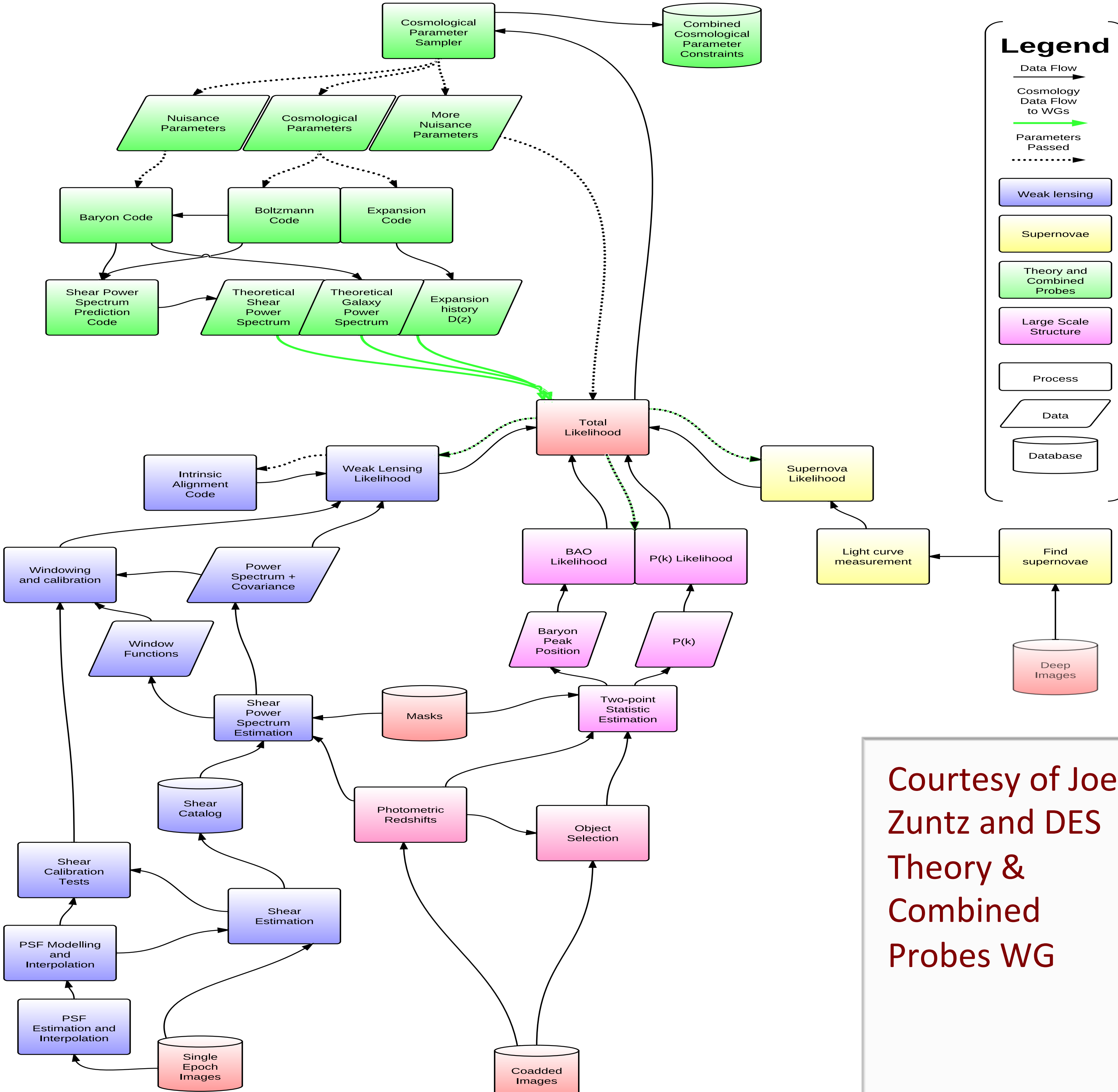
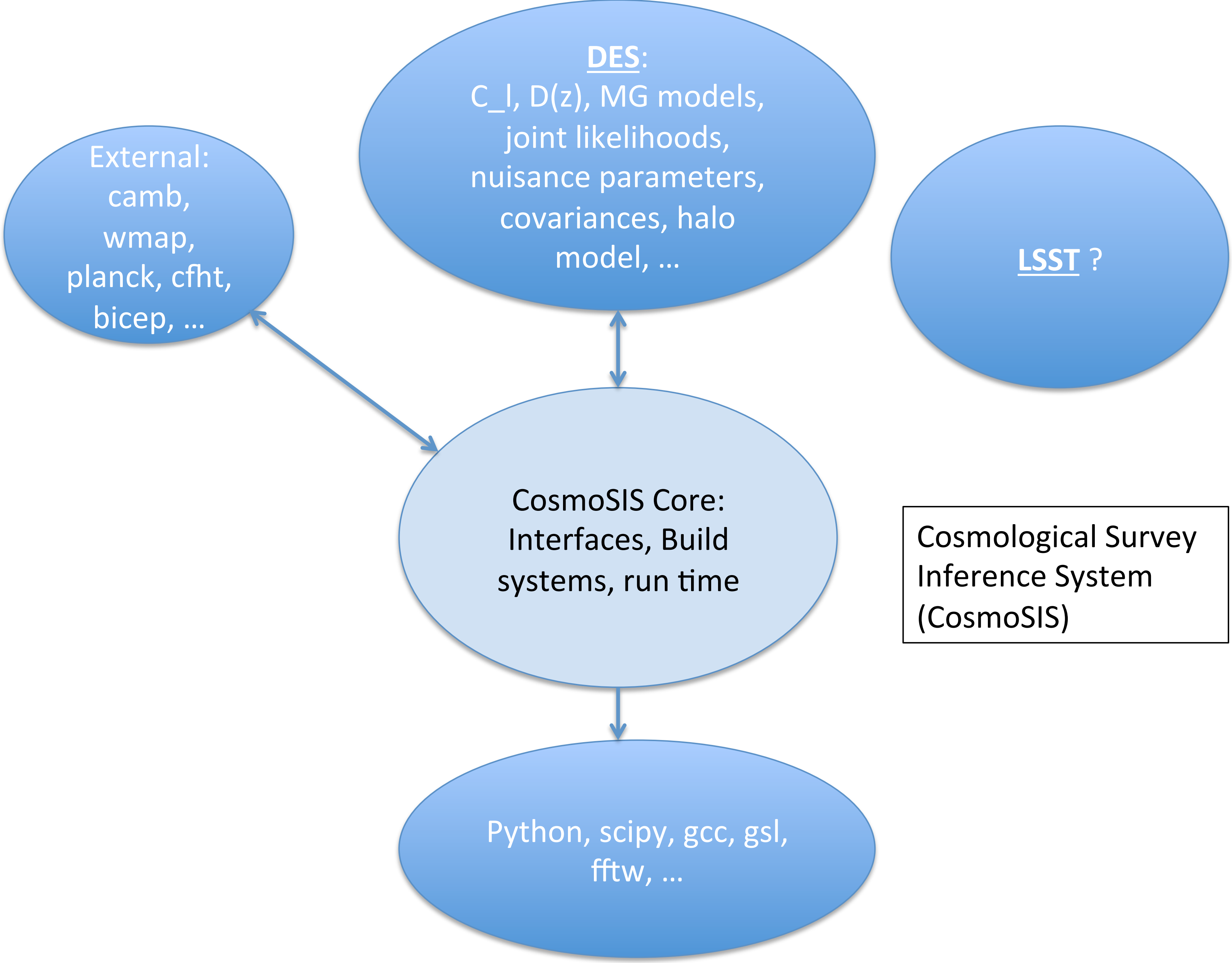
Analysis Frameworks for DES and LSST

Sarah Bridle, Scott Dodelson, Elise Jennings, Steve Kent, Alessandro Manzotti, Marc Paterno, Doug Rudd, Saba Sehrish, Joe Zuntz

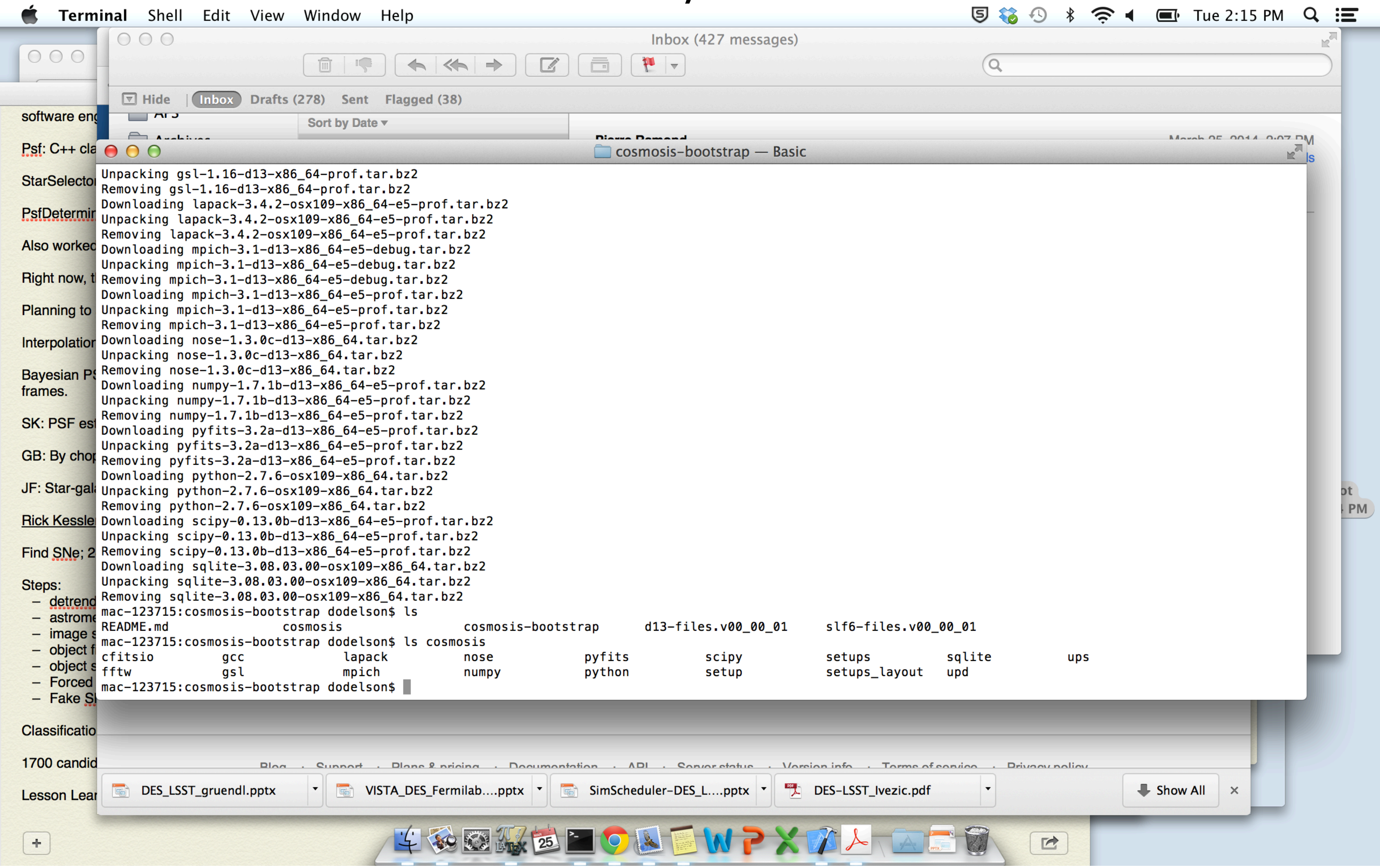


Courtesy of Joe Zuntz and DES Theory & Combined Probes WG



Cosmological Survey Inference System (CosmoSIS)

One-click download of all dependencies (currently for mavericks and SLF6) and CosmoSIS core



Repos: Collaboration members push/ pull their modules for the benefit of all

Terminal — Basic

```
bash-4.1$ hg clone https://dodelson@bitbucket.org/joezuntz/des-pipe
warning: bitbucket.org certificate with fingerprint 24:9c:45:8b:9c:aa:ba:55:4e:01:6d:58:ff:e4:28:7d:2a:14:ae:3b not verif
ied (check hostfingerprints or web.cacerts config setting)
http authorization required
realm: Bitbucket.org HTTP
user: dodelson
password:
warning: bitbucket.org certificate with fingerprint 24:9c:45:8b:9c:aa:ba:55:4e:01:6d:58:ff:e4:28:7d:2a:14:ae:3b not verif
ied (check hostfingerprints or web.cacerts config setting)
destination directory: des-pipe
warning: bitbucket.org certificate with fingerprint 24:9c:45:8b:9c:aa:ba:55:4e:01:6d:58:ff:e4:28:7d:2a:14:ae:3b not verif
ied (check hostfingerprints or web.cacerts config setting)
requesting all changes
warning: bitbucket.org certificate with fingerprint 24:9c:45:8b:9c:aa:ba:55:4e:01:6d:58:ff:e4:28:7d:2a:14:ae:3b not verif
ied (check hostfingerprints or web.cacerts config setting)
adding changesets
adding manifests
adding file changes
added 848 changesets with 3010 changes to 1725 files
warning: bitbucket.org certificate with fingerprint 24:9c:45:8b:9c:aa:ba:55:4e:01:6d:58:ff:e4:28:7d:2a:14:ae:3b not verif
ied (check hostfingerprints or web.cacerts config setting)
warning: bitbucket.org certificate with fingerprint 24:9c:45:8b:9c:aa:ba:55:4e:01:6d:58:ff:e4:28:7d:2a:14:ae:3b not verif
ied (check hostfingerprints or web.cacerts config setting)
warning: bitbucket.org certificate with fingerprint 24:9c:45:8b:9c:aa:ba:55:4e:01:6d:58:ff:e4:28:7d:2a:14:ae:3b not verif
ied (check hostfingerprints or web.cacerts config setting)
updating to branch default
```

Repos: Collaboration members push/ pull their modules for the benefit of all

Terminal — Basic

```
bash-4.1$ hg clone https://dodelson@bitbucket.org/joezuntz/des-pipe
warning: bitbucket.org certificate with fingerprint 24:9c:45:8b:9c:aa:ba:55:4e:01:6d:58:ff:e4:28:7d:2a:14:ae:3b not verif
ied (check hostfingerprints or web.cacerts config setting)
http authorization required
realm: Bitbucket.org HTTP
user: dodelson
password:
warning: bitbucket.org certificate with fingerprint 24:9c:45:8b:9c:aa:ba:55:4e:01:6d:58:ff:e4:28:7d:2a:14:ae:3b not verif
ied (check host
destination dir
warning: bitbuc
ied (check host
requesting all
warning: bitbuc
ied (check host
adding changese
adding manifest
adding file cha
added 848 chang
warning: bitbuc
ied (check host
warning: bitbuc
ied (check host
warning: bitbuc
ied (check host
updating to bra
```

Terminal — Basic

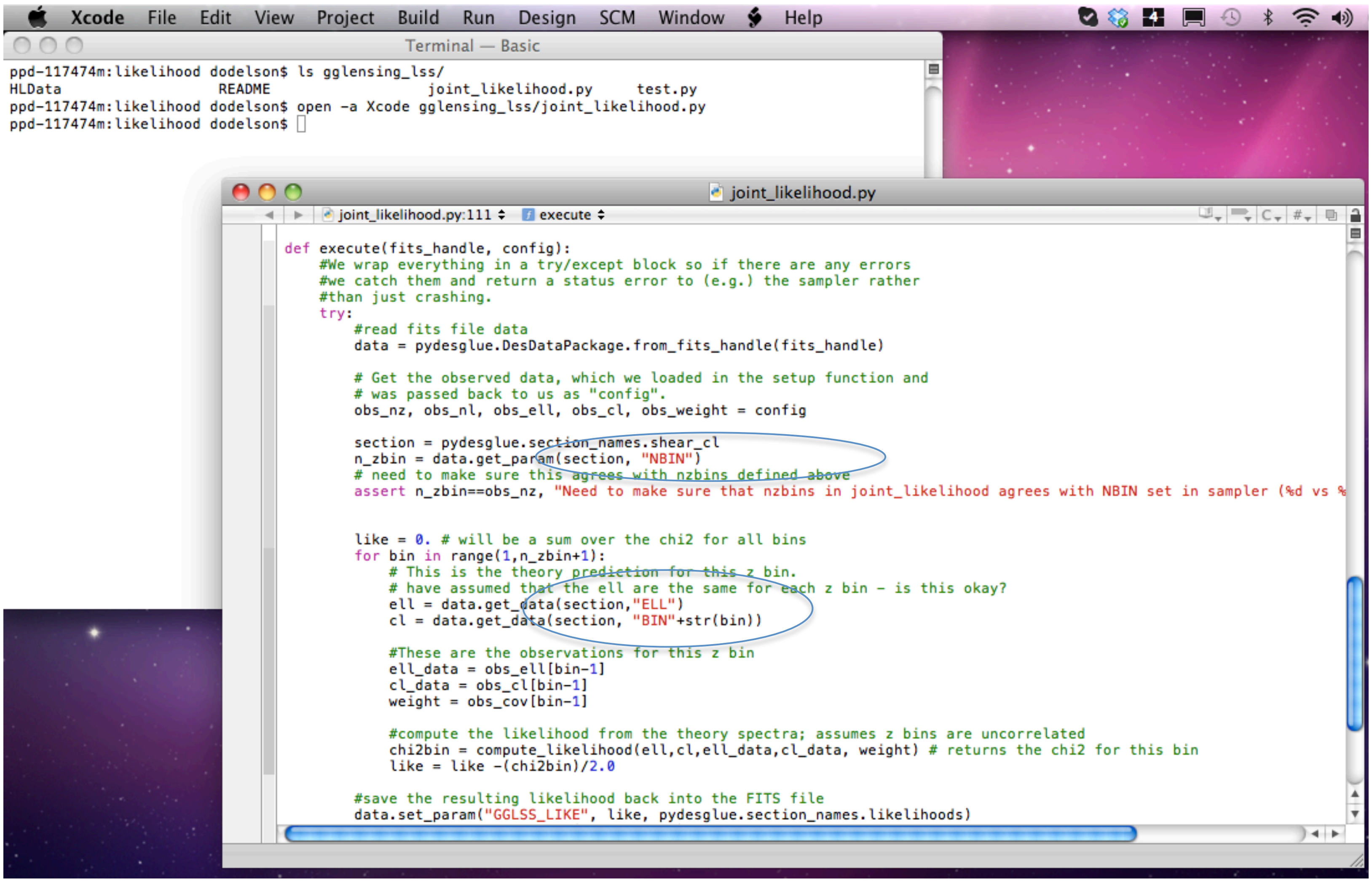
```
bash-4.1$ cd des-pipe/
bash-4.1$ ls
baryon       data         doc                       glue                  likelihood  makefiles        nonlinear              README    supernovae       utils
boltzmann    default_run  example                   interface_example     lss         mock_likelihood  number_density         sampler   templates
dark_energy  DEVELOPING   FORTRAN_INTERFACE_TOOLS   intrinsic_alignments  Makefile    MODULES          PYTHON_INTERFACE_TOOLS shear     TROUBLESHOOTING
bash-4.1$ ls lss/dodelson/
bias.f90  bin5  cl.f        dodelson_cl.py   dodelson_xi.py  IF1.f          InterpolateFile.f  polint.f  sigma.f           twodint.f  zeros_J0
bin1      chi.f cosmology.h dodelson_lss.py  hunt.f          interface.f90  Makefile           qsim.f    testinterface.py  w.f
bash-4.1$
```

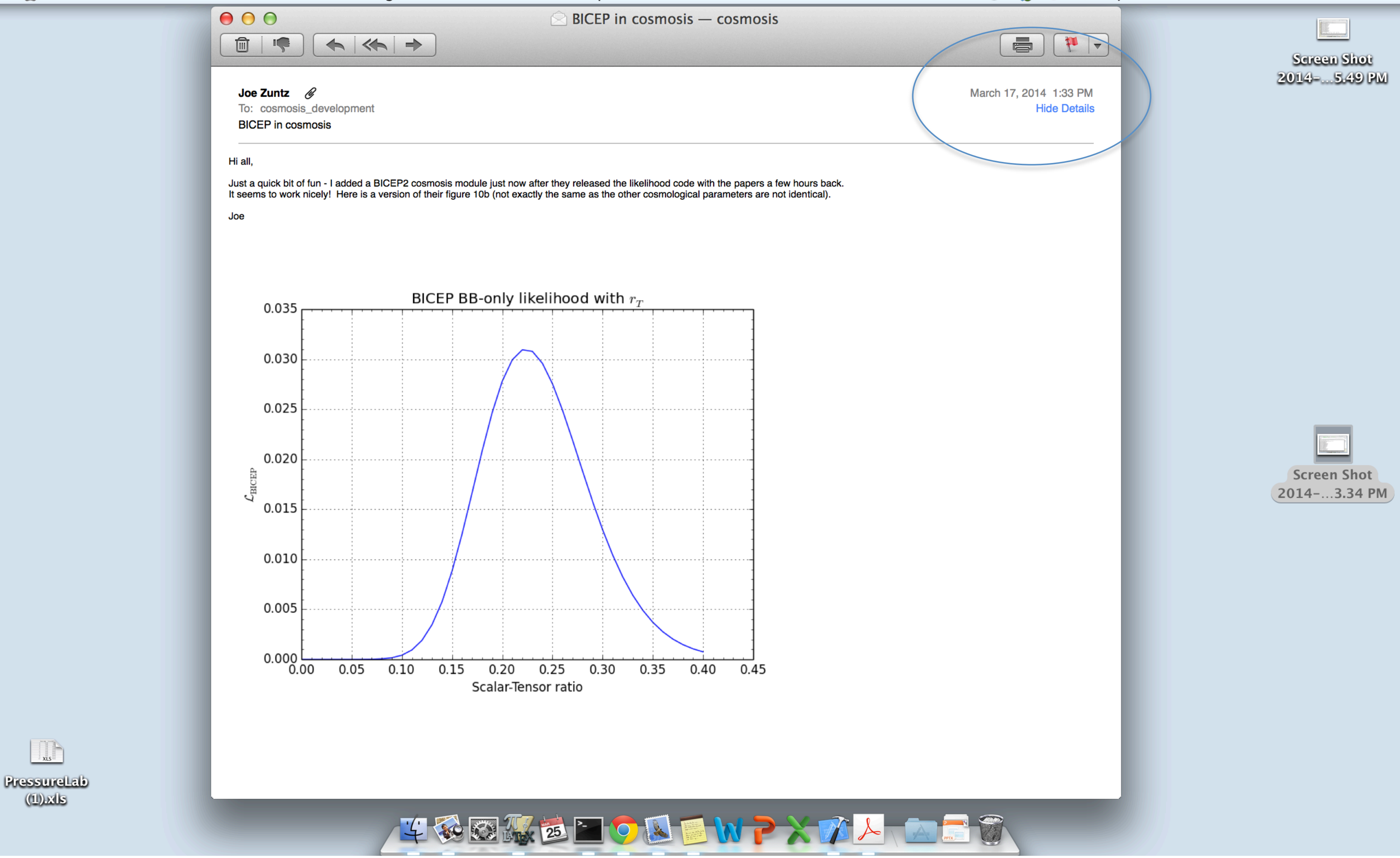
Development: Can now edit any of the files and add your code; run by simply putting a list of modules to include in a config file

Terminal — Basic

```
[pipeline]
root = ../..                                            ; The top-level directory where all the modules are to be found.
modules = dodelson_shear_single dodelson_like_single HST      ; This list of modules to be run.  They are described below.
likelihoods = WL HST                                    ; The list of likelihood values that will be generated by the pipeline and summed.
values = dodelson_params.txt                                       ;file containing parameter ranges and start values

; For example, the pymc sampler can read different options, say:
[pymc]
params = omega_m  h0  sigma_8                                     ;parameters to vary
values = shear_params.txt                                       ;file containing parameter ranges and start values
covmat = shear_covmat.txt                                       ;file containing covariance matrix
name = mice_shear.1                                              ;output dir name for the run
samples = 10000                                                       ;number of MCMC samples to get

; These sections describe what the different named modules are and where to find them.
; You can have modules here that are not actually used, so you can switch between them just by changing the modules
; option at the top, rather than re-writing the whole file.
[camb]
file = boltzmann/camb/camb_all.so
function = execute
setup = setup

[wmap]
; The WMAP7 likelihood code
file = likelihood/wmap7/wmap_interface.so
function = execute

[dodelson_shear]
;Scott's code to generate C_ells in 5 redshift bins
file = shear/dodelson_shear/dodelson_cl_multibin.py
function = execute

[dodelson_like]
;Scott's code to get likelihoods of C_ell theory predictions for 5 redshift bins
;Based on BCC
--More--(66%)
```

Interfaces: Need to be able to pass inputs/outputs easily from one module/user to another

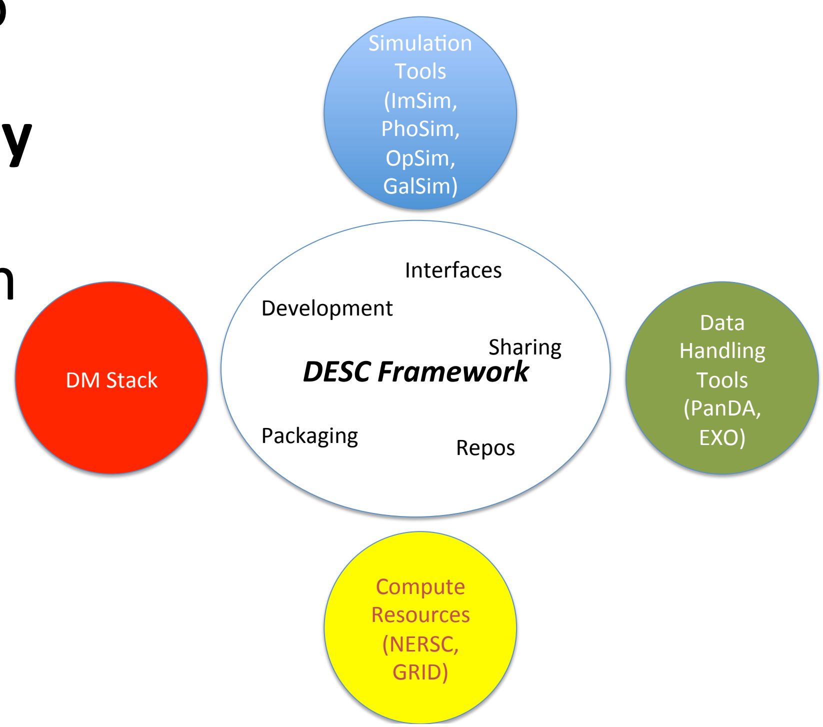




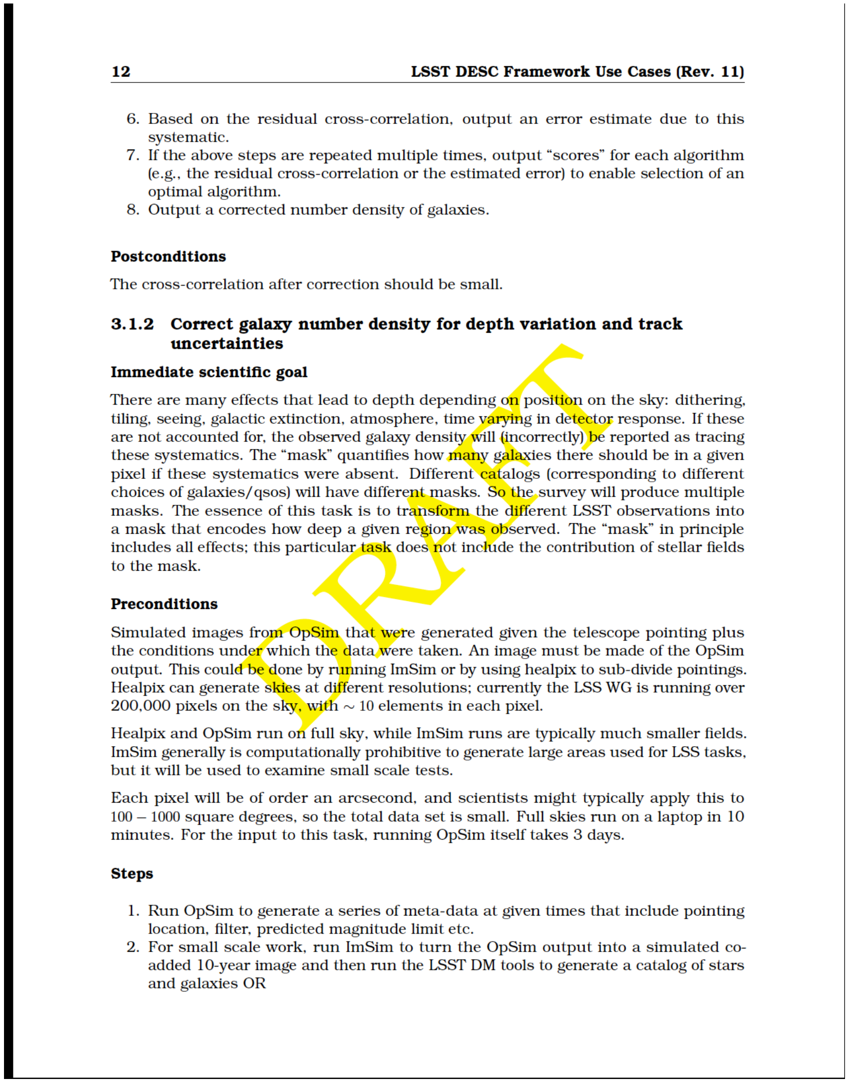
Breaking out v1.0 in mid-May

All of this will be much harder for LSST

- Not simply a parameter estimation toolkit: we need to understand **what the Science WGs want to do and how they do them**
- Incorporate project simulation tools
- Incorporate aspects of DM
- Leverage data handling tools developed for other experiments
- Use Compute resources (Grid, NERSC) in least painful way



Example: Use Case for Large Scale Structure



6. Based on the residual cross-correlation, output an error estimate due to this systematic.
7. If the above steps are repeated multiple times, output "scores" for each algorithm (e.g., the residual cross-correlation or the estimated error) to enable selection of an optimal algorithm.
8. Output a corrected number density of galaxies.

Postconditions

The cross-correlation after correction should be small.

3.1.2 Correct galaxy number density for depth variation and track uncertainties

Immediate scientific goal

There are many effects that lead to depth depending on position on the sky: dithering, tiling, seeing, galactic extinction, atmosphere, time varying in detector response. If these are not accounted for, the observed galaxy density will (incorrectly) be reported as tracing these systematics. The "mask" quantifies how many galaxies there should be in a given pixel if these systematics were absent. Different catalogs (corresponding to different choices of galaxies/qsos) will have different masks. So the survey will produce multiple masks. The essence of this task is to transform the different LSST observations into a mask that encodes how deep a given region was observed. The "mask" in principle includes all effects; this particular task does not include the contribution of stellar fields to the mask.

Preconditions

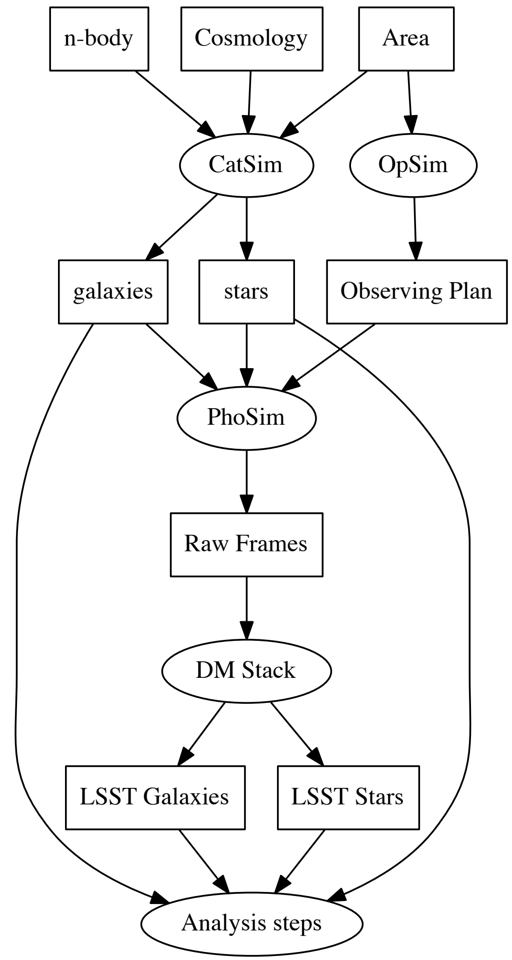
Simulated images from OpSim that were generated given the telescope pointing plus the conditions under which the data were taken. An image must be made of the OpSim output. This could be done by running ImSim or by using healpix to sub-divide pointings. Healpix can generate skies at different resolutions; currently the LSS WG is running over 200,000 pixels on the sky, with ~ 10 elements in each pixel.

Healpix and OpSim run on full sky, while ImSim runs are typically much smaller fields. ImSim generally is computationally prohibitive to generate large areas used for LSS tasks, but it will be used to examine small scale tests.

Each pixel will be of order an arcsecond, and scientists might typically apply this to 100 – 1000 square degrees, so the total data set is small. Full skies run on a laptop in 10 minutes. For the input to this task, running OpSim itself takes 3 days.

Steps

1. Run OpSim to generate a series of meta-data at given times that include pointing location, filter, predicted magnitude limit etc.
2. For small scale work, run ImSim to turn the OpSim output into a simulated co-added 10-year image and then run the LSST DM tools to generate a catalog of stars and galaxies OR



Walk through this tomorrow

3. For large scale work, use Healpix, source count laws, and a Milky Way model to generate a map of simulated star and galaxy counts directly from the output of OpSim
4. Using the information generated in the first 3 steps, estimate the mask at each pixel via the ratio $N_{\mathrm{input}}/N_{\mathrm{found}}$.
5. Correct the galaxy density using the mask
6. Cross-correlate the resulting galaxy density with observables such as the stellar density, the mask itself, or other sources with which it should not be correlated.
7. Using these cross-correlation functions, improve the estimate of the mask and then repeat Steps 5 and 6.
8. Evaluate the residual cross-correlation in the corrected image.
9. Based on the residual cross-correlation, obtain an error estimate on the mask.
10. Output a mask and its uncertainty at all pixels.
11. Other outputs include: a corrected map of the galaxy density; an angular correlation function of the galaxies, both corrected and un-corrected; and the range of scales that is unpolluted by uncertainties in the mask.

3.1.3 Inject fake sources into data and rerun DM

Immediate Scientific Goal

Real data (*e.g.* from SDSS) added to fake sources to see if the DM recovers the sources in the presence of realistic fields. The initial goal is to obtain completeness (probability that a source is recovered) as a function of magnitude and size of the object. This task