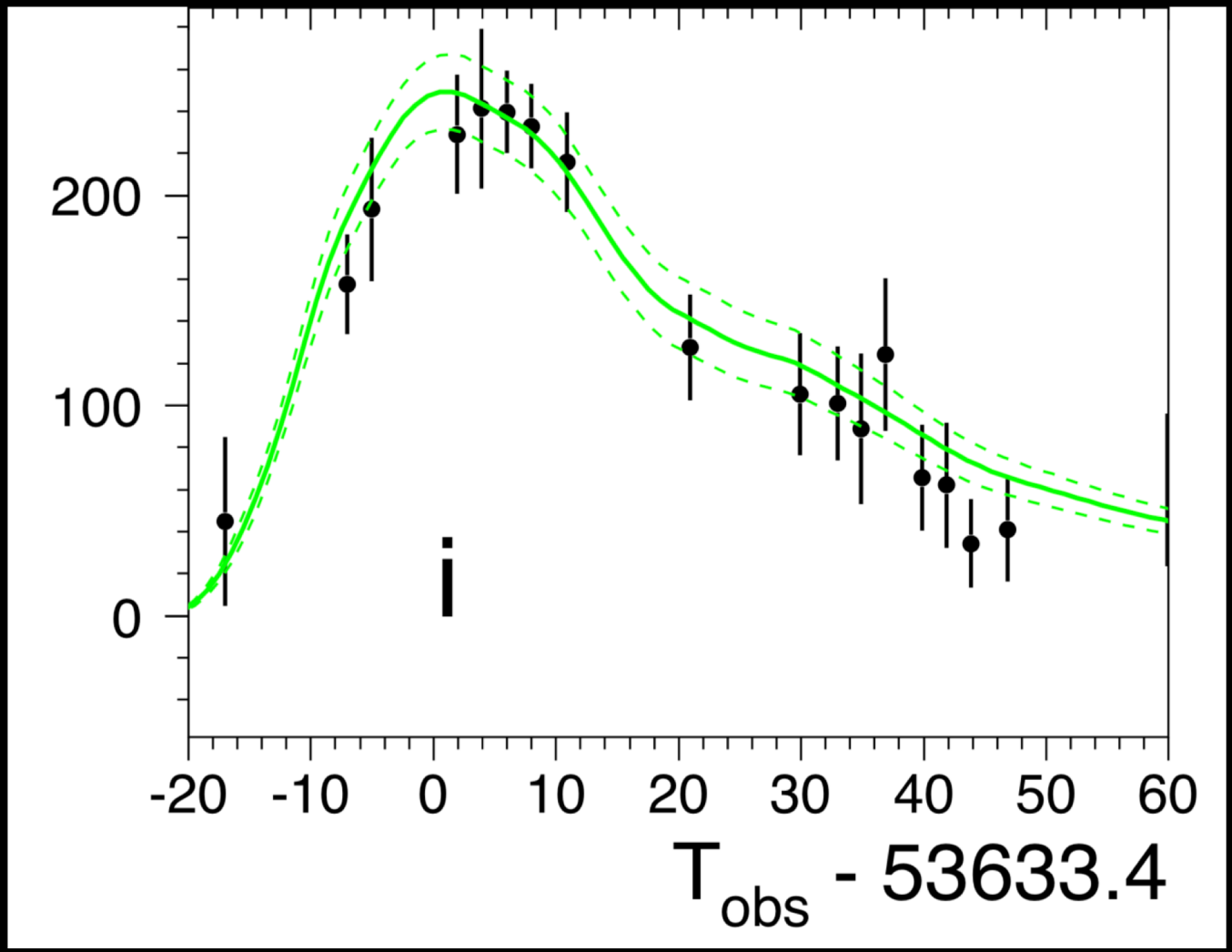
U Chicago Research Computing Center:
Sep 2018 for PLAsTiCC

- 117 million light curves generated in 8 hr on 40 cores → 100/sec
- 3.5 million written to data file → 3/sec

No API for Simulation

- However, without re-compiling can add
  + new survey, filters, Cadence, HOSTLIB, calibration info
  + new SED templates
- Supernova Standardization Team (SST) has added ``Build Your Own SED'' model (BYOSED) in python, with C calling python.
- SST Goal: arbitrary complexity in sim data to test SALT2 training.

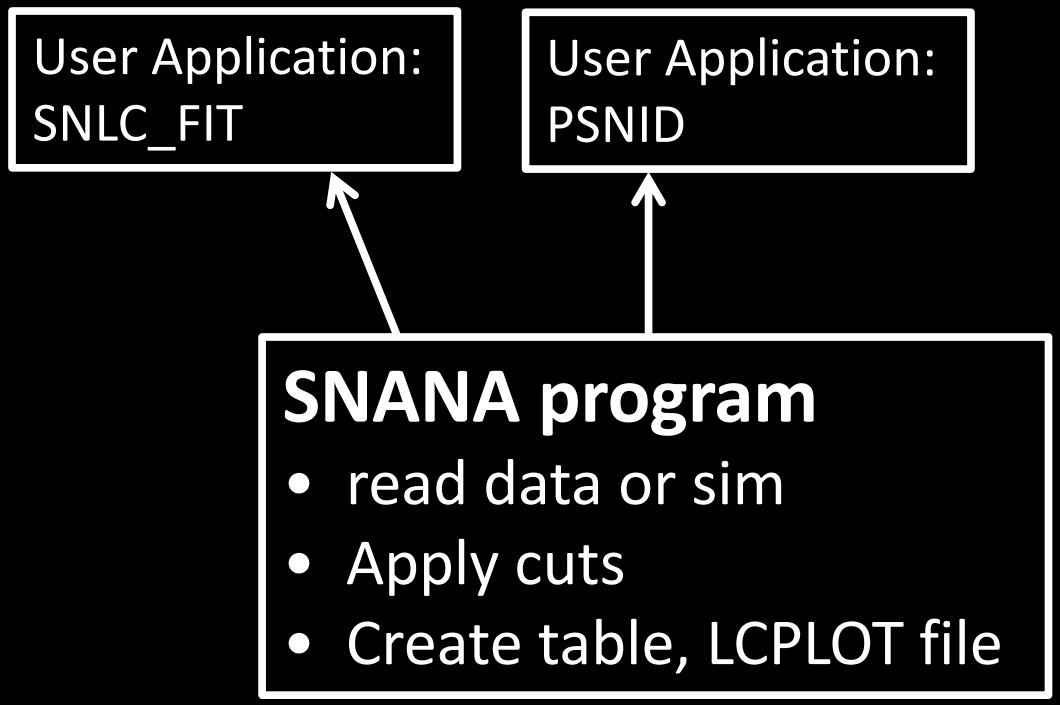
Analysis and Light Curve Fitting



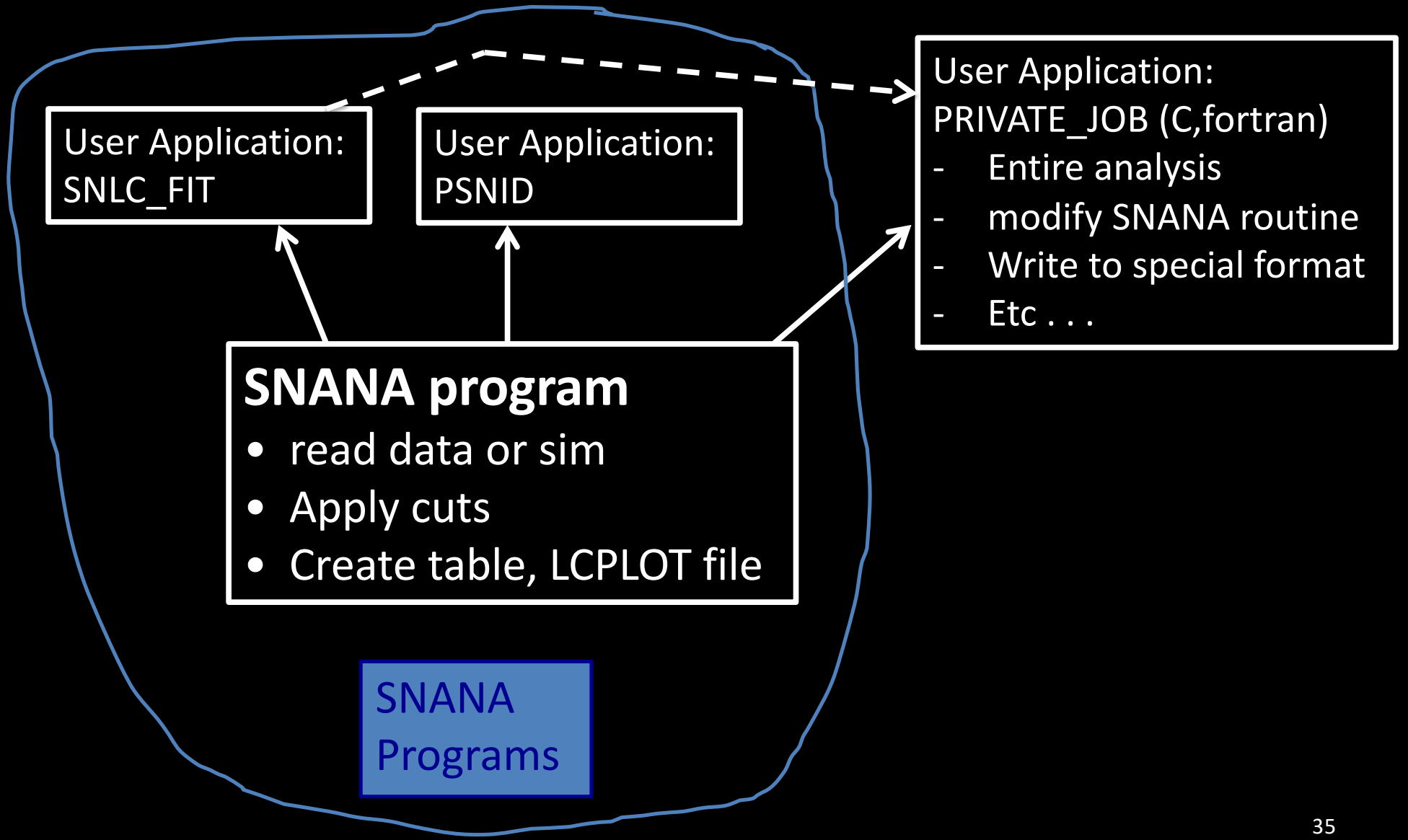
Architecture for Analysis

SNANA program
- read data or sim
- Apply cuts
- Create table, LCPLOT file

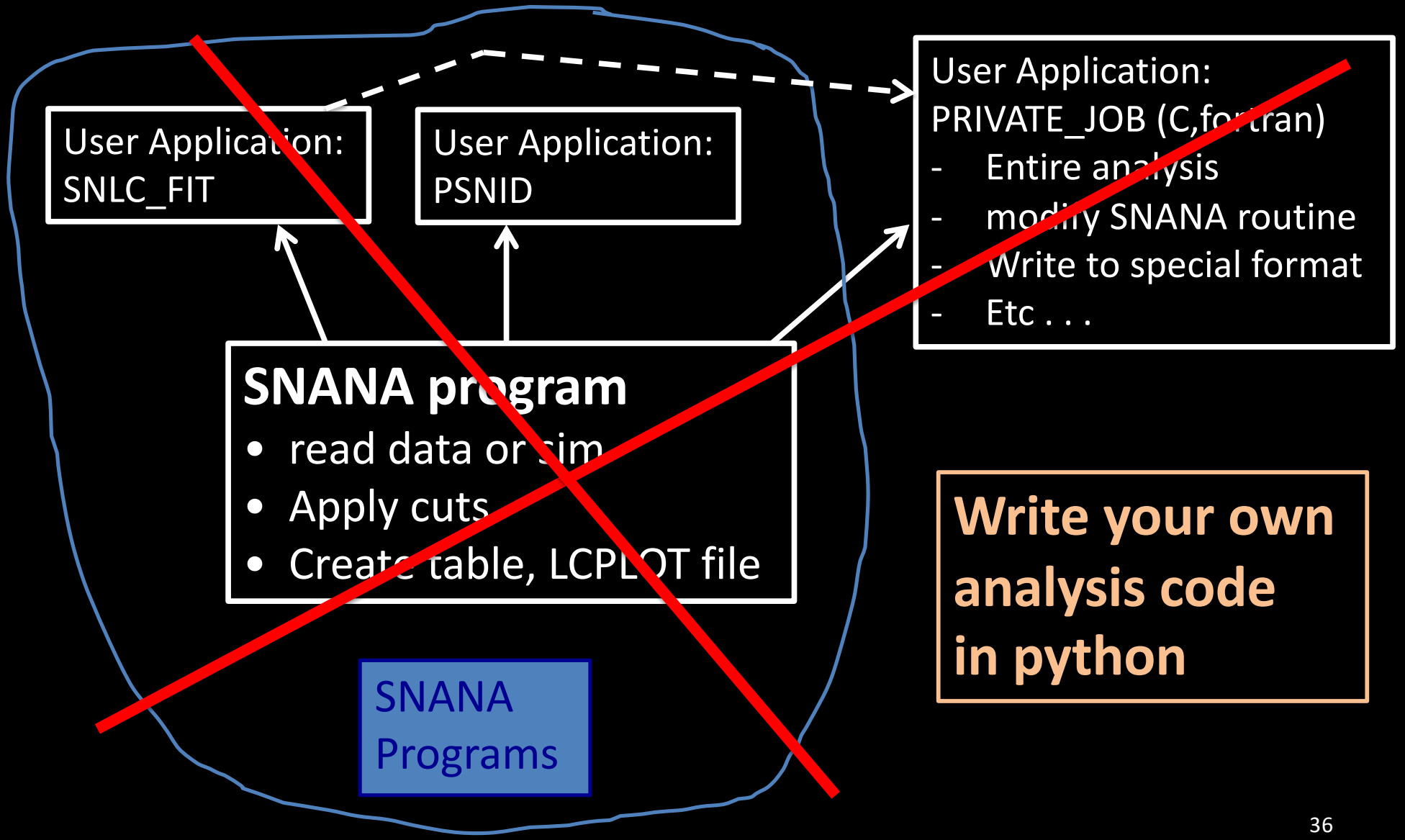
Interface for Analysis



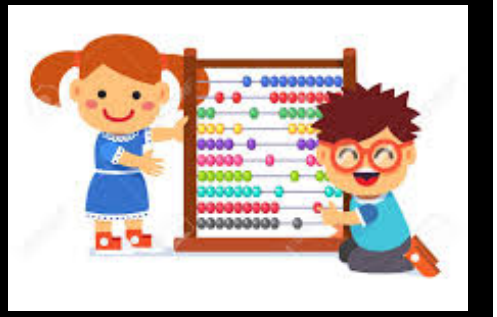
Interface for Analysis



Interface for Analysis



Misc Analysis Options

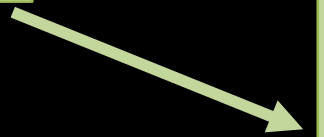


- ``CUTWIN" options to apply selection requirements
- Systematic variations (e.g., calibration, MW-extinct, bands..)
- Override data variables with supplement file (e.g., try out different pec. velocities)
- Apply fluxErr map to scale uncertainties

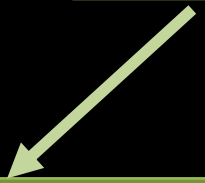
BEAMS with Bias Correction (BBC)

SALT2 fit params from:

- Real data
- Simulated data (Ia)
- Simulated data (CC)



BBC fit program

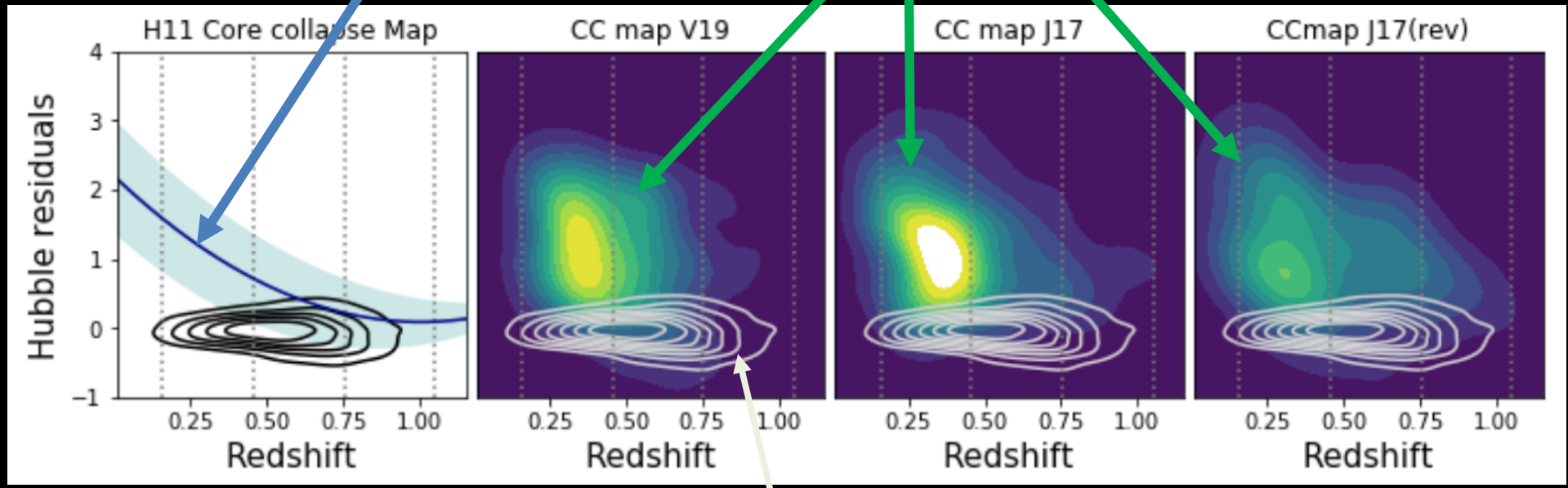


z-binned Hubble diagram corrected for bias from selection and CC contam.

BEAMS with Bias Correction (BBC)

Analytic CC prior (Hlozek 2011)

Simulated CC prior:



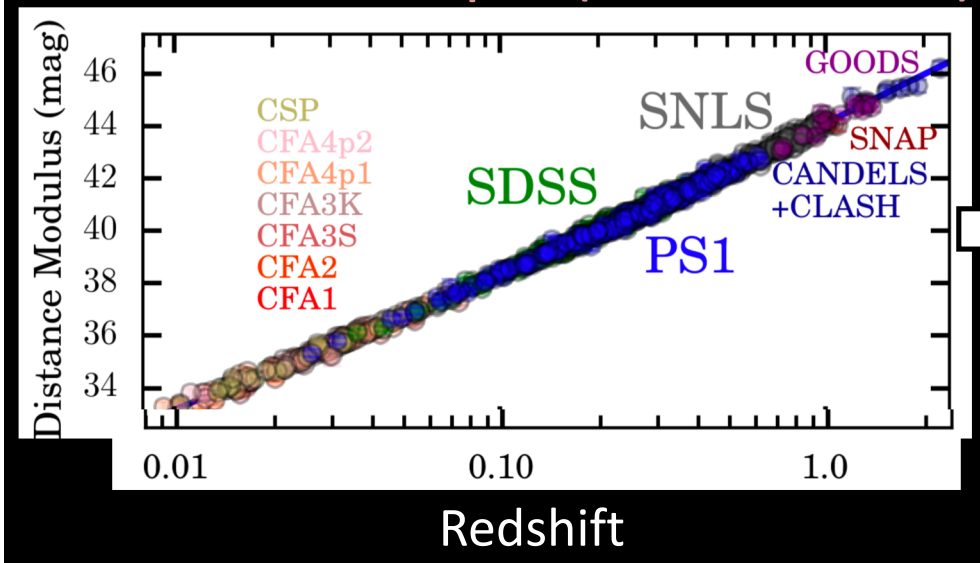
Plot courtesy of M.Vincenzi

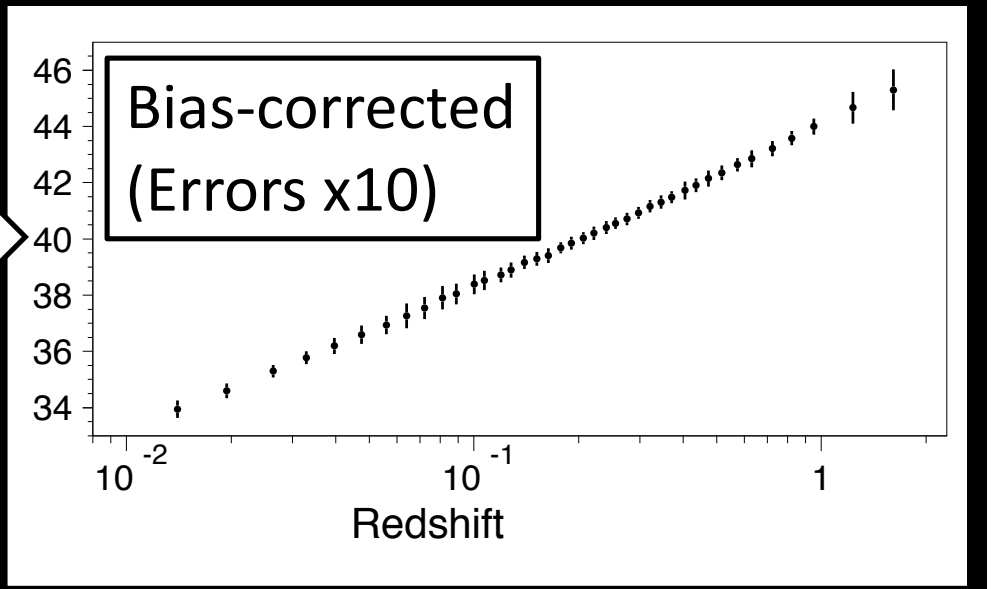
SNIa prior (analytic)

BBC Output

(input to Cosmology fit)

Pantheon Sample (Scolnic 2018)



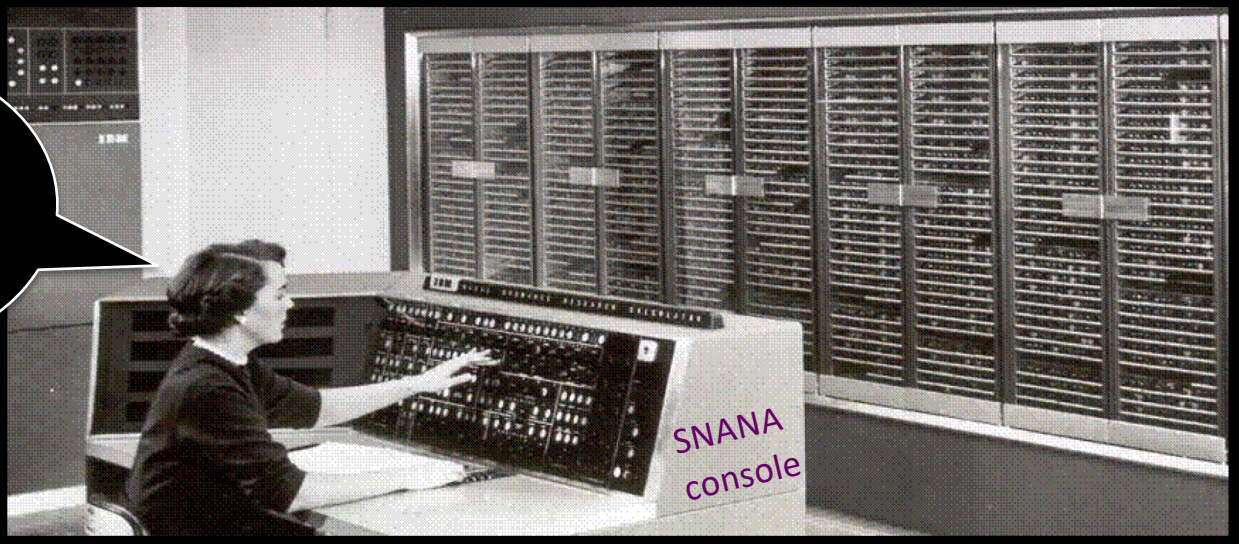


Systematics & Multi-Core Jobs

- Precision analyses require many iterations of analysis, each with a small variation in parameters or method.
- SNANA has tools to implement multi-iteration analyses using multi-core platforms.

  (e.g., Fermilab, NERSC, Argonne, Midway, Folio . . .)

Switching to C11 Intrinsic scatter model



Systematics & Multi-Core Jobs

- Sim & Analysis codes read input instructions from a text file.
- Specify variations with **command-line overrides** to avoid more input files.
- SNANA Scripts use **command-line override** feature to launch multiple jobs in batch system.

sim_SNmix.pl
Launch multiple SIM(Ia+CC) jobs

split_and_fit.pl
Launch multiple lightcurve fit jobs

SALT2mu_fit.pl
Launch multiple BBC fit jobs

Systematics & Multi-Core Jobs

Example: Subset of sim jobs for JLA systematics. Each job → separate core

```
# vary intrinsic scatter models
GENVERSION: JLA_SDSS3year_G10smear
GENOPT:  GENMAG_SMEAR_MODELNAME  G10
GENOPT:  SEARCHEFF_SPEC_FILE  SPECEFF_SDSS/SEARCHEFF_SPEC_SDSS_G10smear.DAT

GENVERSION: JLA_SDSS3year_COHsmear
GENOPT:  GENMAG_SMEAR 0.13  GENMAG_SMEAR_MODELNAME NONE
GENOPT:  SEARCHEFF_SPEC_FILE  SPECEFF_SDSS/SEARCHEFF_SPEC_SDSS_COHsmear.DAT

GENVERSION: JLA_SDSS3year_C11-0smear
GENOPT:   GENMAG_SMEAR_MODELNAME C11_0
GENOPT:   SEARCHEFF_SPEC_FILE  SPECEFF_SDSS/SEARCHEFF_SPEC_SDSS_C11-0smear.DAT

GENVERSION: JLA_SDSS3year_C11-1smear
GENOPT:   GENMAG_SMEAR_MODELNAME C11_1
GENOPT:   SEARCHEFF_SPEC_FILE  SPECEFF_SDSS/SEARCHEFF_SPEC_SDSS_C11-1smear.DAT

GENVERSION: JLA_SDSS3year_C11-2smear
GENOPT:   GENMAG_SMEAR_MODELNAME C11_2
GENOPT:   SEARCHEFF_SPEC_FILE  SPECEFF_SDSS/SEARCHEFF_SPEC_SDSS_C11-2smear.DAT

GENVERSION: JLA_SDSS3year_NOsmear
GENOPT:   GENMAG_SMEAR_MODELNAME NONE
GENOPT:   SEARCHEFF_SPEC_FILE  SPECEFF_SDSS/SEARCHEFF_SPEC_SDSS_NOsmear.DAT
```

sim_SNmix.pl
Launch multiple SIM(Ia+CC) jobs

Systematics & Multi-Core Jobs

```
FITOPT: MAGOBS_SHIFT_ZP 'g .01'
FITOPT: MAGOBS_SHIFT_ZP 'r .01'
FITOPT: MAGOBS_SHIFT_ZP 'i .01'
FITOPT: MAGOBS_SHIFT_ZP 'z .01'
FITOPT: FITMODEL_NAME  'SALT2.JLA_systematic/sys0'
FITOPT: FITMODEL_NAME  'SALT2.JLA_systematic/sys1'
FITOPT: FITMODEL_NAME  'SALT2.JLA_systematic/sys2'
FITOPT: FITMODEL_NAME  'SALT2.JLA_systematic/sys3'
FITOPT: FITMODEL_NAME  'SALT2.JLA_systematic/sys4'
FITOPT: MWEBV_SCALE 1.10  MWEBV_SHIFT   0.00
FITOPT: MAGOBS_SHIFT_ZP 'g 0.0024 r -0.003 i -0.008 z -0.013'
FITOPT: MAGOBS_SHIFT_PRIMARY 'g  0.00    r -0.00   i -0.00   z 0.0'
FITOPT: MAGOBS_SHIFT_PRIMARY 'g -0.0008  r -0.011  i -0.0051 z 0.016'
FITOPT: MAGOBS_SHIFT_PRIMARY 'g -0.011   r -0.0053 i  0.0014 z 0.0056'
FITOPT: MAGOBS_SHIFT_PRIMARY 'g -0.0037  r -0.0066 i -0.0043 z 0.008'
```

Example: Subset of fit jobs for PS1 systematics

split_and_fit.pl
Launch multiple lightcurve fit jobs

Systematics & Multi-Core Jobs

```
# define Light curve fit output from split_and_fit
INPDIR+:  $DES_ROOT/analysis/DES_LCFITS
INPDIR+:  $DES_ROOT/analysis/LOWZ_LCFITS
INPDIR+:  $DES_ROOT/analysis/FOUND_LCFITS

# define BBC variations
MUOPT:  simfile_biascor=DES_variant1.FITRES,LOWZ_variant1.FITRES
MUOPT:  simfile_biascor=DES_variant2.FITRES,LOWZ_variant2.FITRES
MUOPT:  nbin_logmass=2  nzbin=16
MUOPT:  CUTWIN x1 -2 2
etc ...
```

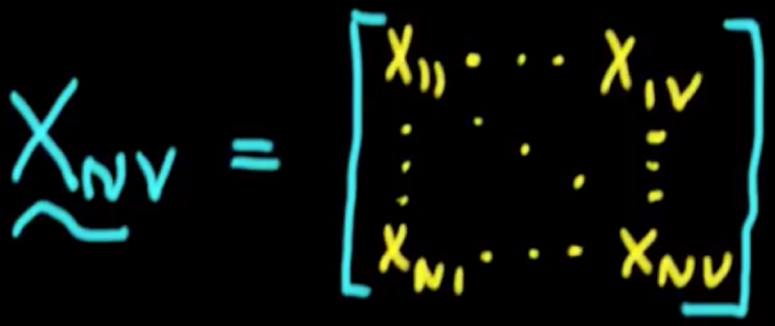
SALT2mu_fit.pl
Launch multiple BBC fit jobs

Systematics Summary

- Systematics are expressed via
  - GENOPTs (sim biasCor variations)
  - FITOPTs (LC fit variations)
  - MUOPTs (BBC variations)

- Creating and testing these ~100 variations is a major part of SNIa-cosmology analysis.

Stat + Syst Covariance Matrix (createCov.py)



- After LC fit and BBC, create cov(stat+syst) for CosmoMC (cosmology fit program)
- Includes systematics from fit options (FITOPT) and BBC options (MUOPT)
- Allows arbitrary error-scaling for each systematic, or each class of systematic.
- Internal bookkeeping for ~100 systematics.

SNANA Output

SIMULATION Output

Data Files

- FITS format for large jobs

  or

- ASCII format for testing (1 file per SN)

SIMULATION Output

Data Files

- FITS format for large jobs

or

- ASCII format for testing (1 file per SN)

ASCII Summary File

- 1 row per accepted SN

or

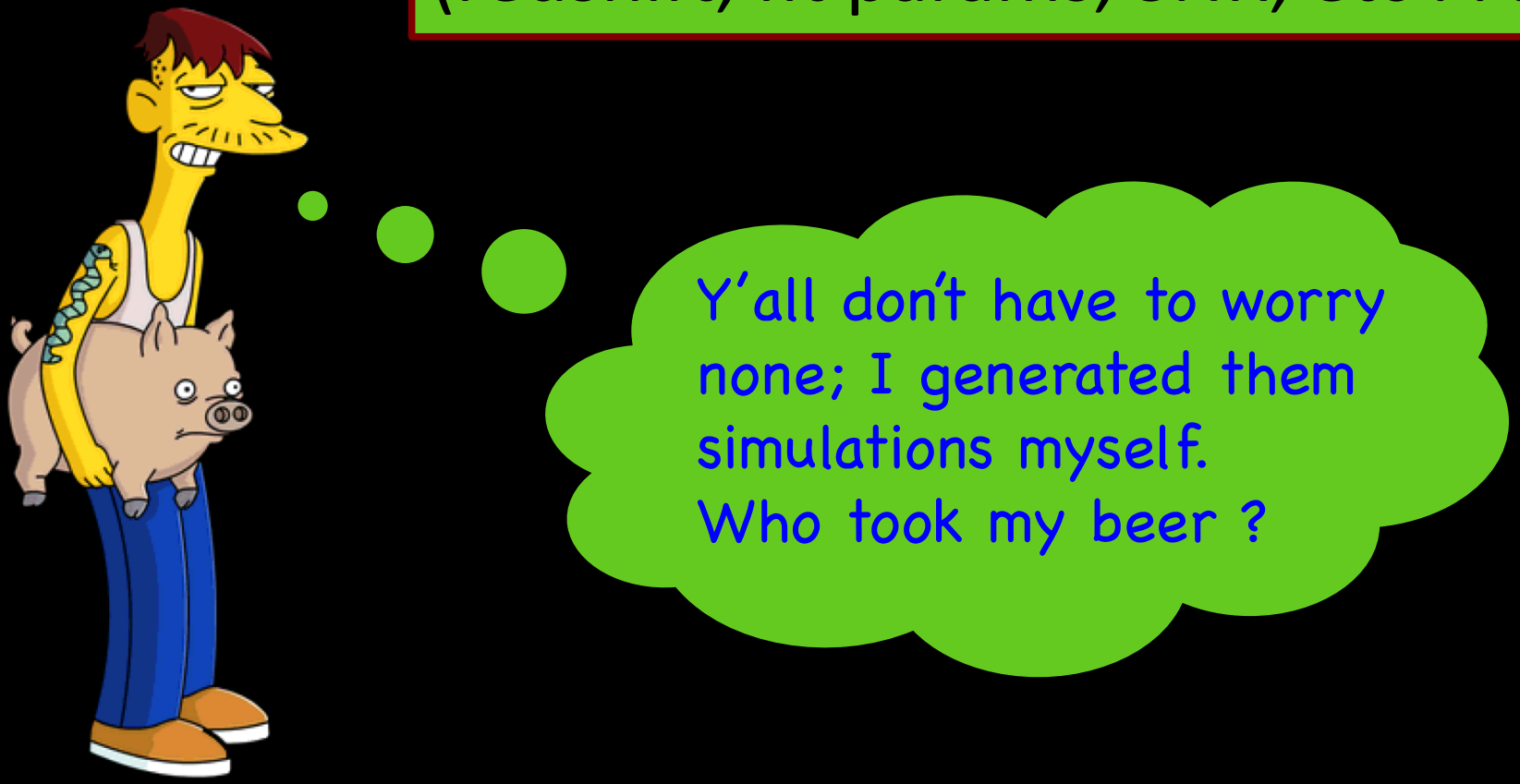
- 1 row per generated SN (for efficiency)

SIMULATION Output

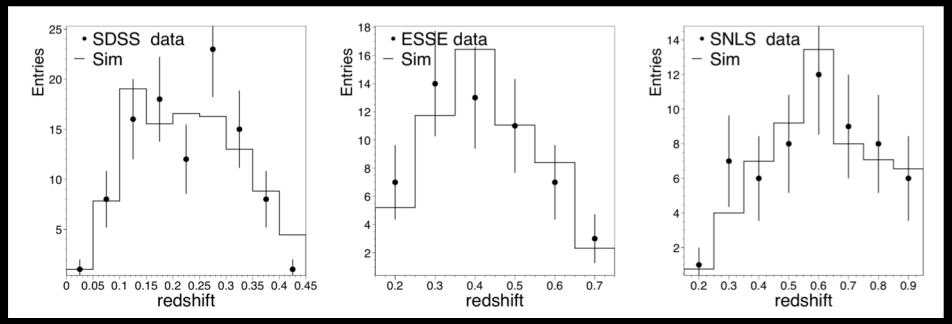
Never Trust Simulation Output

→

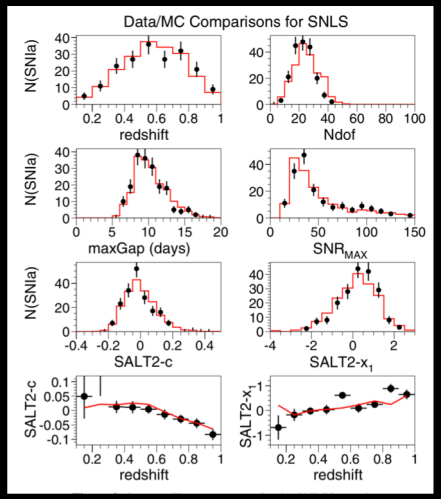
Always check data/MC distributions (redshift, fit params, SNR, etc . . .)



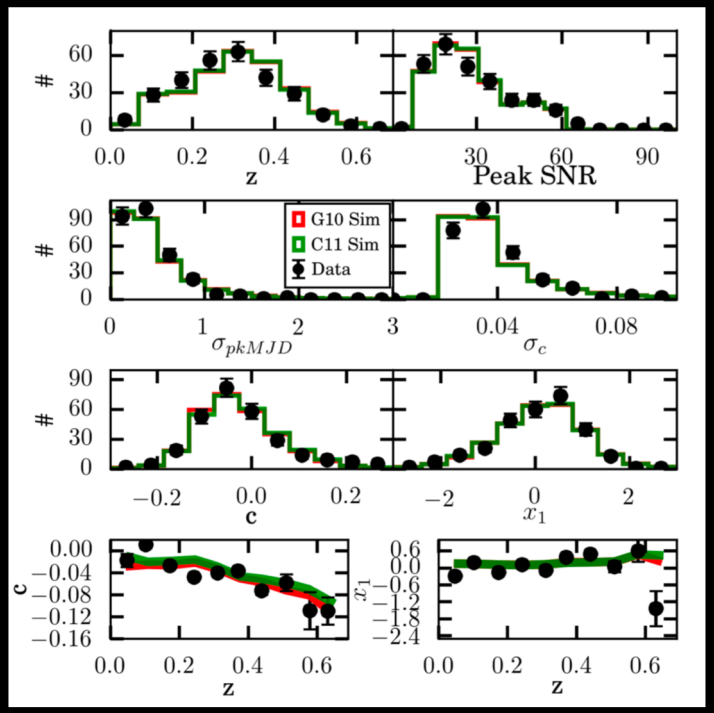
Simulation Output vs. Data



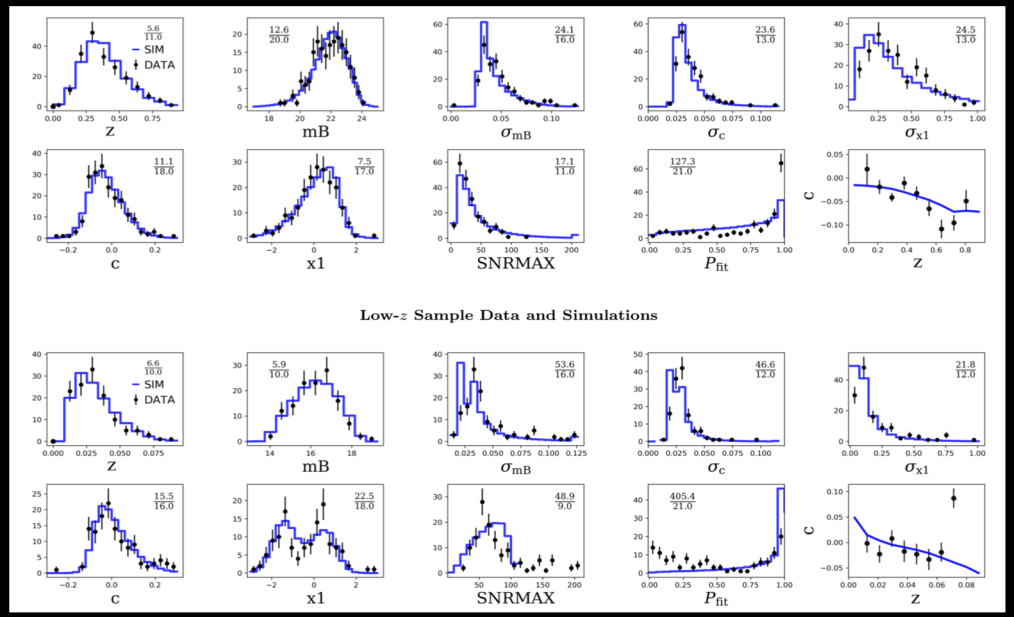
SDSS (Kessler et al 2009)



JLA 2014

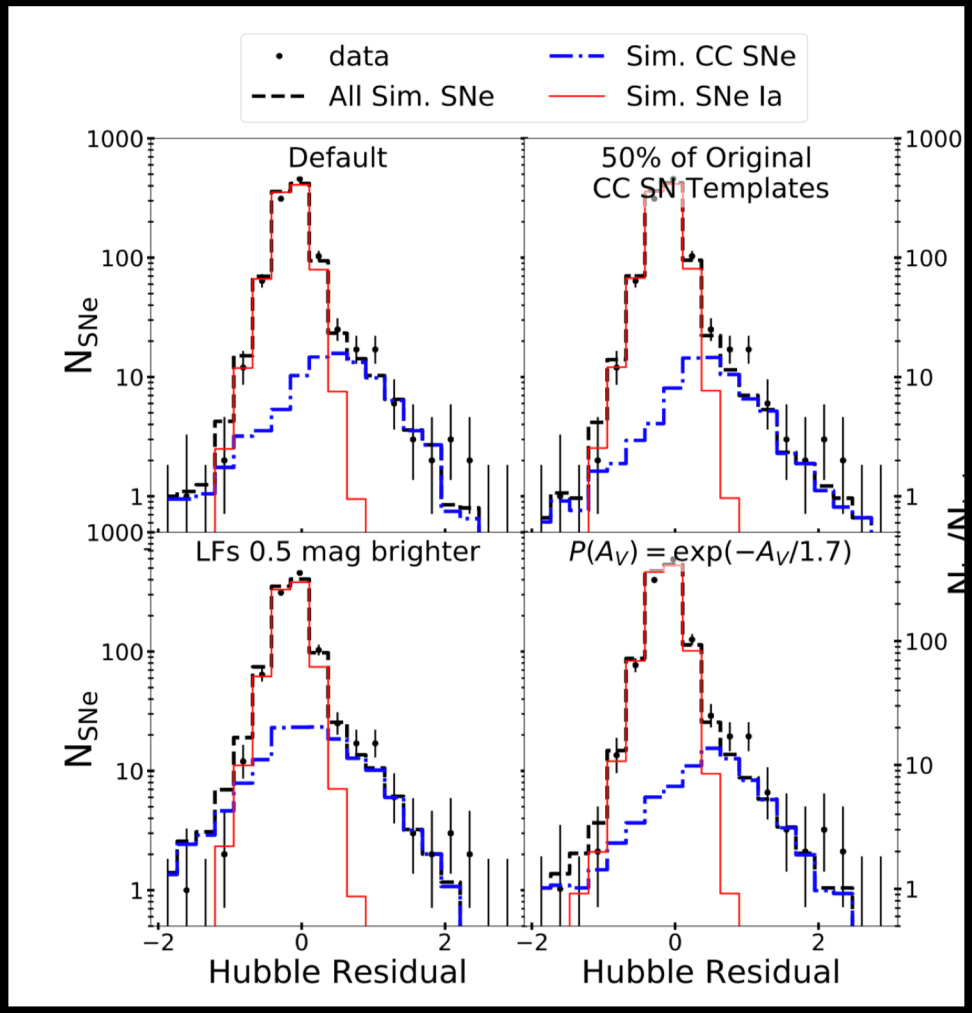


Pantheon (Scolnic et al 2018)



DES 3-Year (Brout et al 2019)

Simulation Output vs. Data



With ***photometric-ID***, Sim(Ia+CC) should match observed HR

PS1 – Jones et al., 2018

Analysis Output → SNTABLEs

- SNANA table before fit (1 row per SN)
- FITRES table after cuts+fit (1 row per SN)
- LCPLOT table with light curve & best-fit curve

Analysis Output → SNTABLEs

- SNANA table before fit (1 row per SN)
- FITRES table after fit (1 row per SN)
- LCPLOT table with light curve & best-fit curve

Formats:
- TEXT (1 file per table)
- HBOOK (all tables → 1 file)
- ROOT (all tables → 1 file)

- Easier to install SNANA without HBOOK or ROOT
  (but will miss most output variables from analysis)
- New format can be added if interface routines are provided

Analysis Output → SNTABLEs

- SNANA table before fit (1 row per SN)
- FITRES table after fit (1 row per SN)

- ALL variables stored in **HBOOK** & **ROOT** format
  (compared to TEXT, more efficient to read & write)
- Subset stored in **TEXT** for input to cosmology fit
- Utility to append **TEXT** file (from HBOOK or ROOT)
- See "sntable_dump" to view, dump, append tables
- Slowly moving away **TEXT** (except for debug)

Putting it all Together in a Photometric-SNIa-Cosmology Analysis

SNANA

- Simulate SNIa biasCor
- Simulate SNCC training
- Light curve fitting (data+sim)
- Beams with Bias Corr (BBC)
- Create COV(stat+syst)

Outside SNANA

- Classification
- Cosmology fit

Putting it all Together in a Photometric-SNIa-Cosmology Analysis

SNANA

- Simulate SNIa biasCor
- Simulate SNCC training
- Light curve fitting (data+sim)
- Beams with Bias Corr (BBC)
- Create COV(stat+syst)

Outside SNANA

- Classification
- Cosmology fit

``Pippin'' glues it all together:

https://github.com/Samreay/Pippin

Putting it all Together in a Photometric-SNIa-Cosmology Analysis

SNN: Moller and Boissiere, arXiv:1901.06384
SNIRF: Dai et al, arXiv:1701.05689
(more classifiers can be added)

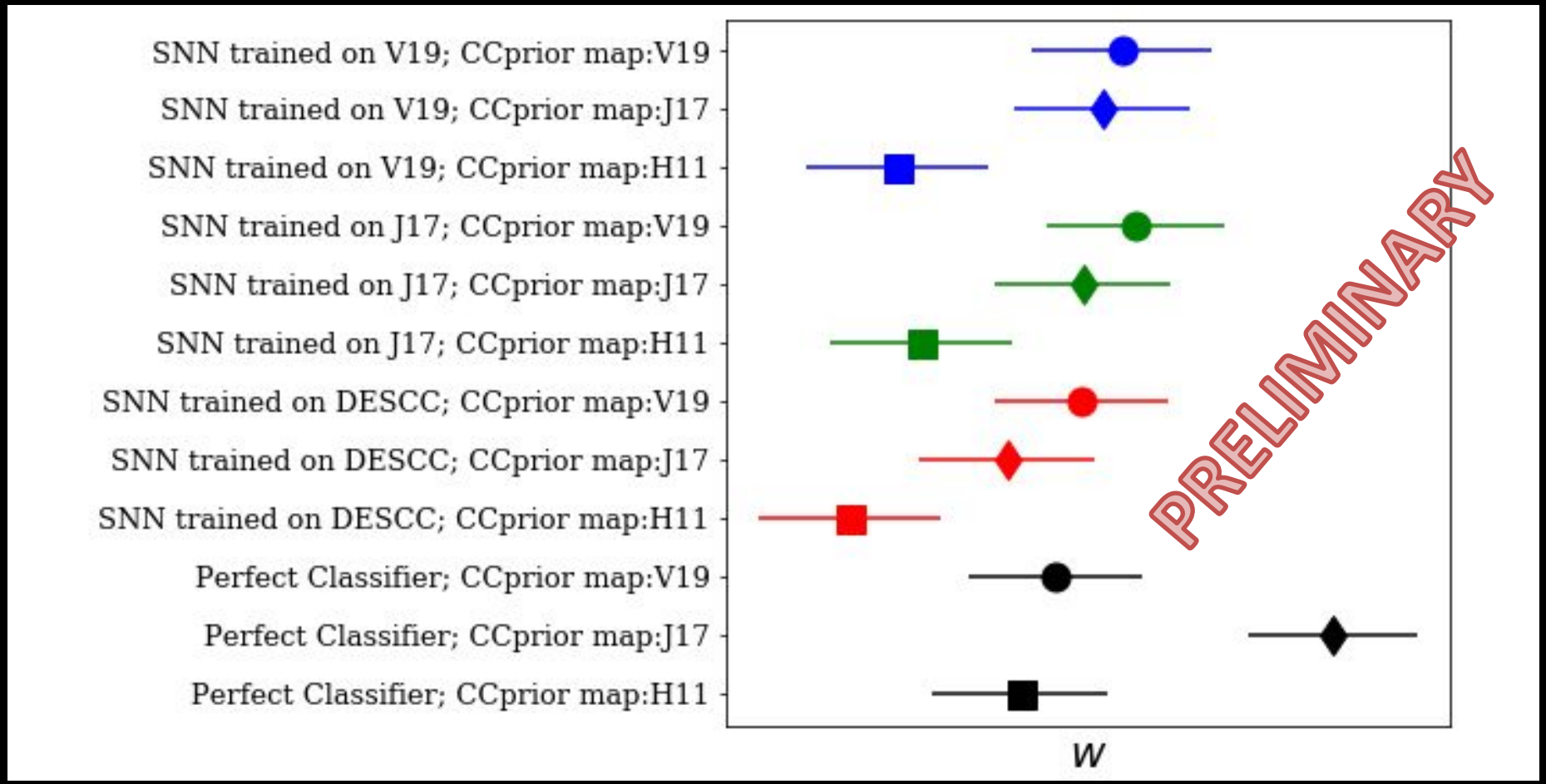
Outside SNANA
- Classification
- Cosmology fit

CosmoMC: Lewis and Bridle, arXiv:astro-ph/0205436

``Pippin'' glues it all together:
https://github.com/Samreay/Pippin

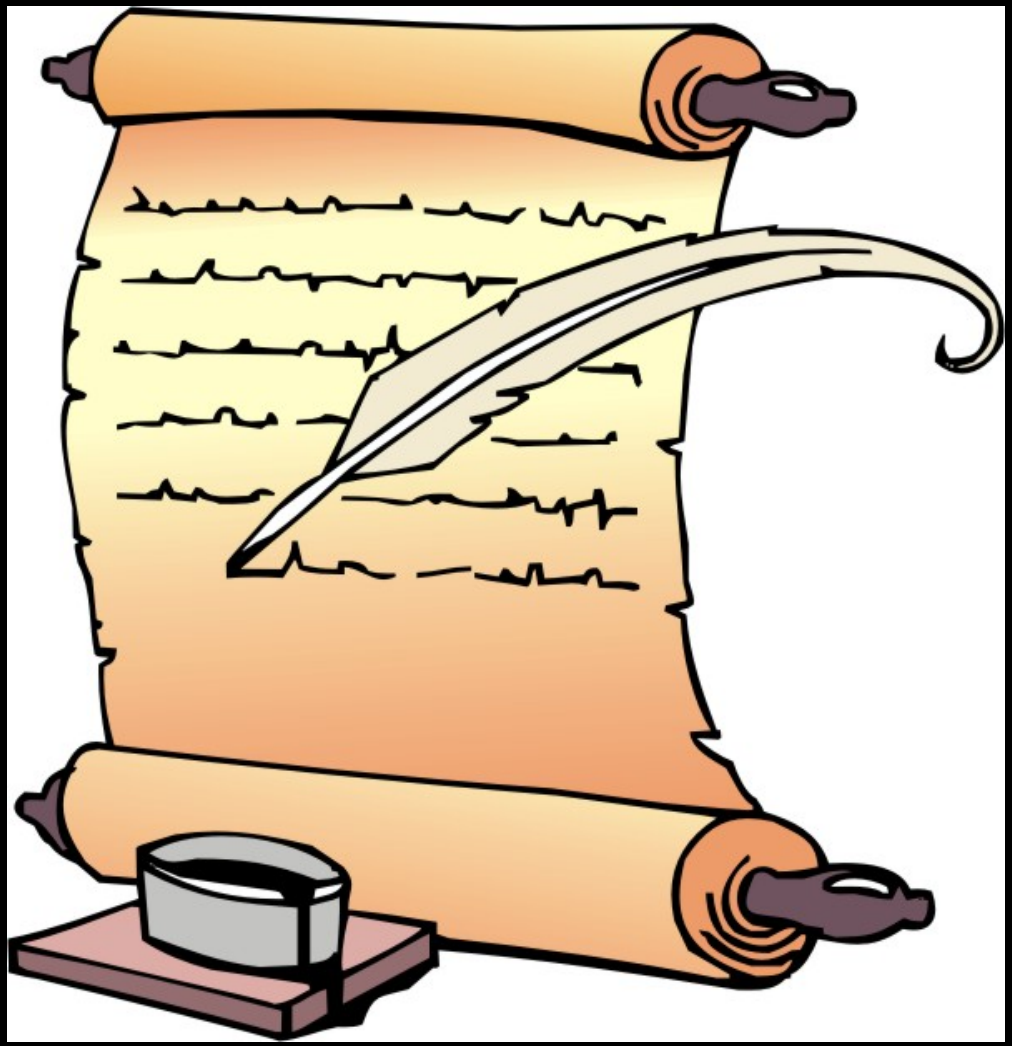
Power of Pippin: Illustration with Simulated DES+LOWZ

Classifier x simCC(training) x simCC(BBC prior)



Plot from M. Vincenzi & S.Hinton

SNANA Documentation



SNANA Documentation

- Learning SNANA is like learning to fly a plane; forget the manual, you need experienced person in the cockpit with you.

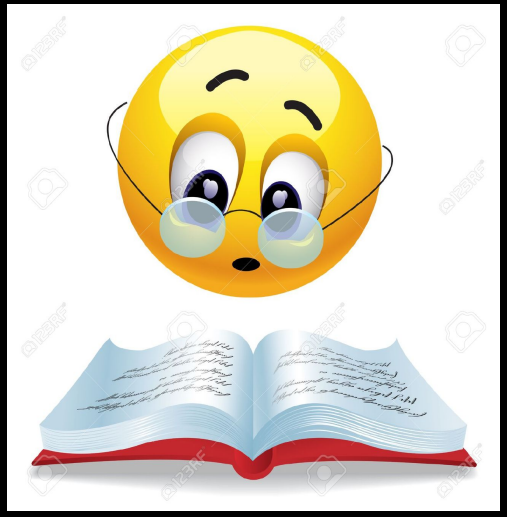
Why did my job abort ?

```
LC: 2000
ION: CODETEST_SALT2_SDSS_TEXT_XXX
CE:  RANDOM
:   SALT2.Guy10

E_INCLUDE:  $SNANA_TESTS/inputs/SIMGEN_INCLUDE_SALT2.input

ILE:  $SNDATA_ROOT/simlib/SDSS/SDSS2005_ugriz.SIMLIB

_Ia:  120 106
FF: 50000
COR_FILE:  $SNDATA_ROOT/kcor/SDSS/SDSS_Doi2010/kcor_SDSS_Bessell90_BD17.fits

GENMAG_SMEAR_MODELNAME: G10

SEARCHEFF_PIPELINE_LOGIC_FILE: $SNDATA_ROOT/models/searcheff/SEARCHEFF_PIPELINE_LOGIC.DAT
SEARCHEFF_PIPELINE_EFF_FILE:   $SNDATA_ROOT/models/searcheff/SEARCHEFF_PIPELINE_SDSS.DAT
SEARCHEFF_SPEC_FILE:  $SNDATA_ROOT/models/searcheff/SEARCHEFF_SPEC_SDSS.DAT
SEARCHEFF_zHOST_FILE: $SNDATA_ROOT/models/searcheff/SEARCHEFF_zHOST_SDSS.DAT

APPLY_SEARCHEFF_OPT: 3

# selection criteria for generation
GENFILTERS:        ugriz
GENRANGE_RA:      -50.0     +59.0   deg
GENRANGE_DECL:    -1.258    +1.258  deg
GENRANGE_PEAKMJD:  53625.0  53685.0
GENSIGMA_SEARCH_PEAKMJD:  1.0

GENRANGE_REDSHIFT:  0.02    0.45
GENSIGMA_REDSHIFT:  0.0012

GENRANGE_TREST:   -20.0    80.0     # rest epoch relative to peak (days)

RANSEED: 128473       # random number seed

# smear flags: 0=off, 1=on
SMEARFLAG_FLUX:    1  # photo-stat smearing of signal, sky, etc ...
SMEARFLAG_ZEROPT:  1  # smear zero-point with zptsig

DNDZ: POWERLAW 2.6E-5 1.5   # SN rate ~ (1+z)^1.5

FORMAT_MASK:  2  ! 2=terse/text  32=FITS  1=verbose

SIMGEN_DUMP: 14  CID GENTYPE SNTYPE GENZ DLMAG S2x1 S2c
                 SNRMAX SNRMAX2 SNRMAX3
                 SNRMAX_g  SNRMAX_r  SNRMAX_i SNRMAX_z
```

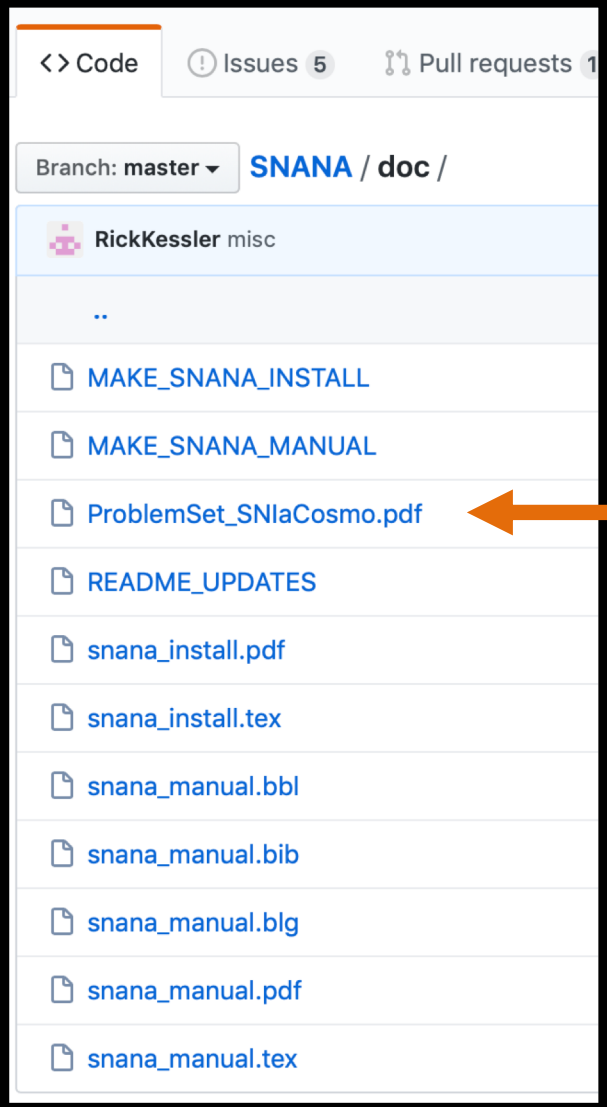
What does the red engine light mean ?

Manual

- 180 page manual with no interface except ‘preview’
- Reading manual not so helpful, but recommend skimming table-of-contents



For Beginners and Profs looking for Homework Problems



Tracking Changes

```
des20.fnal.gov> tail -50 $SNANA_DIR/doc/README_UPDATES
 ================================================================

       v10_42f (Feb 8 2016)

       ***** IMPORTANT(v10_42f) *****

       ***** USEFUL(v10_42f) *****

       ***** MISCELLANEOUS(v10_42f) *****

  sntools_output_root.c:
      in SNTABLE_READPREP_ROOT, add missing return(NVAR) at end.
      This bug was tripped up by the recent -O1 optimization, but
      amazingly seemed to work on other machines

      Add MJD to SNLCPAK tree, so that we don't have to use the
      clumsy method pf MJD = PEAKMJD+TOBS.
```

Ignore boring "**MISC**" that is mainly for me

Tracking Changes

```
------------------------------------------------------------
       v10_42g (Feb 19 2016)

       ***** IMPORTANT(v10_42g) *****

 SALT2mu.c : major refactor and update to implement BEAMS-like
             fit using simulation to define the CC prior.
             See new inputs: simfile_ccprior and varname_p1a

       ***** USEFUL(v10_42g) *****

 New plotting function $SNANA_DIR/util/ovdatamc.py (by D.Jones)
 operates on ascii FITRES files from data and sim.
 Overlays simulation separately for SNIa and SNCC.

 snana.car: new SNLCINP namelist SIMLIB_OUT = 'bla.simlib'
            will create simlib file from data.

       ***** MISCELLANEOUS(v10_42g) *****

snlc_sim.c, sntools_host.c:
   New sim-input key
      HOSTLIB_GALID_PRIORITY: 0 500000
   to give priority to GALID range.

   Fix minor bug in gen_AV() [found by D.Jones]

split_and_fit.pl : new key SALT2mu_SIMVERSION to pass simFile
                   to SALT2mu program (for bias cor and CCprior).
```

You should read **IMPORTANT** & **USEFUL** updates

Miscellaneous

- Report bugs ASAP as Github issue; don't just hack a private fix for yourself.
- Report compilation warnings.
- Think about how you can contribute to SNANA.
- In papers, SNANA citation is not enough; ***also cite source of models, template data, galaxy catalogs, etc***.

Conclusion

From the SNANA Legal Team:

- **SIDE EFFECTS** include, but are not limited to: confusion, frustration, watery eyes, head-aches, weight loss, weight gain, systematics-limited results, incorrect results, denial of tenure.
- Do not drive or operate heavy machinery while using SNANA.