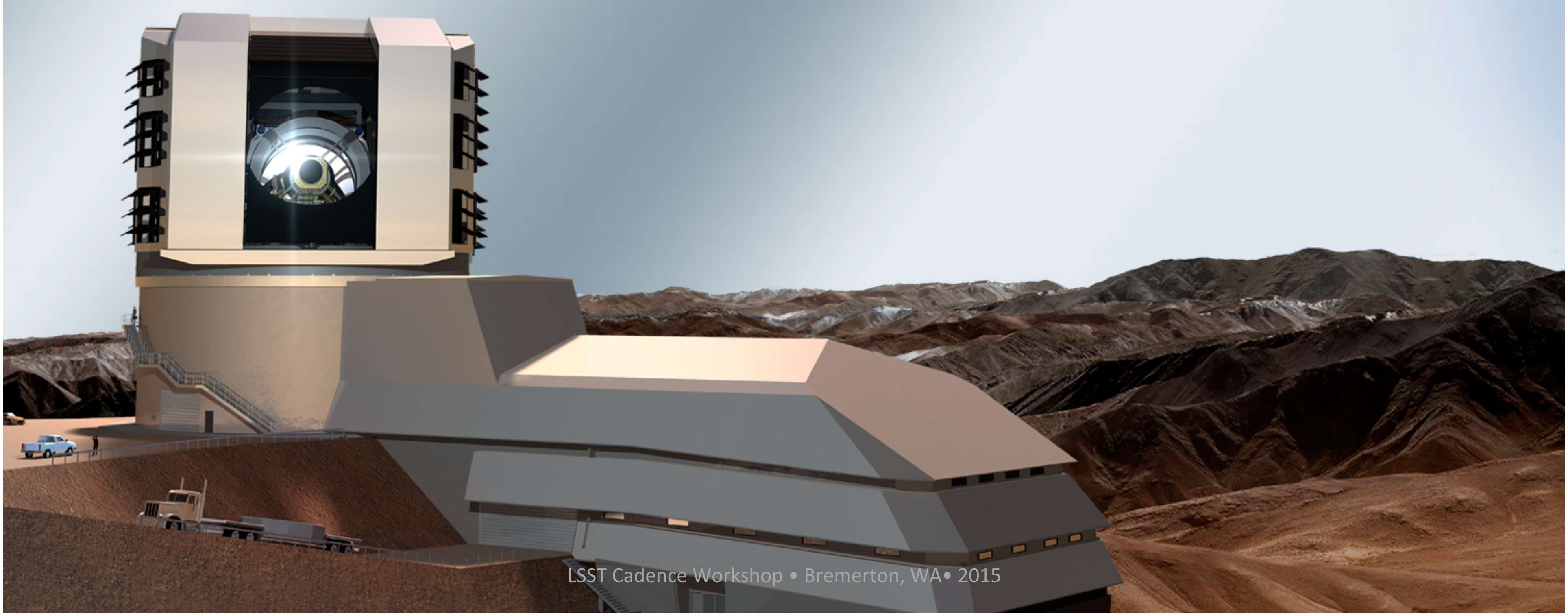
Metric Analysis Framework

Peter Yoachim
University of Washington



What we use MAF for

- Did the scheduler do reasonable things?
- Did we meet the quantified Science Requirements Document (SRD) requirements?
- How well can we do different sciences with the survey?

All Python. Tried to make it easy for the community to extend MAF. Write your own metric and push the results into our git repo!

We can't do it all

- While we have all the tools to simulate LSST end-to-end, that's pretty computationally expensive

CatSim + OpSim -> PhoSim -> DM => observed catalog

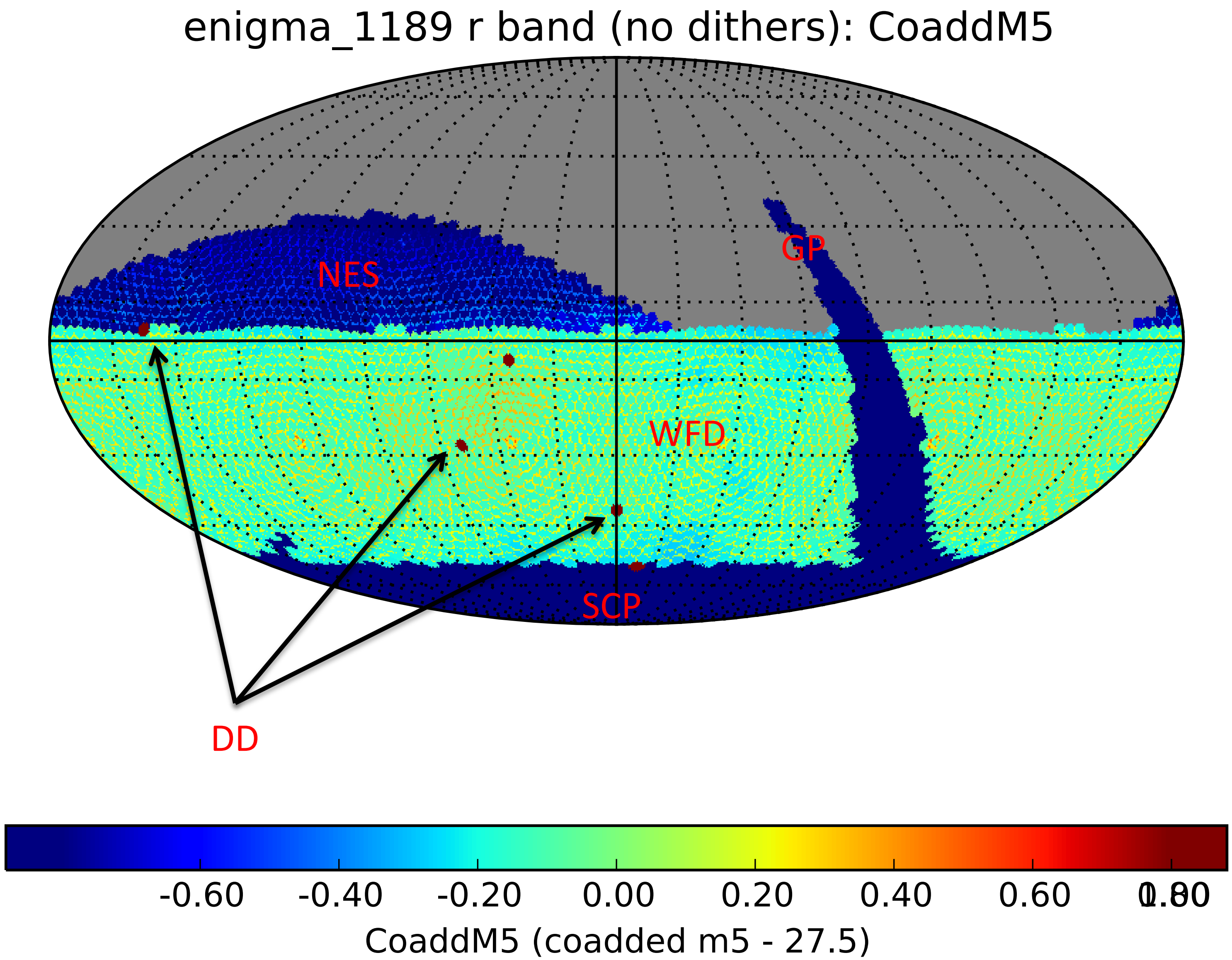
- For a 10 year LSST survey, we will observe 10^{16} pixels! We need to approximate a few of those components when doing the scheduling strategy optimization
  - Rather than use a catalog of objects, assume a simple distribution of sources (e.g., a uniform grid of galaxies)
  - Rather than use PhoSim+DM, just assume Gaussian noise or simple centroiding errors

Opsim + MAF -> what fraction of my galaxies are "well observed", how many SN Ia do I discover, how precise is the astrometric solution, etc.

The Operations Simulator

- OpSim is our source for what these 2.5 million visits might look like.
  - Includes real weather log
  - Scheduled and unscheduled down time
  - A scheduler that balances several science goals
- OpSim scheduler based on "Proposals"
  - Wide-Fast-Deep (aka, "the main survey"): Cover ~20,000 sq deg
  - North Ecliptic Spur: Solar system objects
  - Deep Drilling Fields: ~6 deep fields
  - Galactic Plane
  - South Celestial Pole

The Proposals



What's in OpSim Output

For each visit, Opsim records

RA,Dec
Filter
MJD
Night
visitTime
Seeing
Airmass
Skybightness
Rotation angle of the camera
LST
Alt,Az
Distance to moon
Distance to Sun
Moon position
Moon phase
5-sigma depth (so can calc SNR of an object)
Dithered RA,Dec
And more…

More documentation on OpSim Summary table here: http://ls.st/5d8

OpsSim Warnings

- OpSim records LSST "visits"
  - 1 visit = 2 back-to-back exposures!
- By default, OpSim uses fixed fields and does not dither
- There can be "repeated" entries in the Summary table if a visit is used for multiple proposals (e.g., a visit can count towards WFD and Deep Drilling).

In MAF, we take care of this by doing a SQL "group by" on the expMJD.

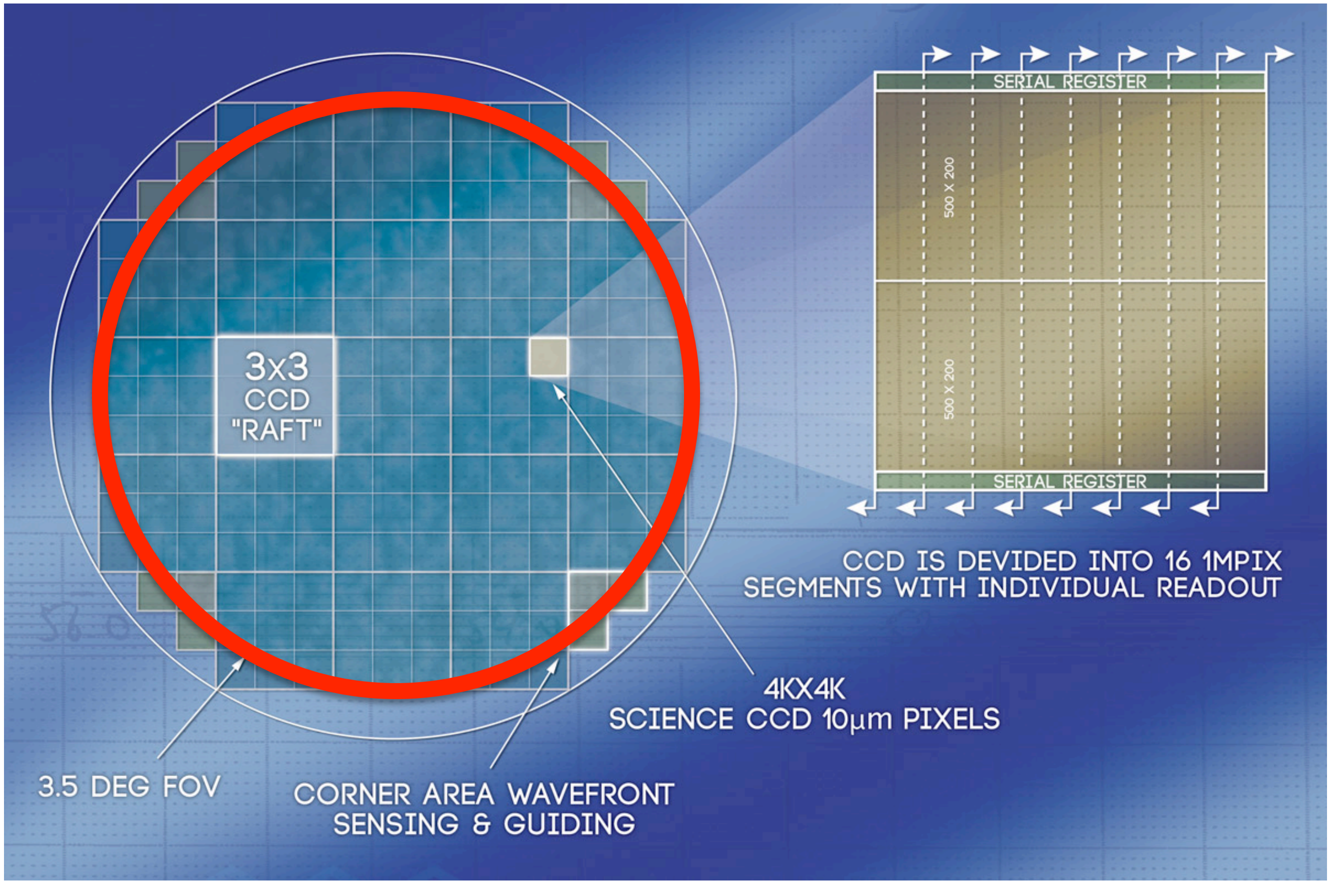
Observation ID	Proposal ID	MJD	RA	DEC
10	DD	3456.4	0:42:44	41:16:09
11	DD	3456.8	10:39:37	43:06:09
12	WFD	3456.8	10:39:37	43:06:09
13	WFD	3457.1	05:34:32	22:00:52

Not unique

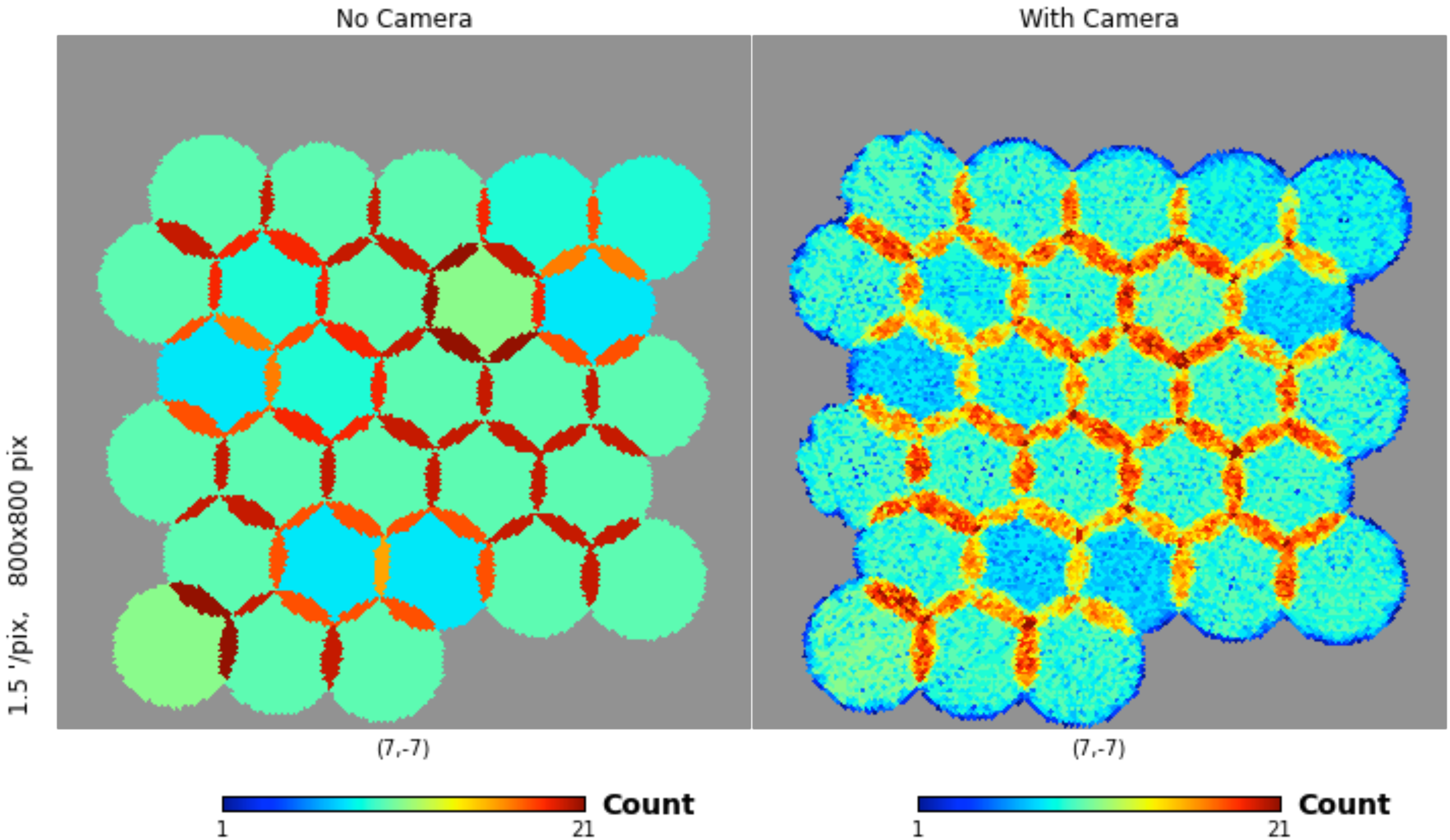
MAF Warnings

- By default, we assume if an object is within 1.75 degrees of the center, it gets observed.
- This is ~11% optimistic given chip gaps, etc.

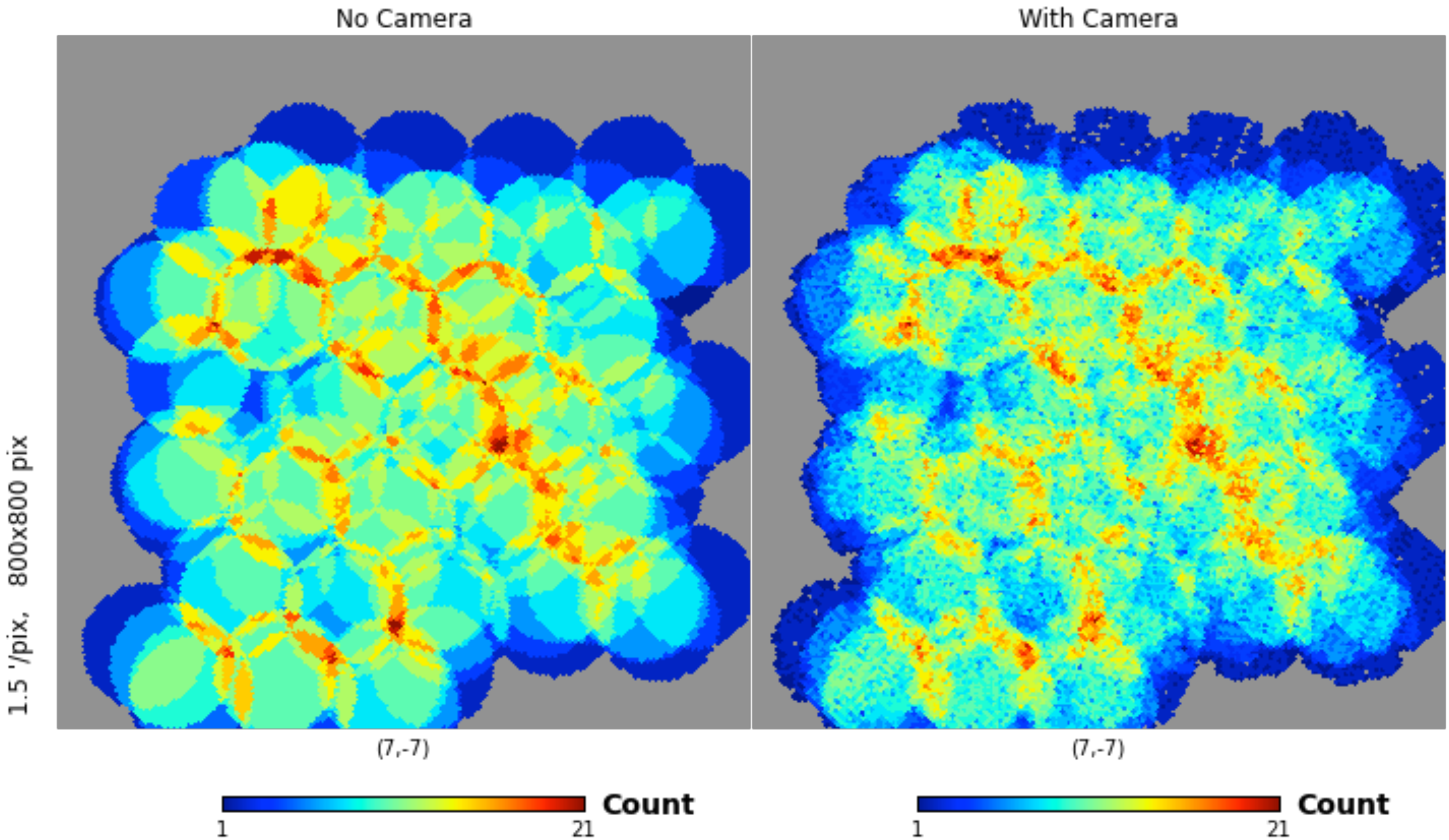
You can run with the full focal-plane geometry, but it slows things down a bit.



With and Without Chip Gaps

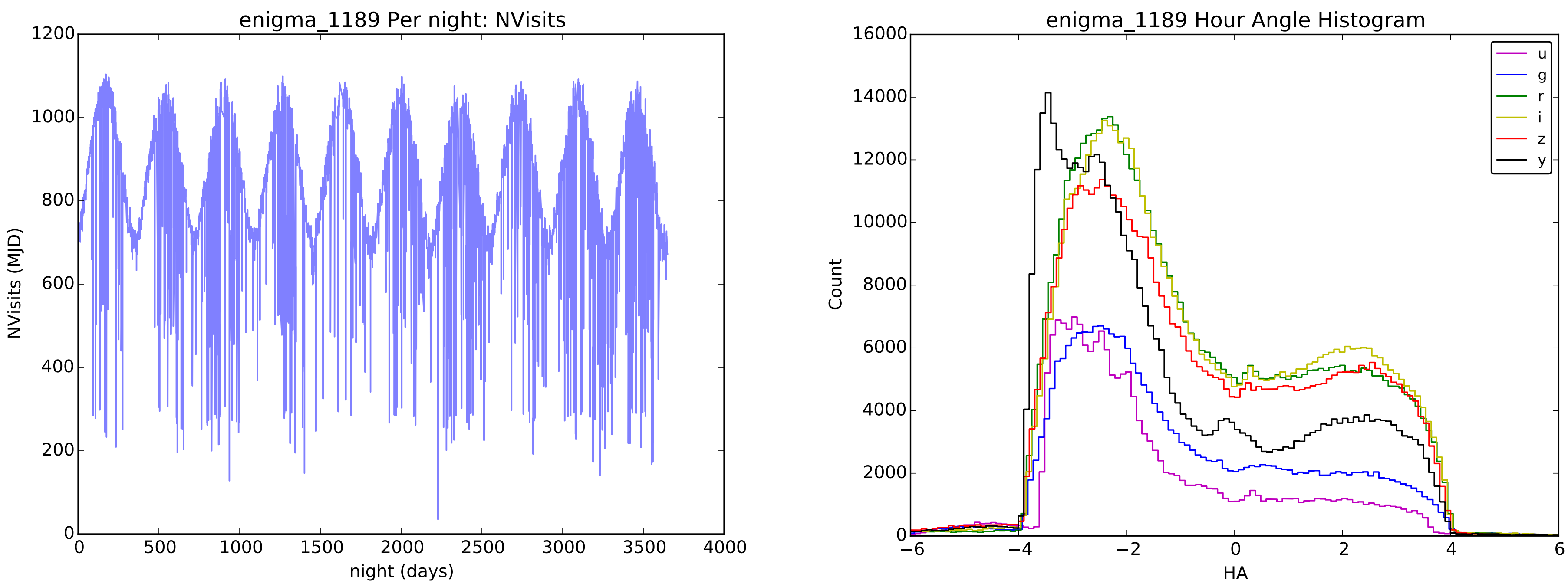


And now with dithering



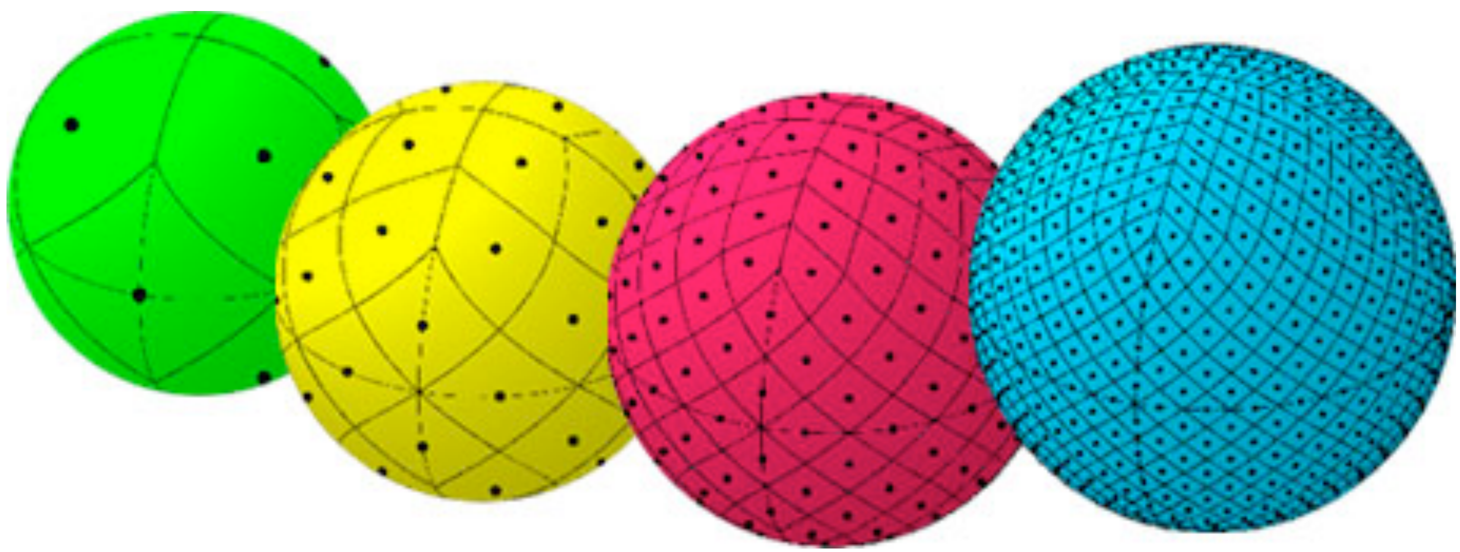
Simple Stuff

- MAF can computes means, medians, etc of columns. Also simple histograms.
- MAF tries to build reasonable titles and axis labels.



HEALpixels

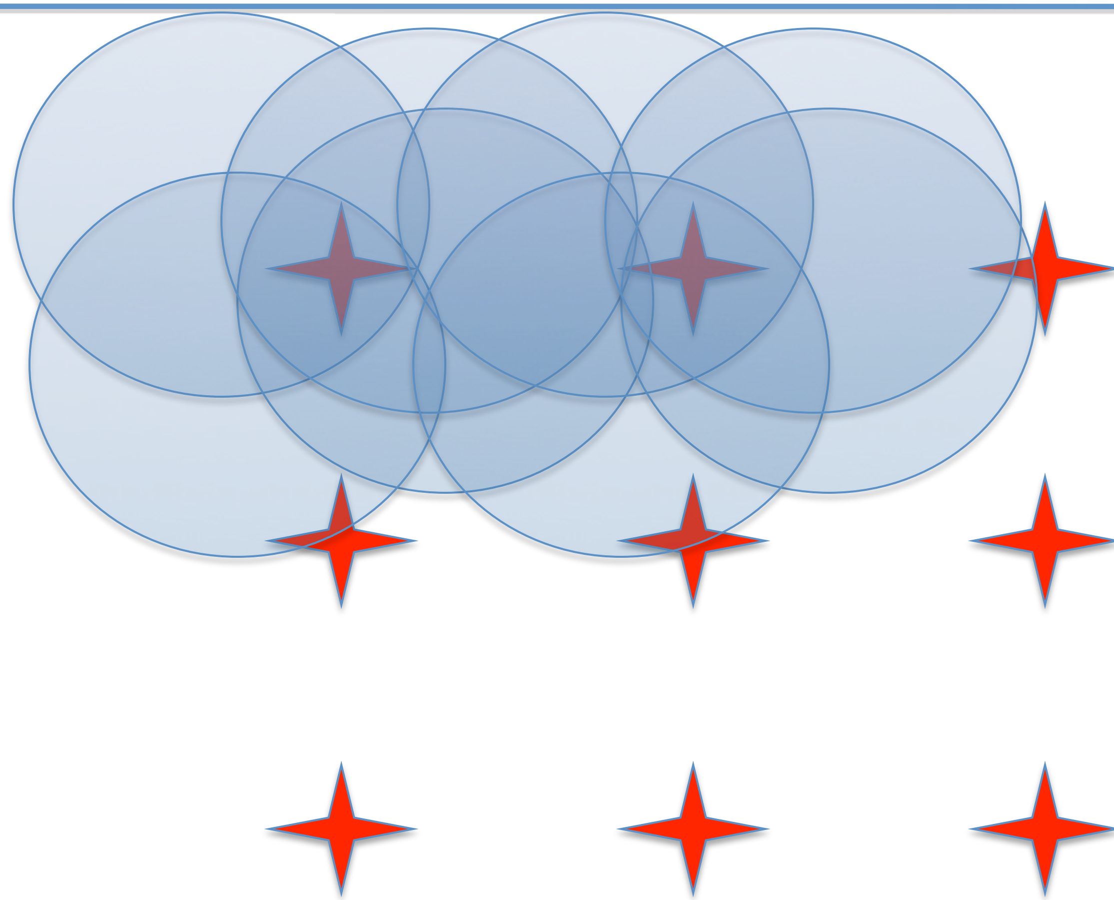
- For our spatial binning, we use HEALpixels
  - Equal area pixels
  - Popular with cosmologists, rapidly compute power spectra
  - Or, use a custom list of RA,Dec points
  - Can also bin by OpSim field ID. Faster, but field overlaps are not resolved



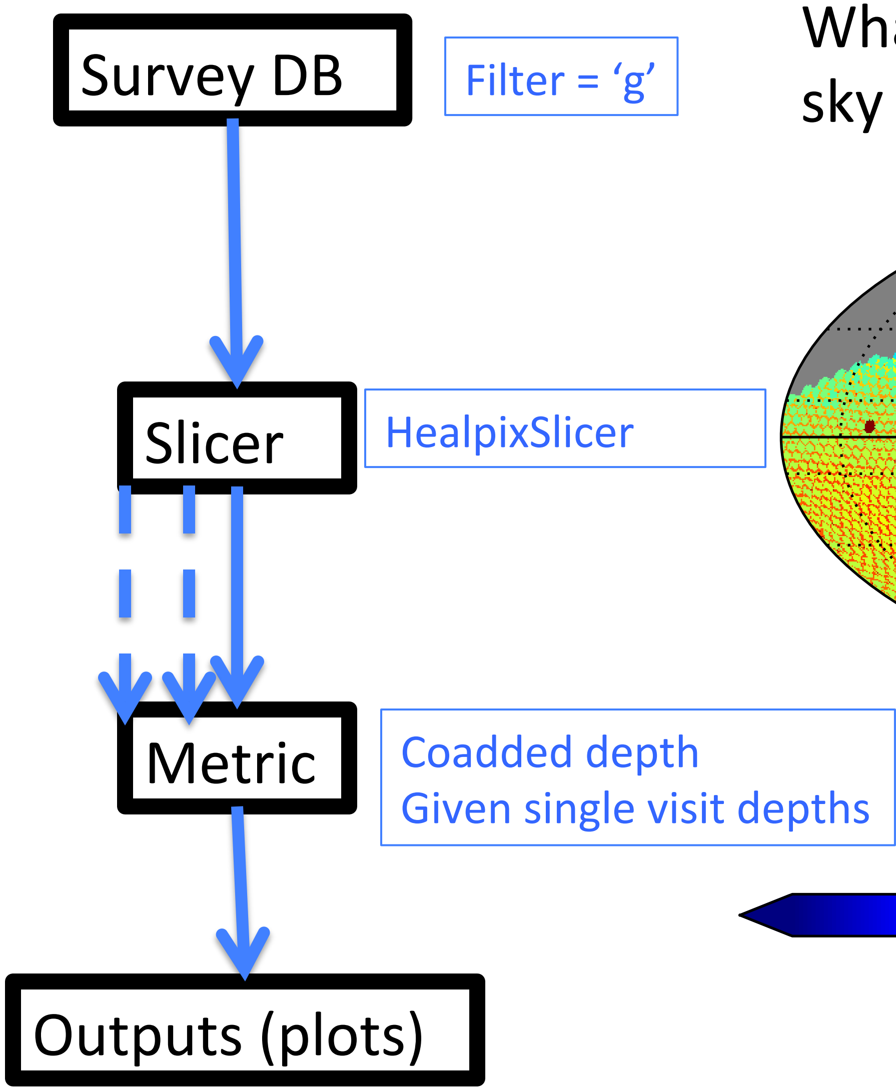
One way to think about the Healpix grid

Which observations overlap star/galaxy/healpixel #1?
Pass those to the metric algorithm and save the result

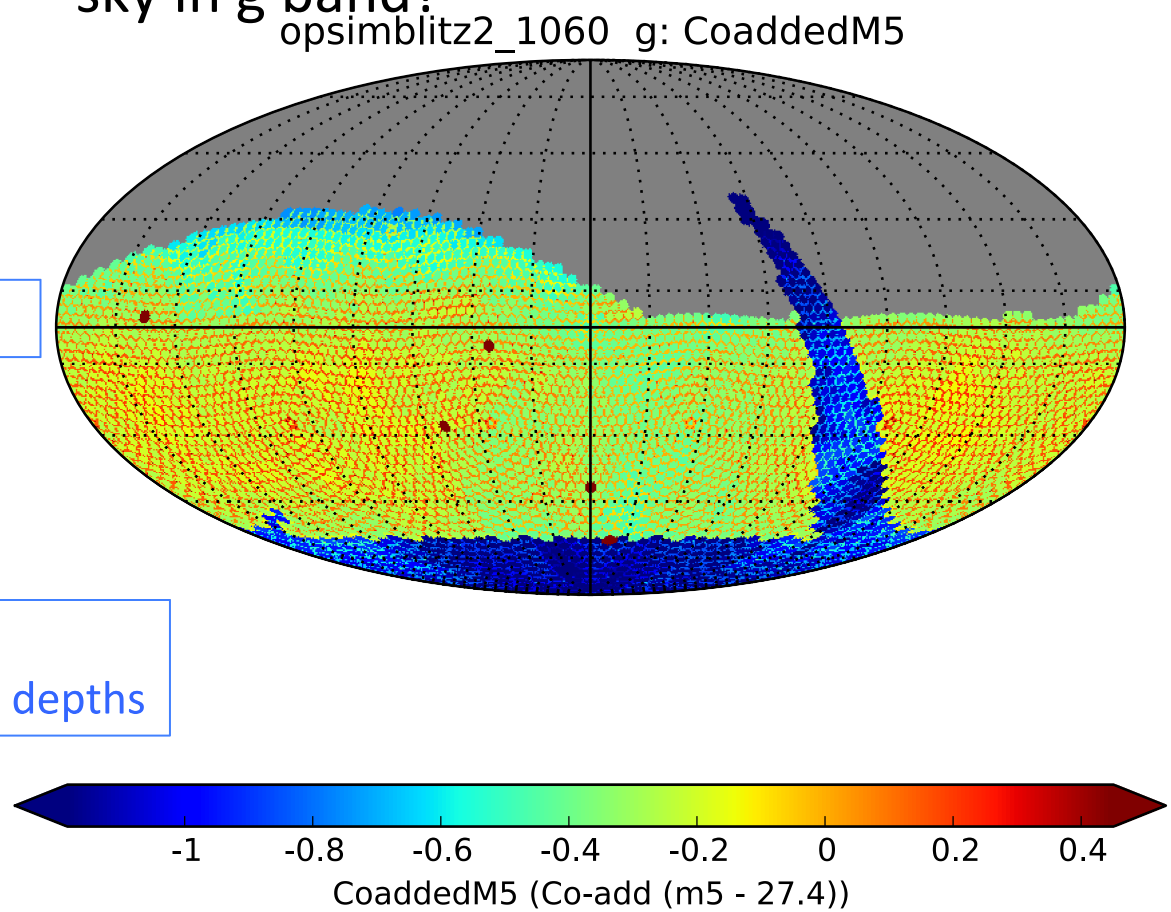
Which observations overlap star/galaxy/healpixel #2?
Pass those to the metric algorithm and save the result.

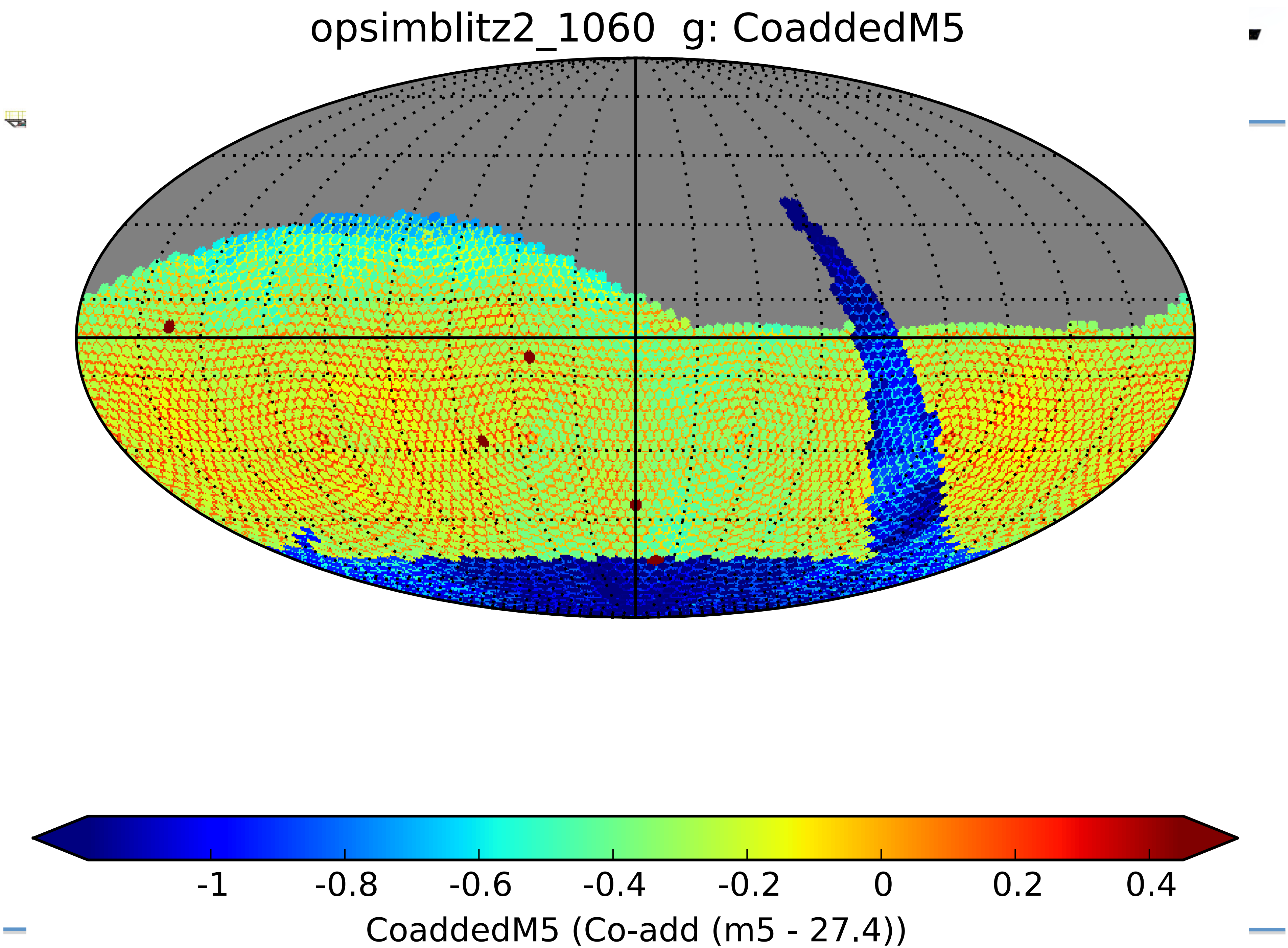


MAF Examples



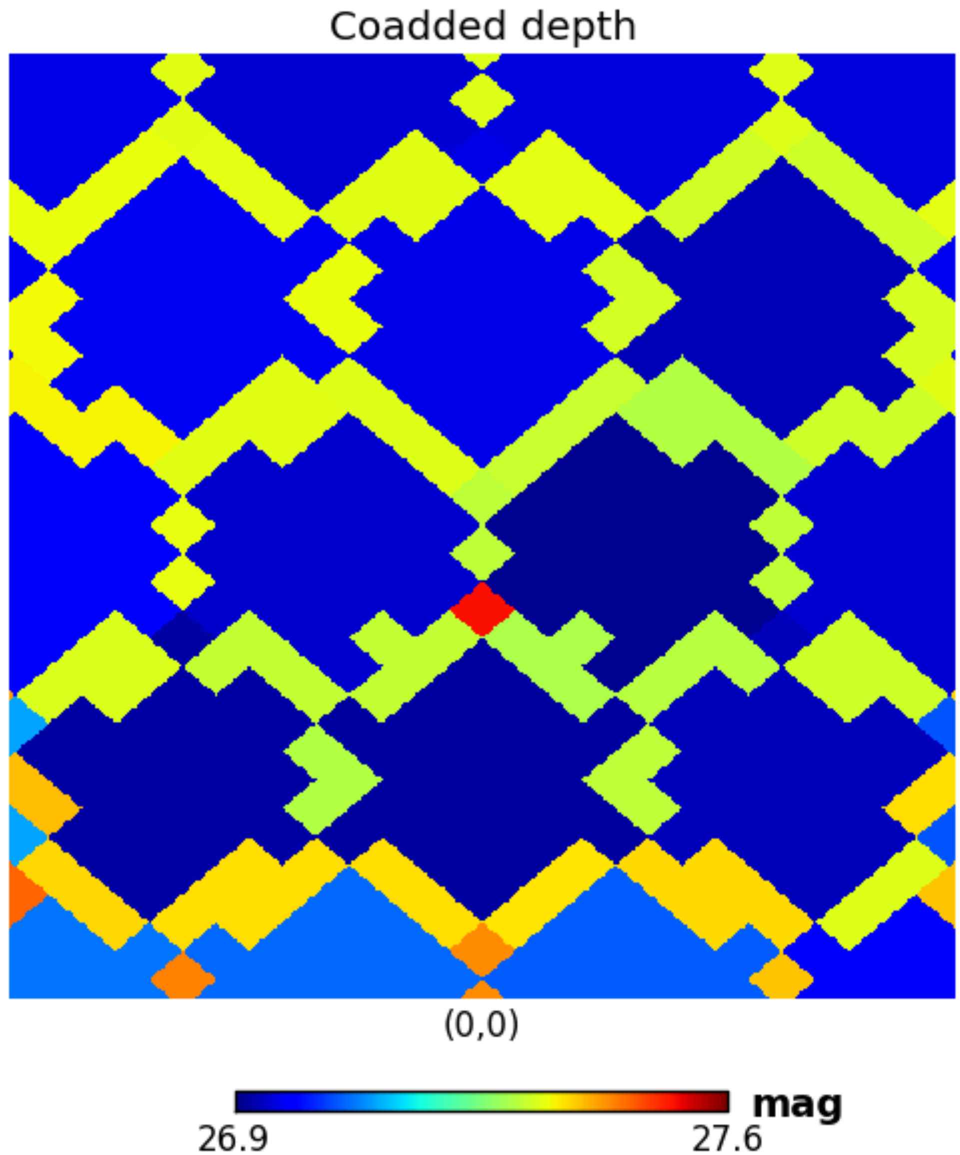
What is the coadded depth across the sky in g band?



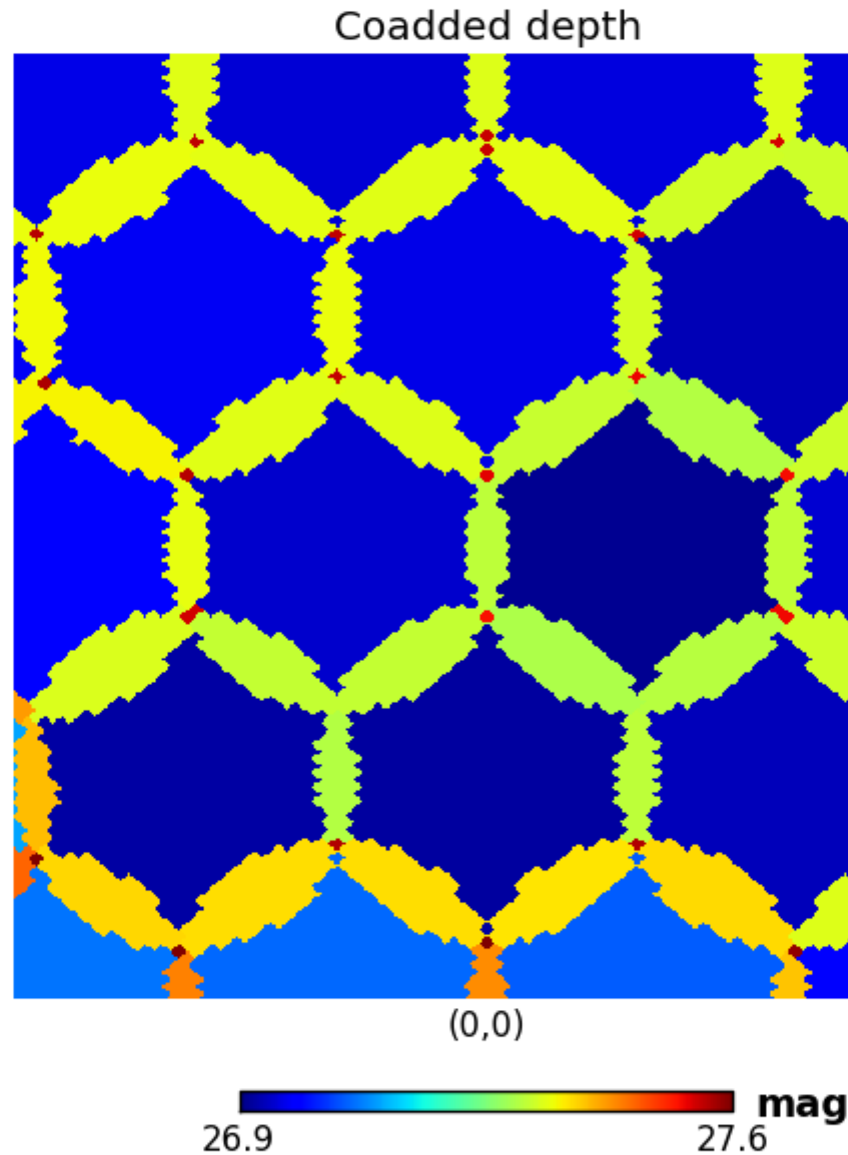


opsimblitz2_1060 g: CoaddedM5
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
CoaddedM5 (Co-add (m5 - 27.4))

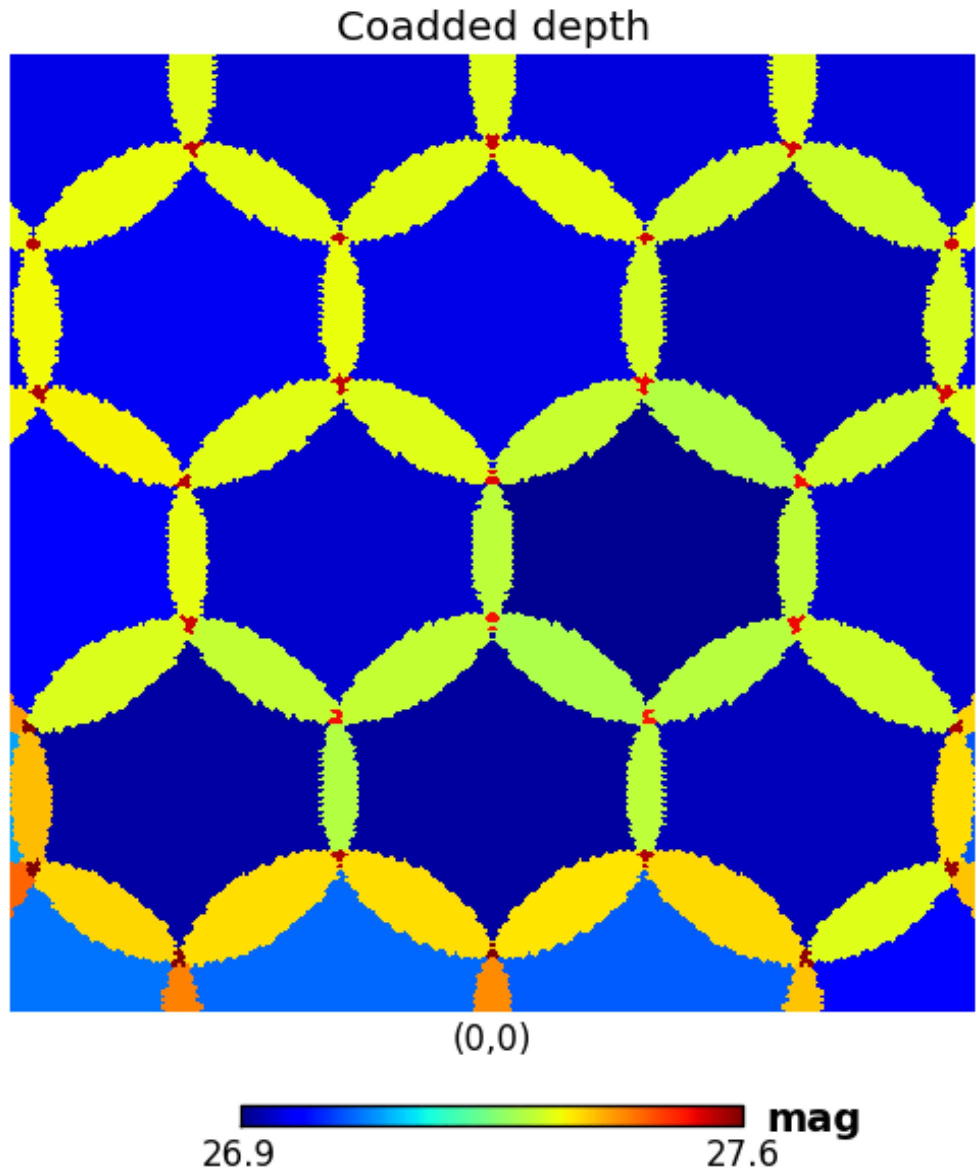
nside=128, 27.5 arcmin resolution



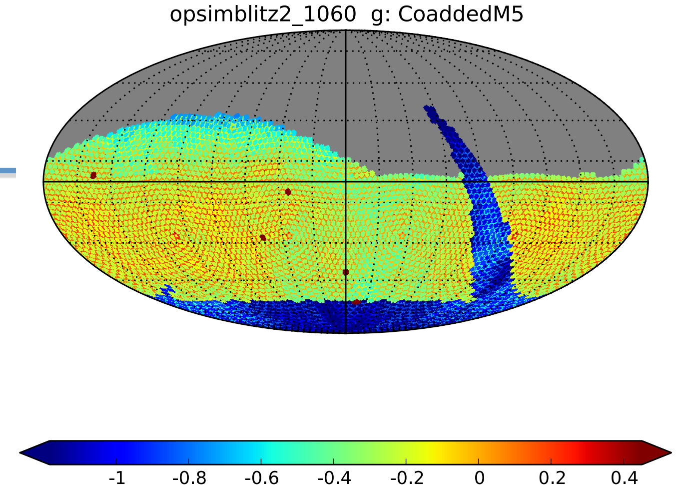
Nside=512, 6.9 arcmin resolution

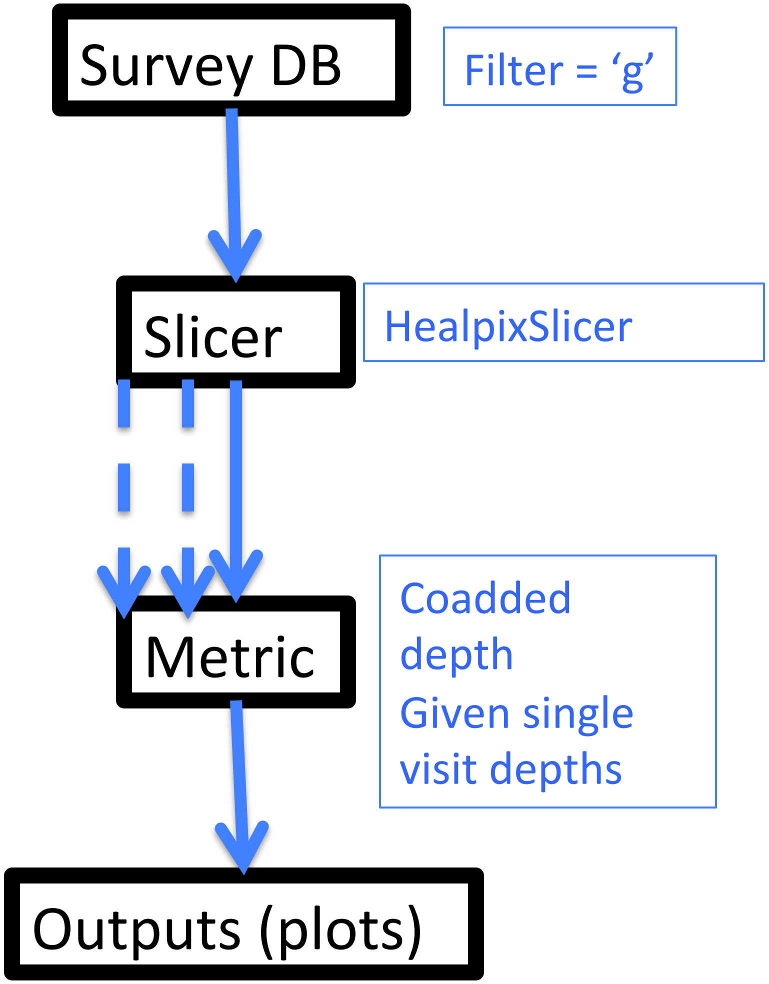


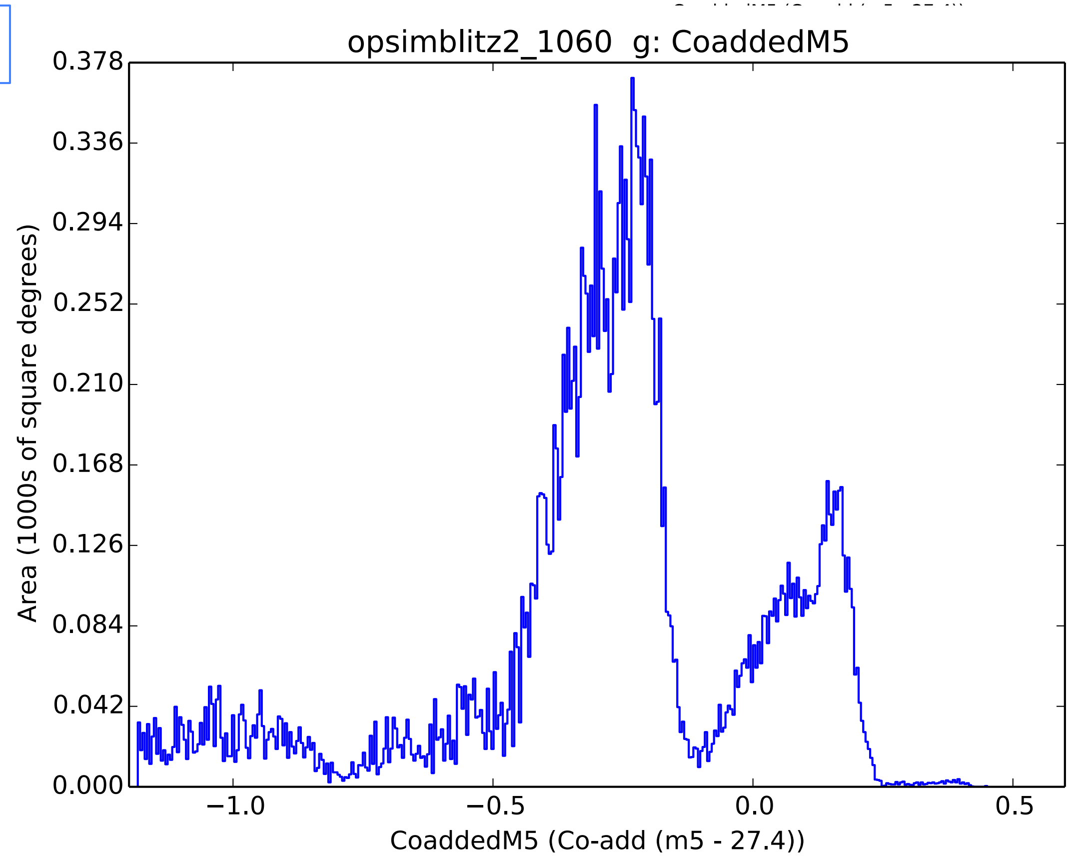
Nside=1024, 3.4 arcmin resolution



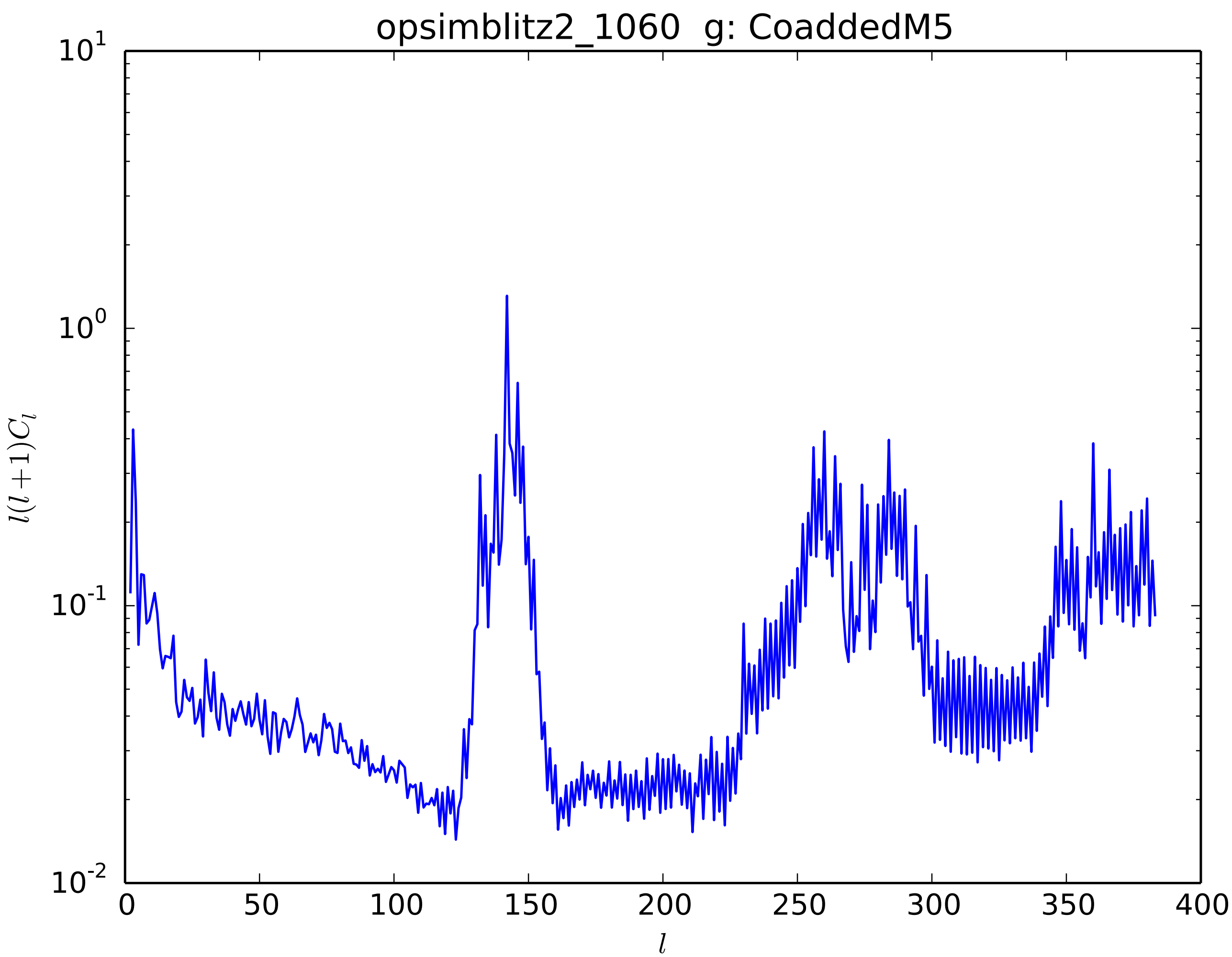
MAF Examples





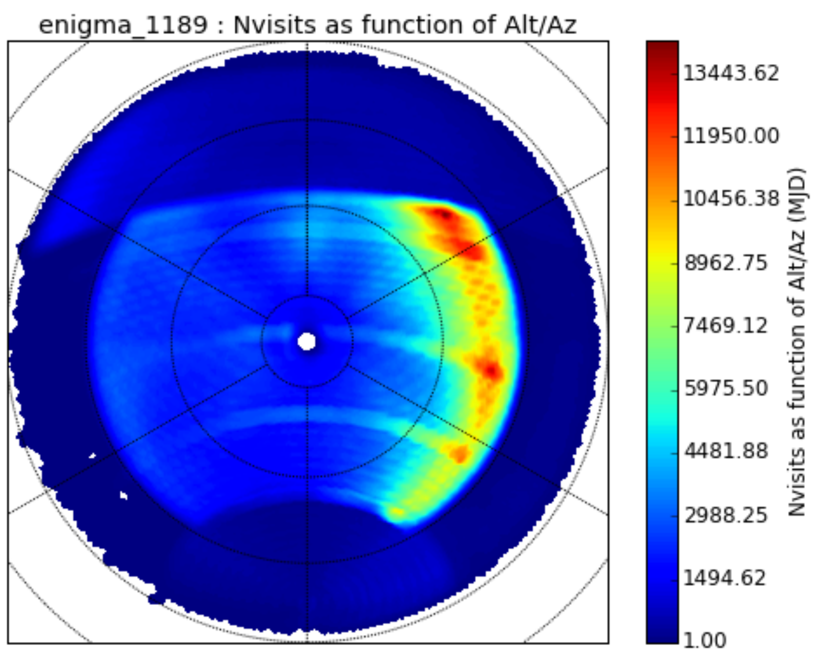


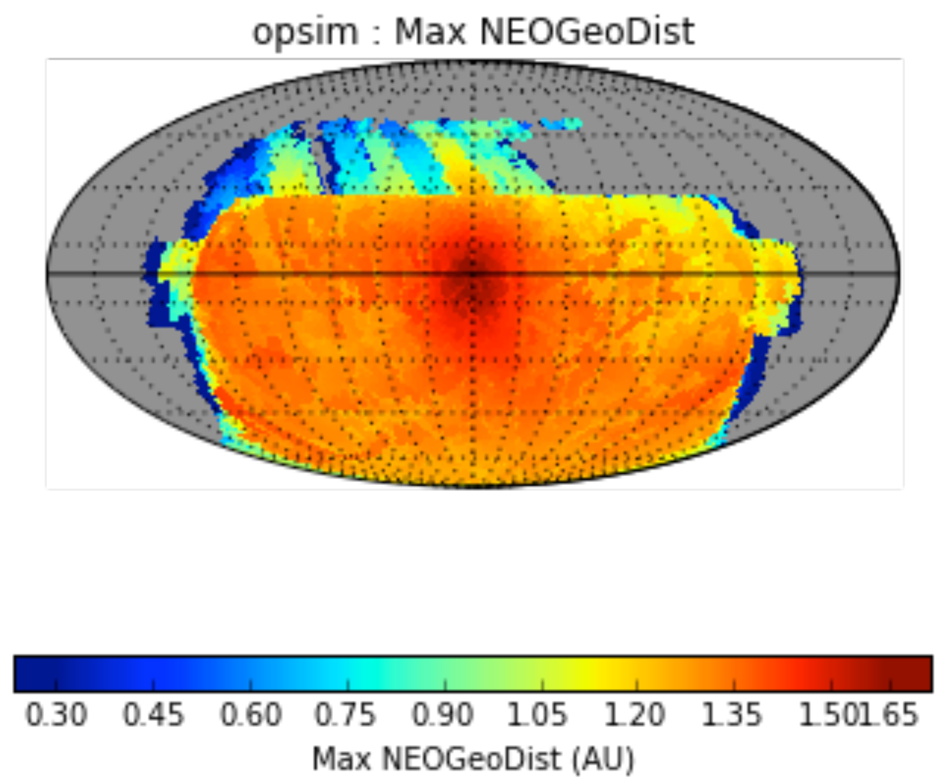
Healpixels let us compute Power Spectra



Peak at l=140 (~1 degree), caused by field overlaps Peak at l=270 from triple overlaps.

- The HealpixSlicer defaults to RA,Dec, but can be set to use Alt,Az or ecliptic coordinates.

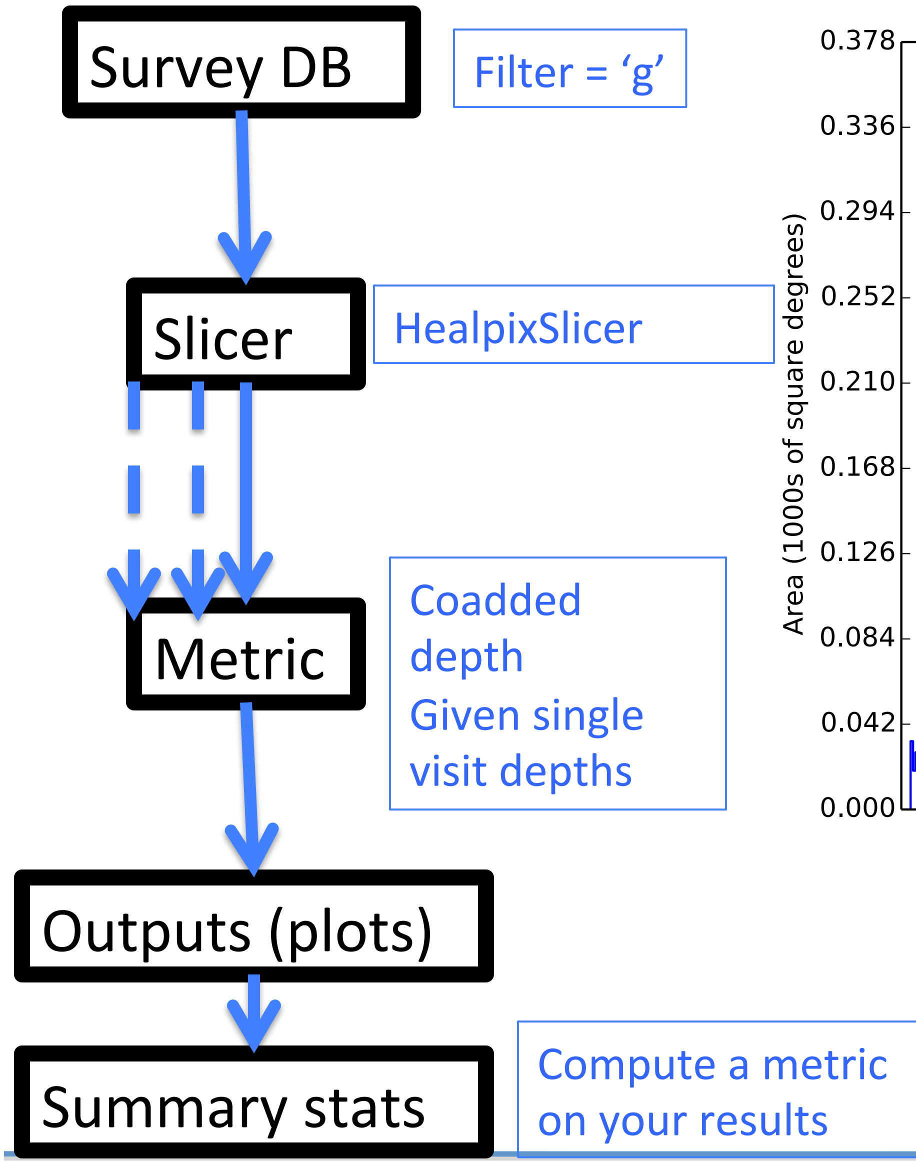


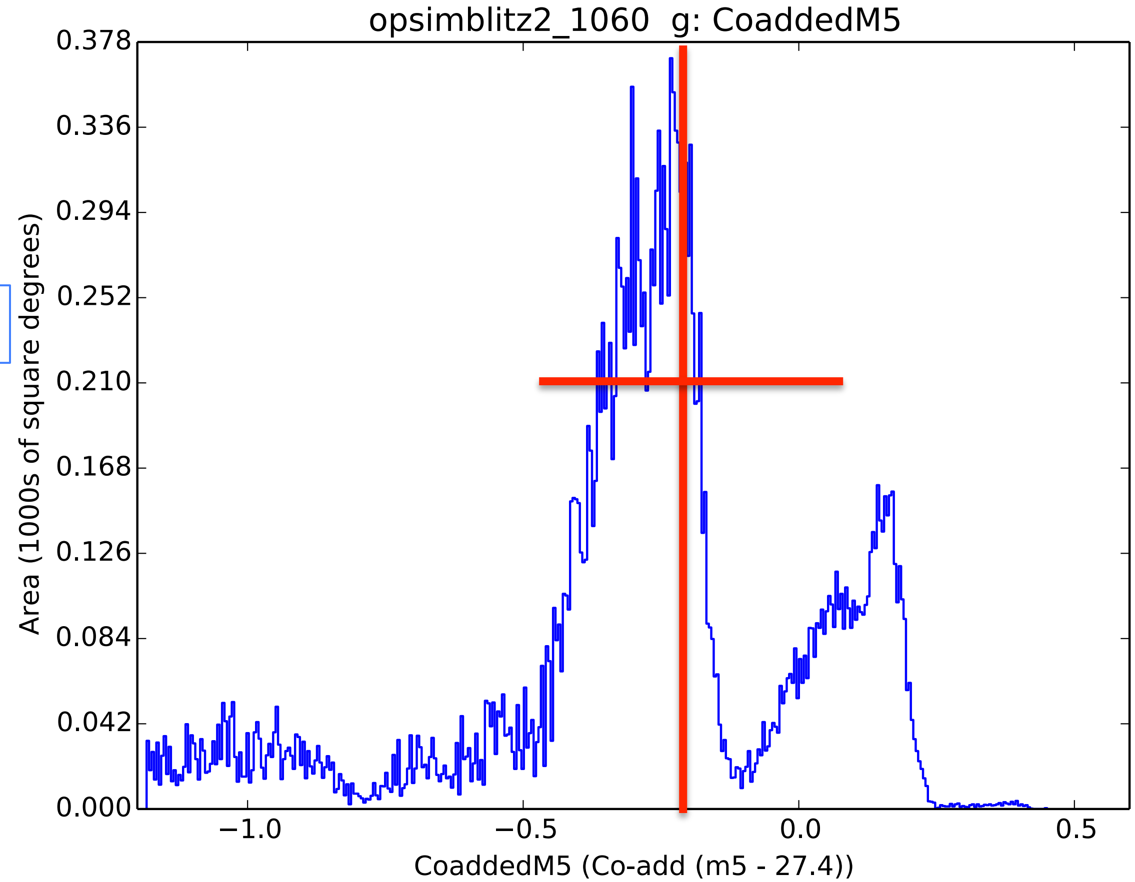


Additional Capabilities

- Summary Statistics
- Stackers: Adding derived columns

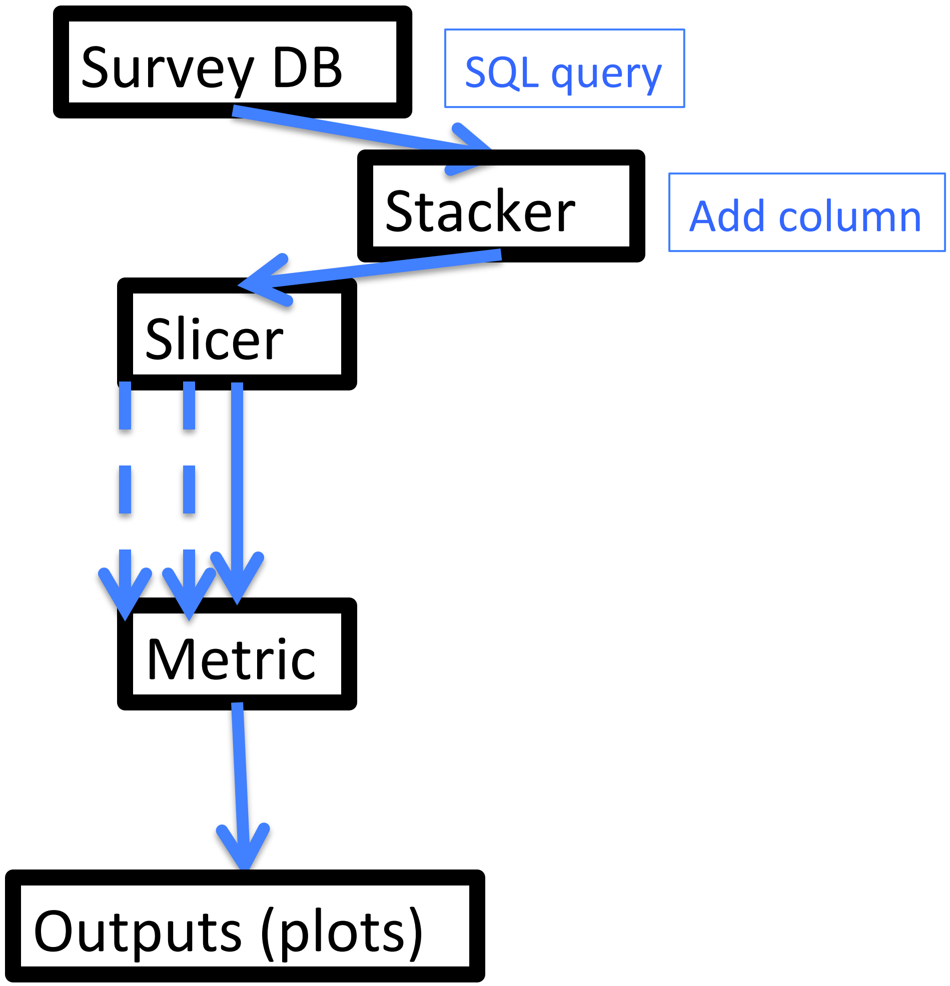
Summary Stats on a HealpixSlicer





e.g.,
Mean and RMS of metric values

Add a pseudo-column



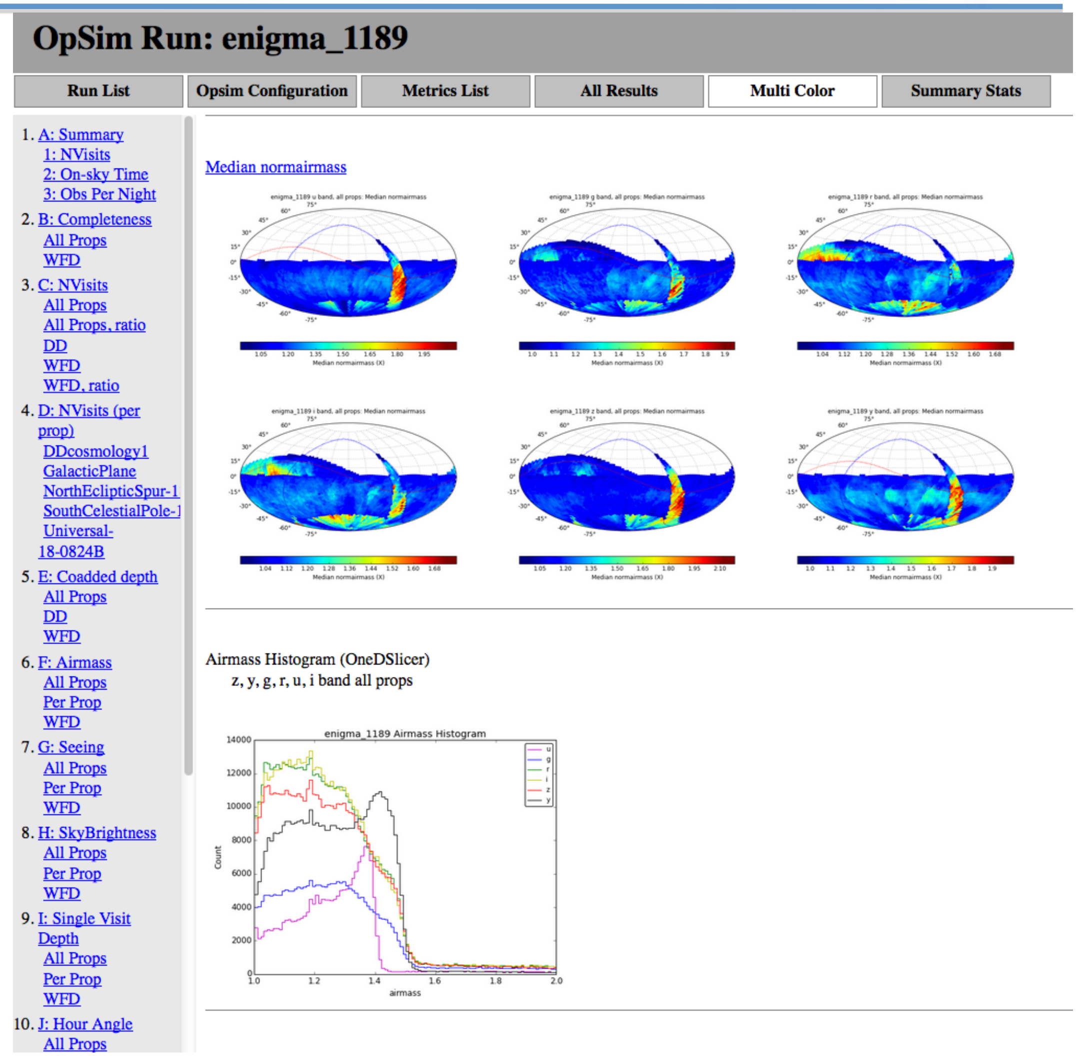
Sometimes it's convenient to calculate a new column that isn't present in Opsim before slicing the data.

For example, normalized airmass, parallax factor, dither offset, hour angle.

We have "stackers" that allow you to stack new columns onto the SQL results before they are passed to the slicer. (we're calling them stackers since we are using the numpy stack functions to add columns to numpy arrays)

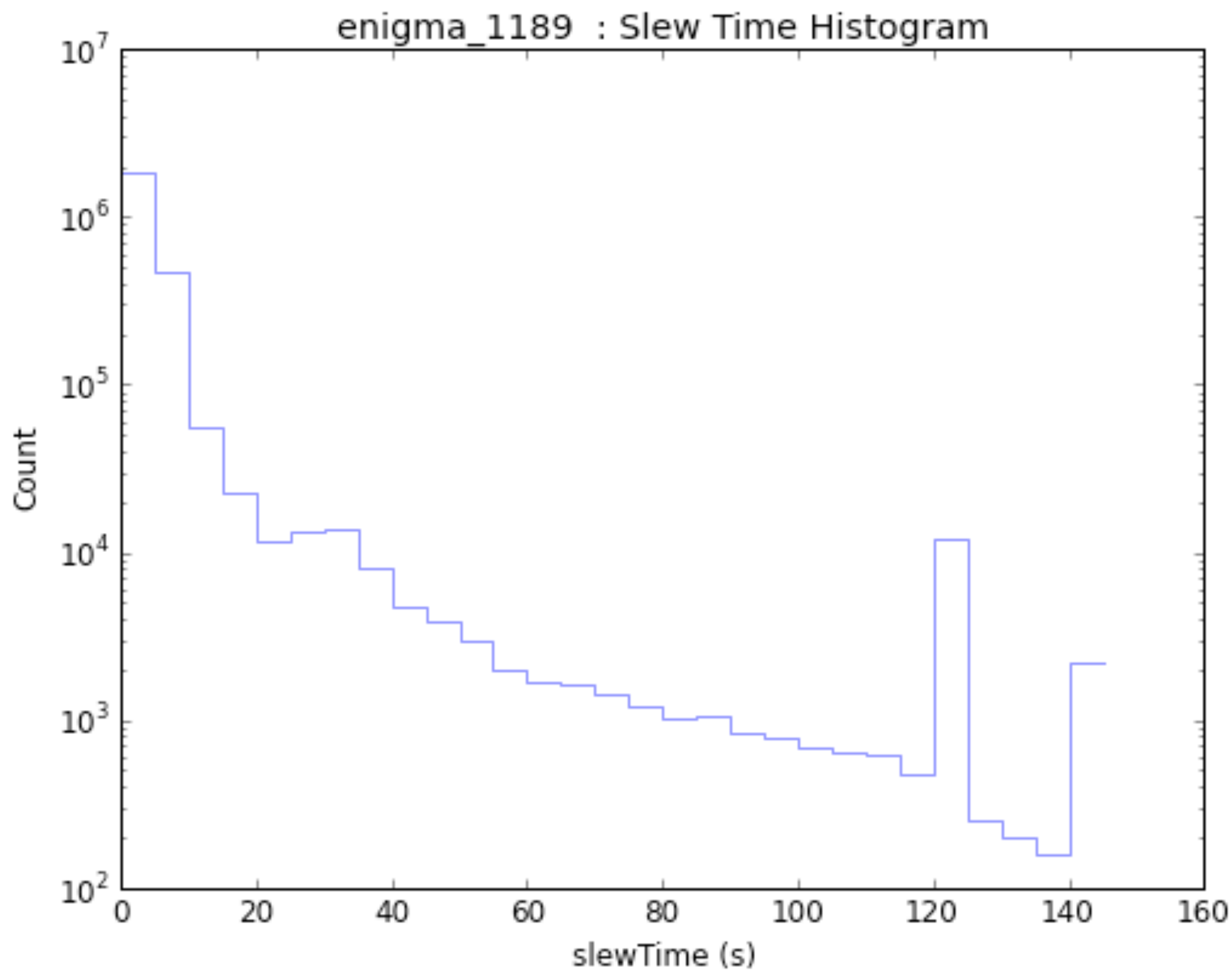
Scheduler Validation

- Checks things like number of visits, airmass distribution, coadded depth.
- MAF includes a semi-intelligent web display (u,g,r,i,z,y order, etc)



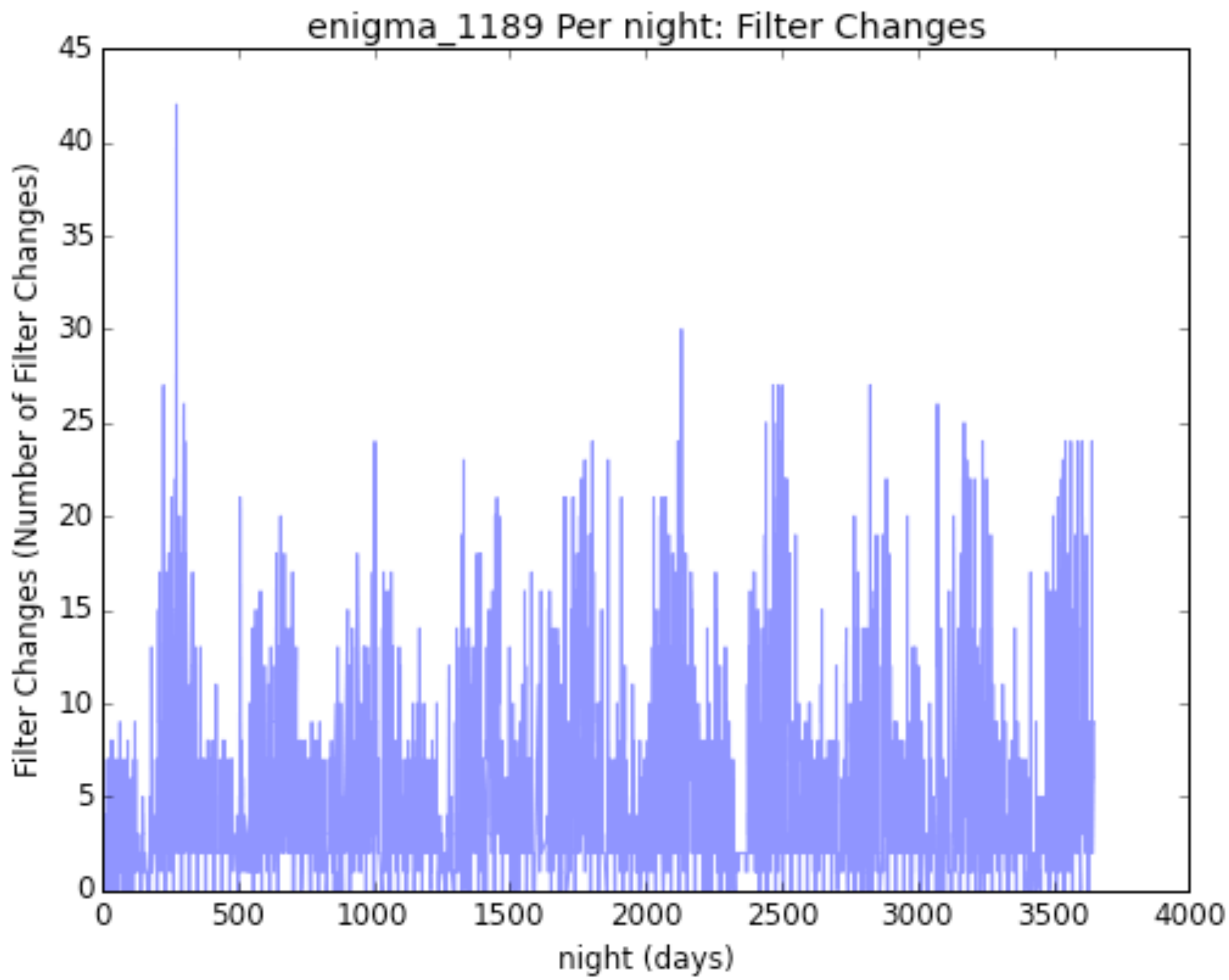
Scheduler Validation, Examples

Slew distribution (most slews are short, small peak for filter changes).

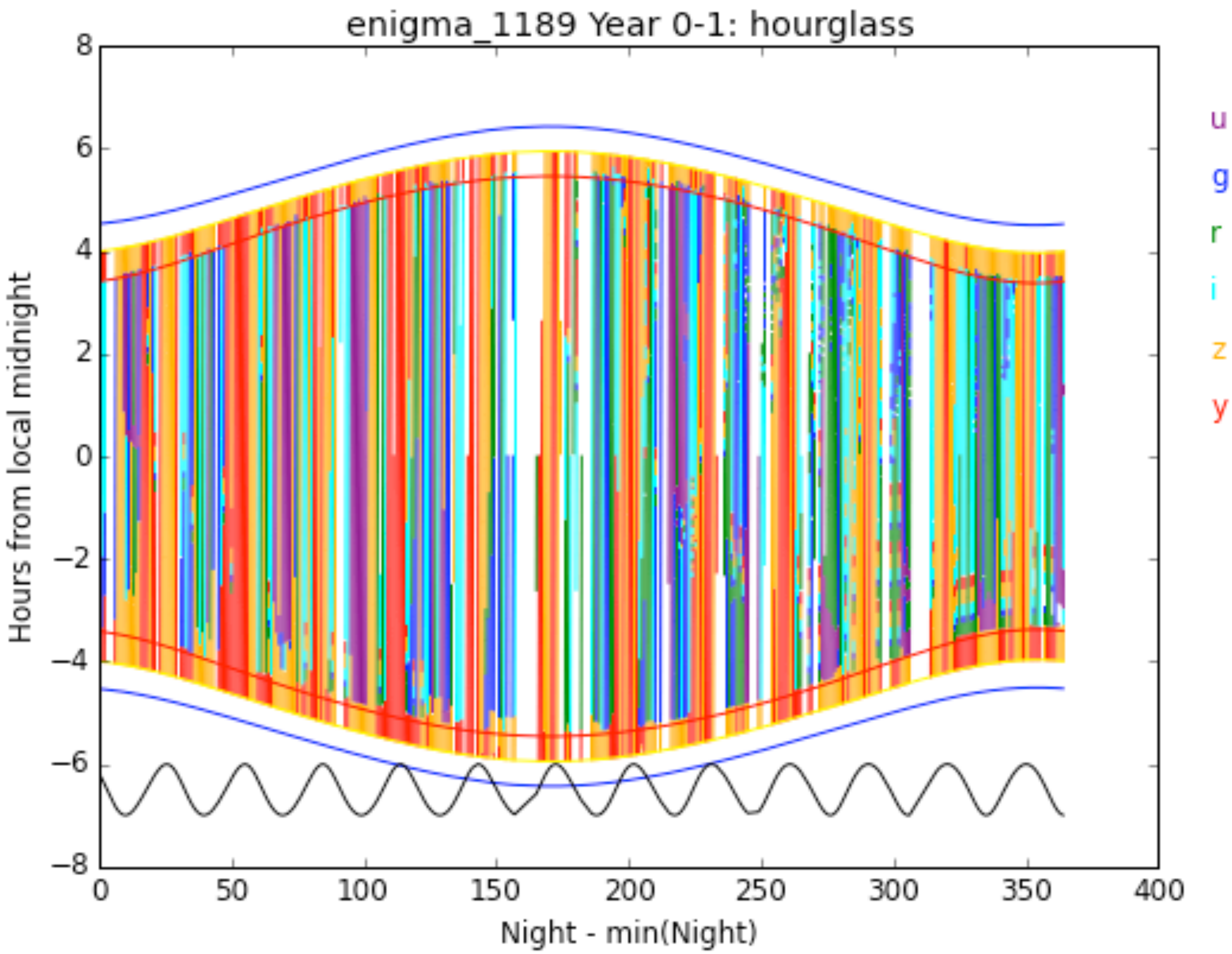


Scheduler Validation, Examples

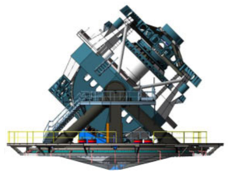
Filter changes per night



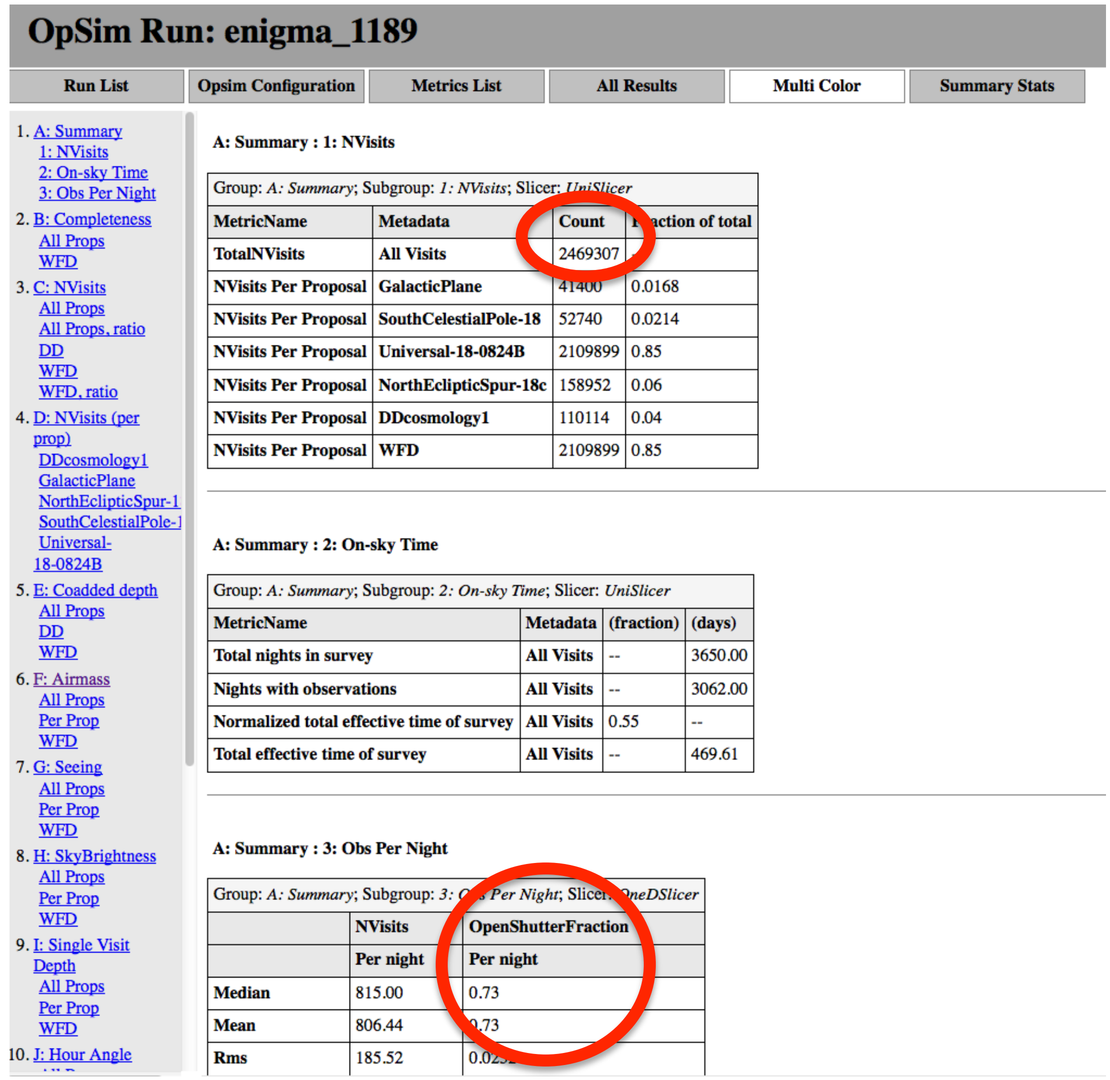
Scheduler Validation, Examples



Scheduler Validation, Examples



2.4 million visits



OpSim Run: enigma_1189

Run List | Opsim Configuration | Metrics List | All Results | Multi Color | Summary Stats

1. A: Summary
   - 1: NVisits
   - 2: On-sky Time
   - 3: Obs Per Night
2. B: Completeness
   - All Props
   - WFD
3. C: NVisits
   - All Props
   - All Props, ratio
   - DD
   - WFD
   - WFD, ratio
4. D: NVisits (per prop)
   - DDcosmology1
   - GalacticPlane
   - NorthEclipticSpur-1
   - SouthCelestialPole-1
   - Universal-18-0824B
5. E: Coadded depth
   - All Props
   - DD
   - WFD
6. F: Airmass
   - All Props
   - Per Prop
   - WFD
7. G: Seeing
   - All Props
   - Per Prop
   - WFD
8. H: SkyBrightness
   - All Props
   - Per Prop
   - WFD
9. I: Single Visit Depth
   - All Props
   - Per Prop
   - WFD
10. J: Hour Angle

A: Summary : 1: NVisits

Group: *A: Summary*; Subgroup: *1: NVisits*; Slicer: *UniSlicer*

MetricName	Metadata	Count	Fraction of total
TotalNVisits	All Visits	2469307	[illegible]
NVisits Per Proposal	GalacticPlane	41400	0.0168
NVisits Per Proposal	SouthCelestialPole-18	52740	0.0214
NVisits Per Proposal	Universal-18-0824B	2109899	0.85
NVisits Per Proposal	NorthEclipticSpur-18c	158952	0.06
NVisits Per Proposal	DDcosmology1	110114	0.04
NVisits Per Proposal	WFD	2109899	0.85

A: Summary : 2: On-sky Time

Group: *A: Summary*; Subgroup: *2: On-sky Time*; Slicer: *UniSlicer*

MetricName	Metadata	(fraction)	(days)
Total nights in survey	All Visits	--	3650.00
Nights with observations	All Visits	--	3062.00
Normalized total effective time of survey	All Visits	0.55	--
Total effective time of survey	All Visits	--	469.61

A: Summary : 3: Obs Per Night

Group: *A: Summary*; Subgroup: *3: Obs Per Night*; Slicer: *OneDSlicer*

	NVisits	OpenShutterFraction
	Per night	Per night
Median	815.00	0.73
Mean	806.44	0.73
Rms	185.52	0.02[illegible]

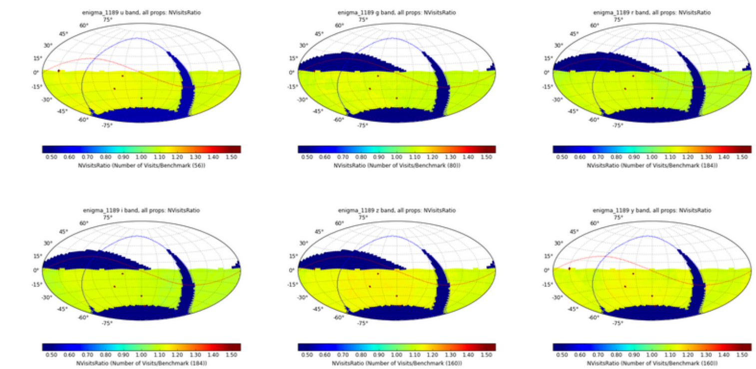
Shutter open 73% of the time (so 27% reading, slewing, changing filter)

Scheduler Validation, Examples

- Number of visits
- Co-added depth
- Mean/median seeing, airmass, skybrightness
- 572 plots generated per OpSim run by Scheduler Validation

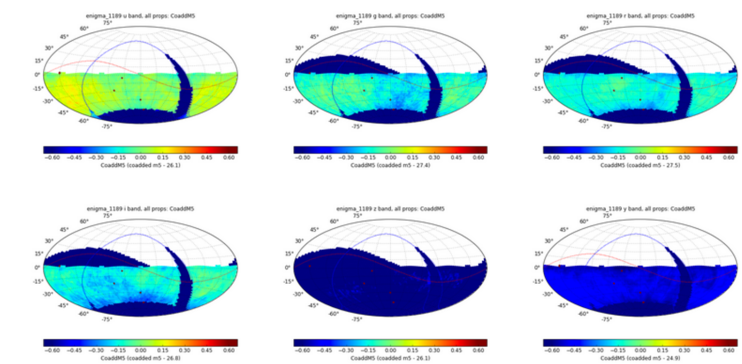
C: NVisits : All Props, ratio

NVisitsRatio



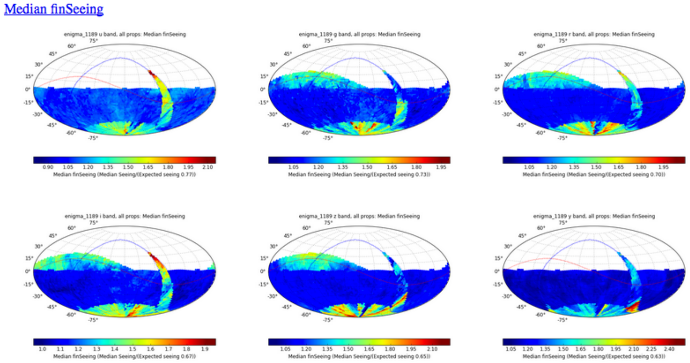
E: Coadded depth : All Props

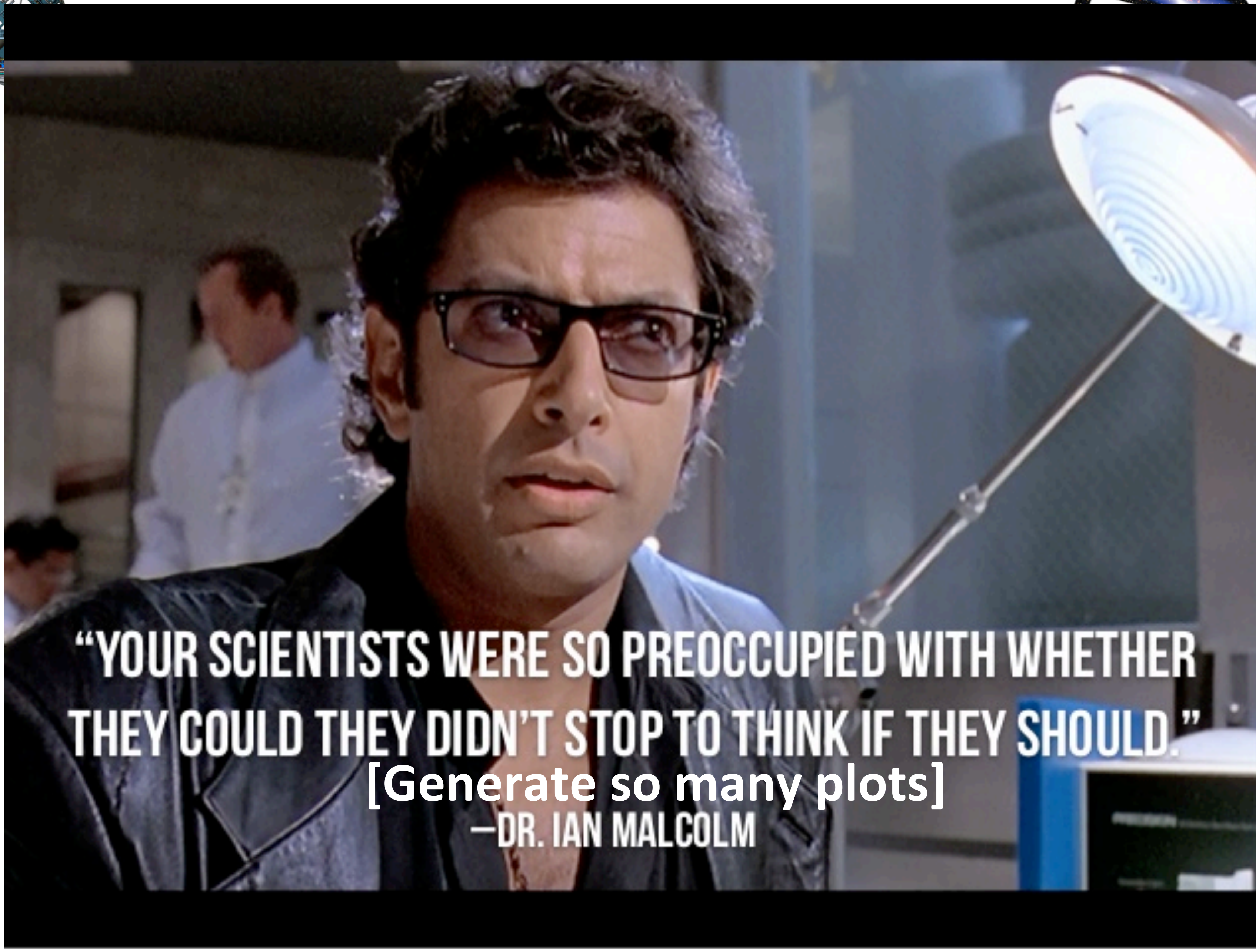
CoaddM5



G: Seeing : All Props

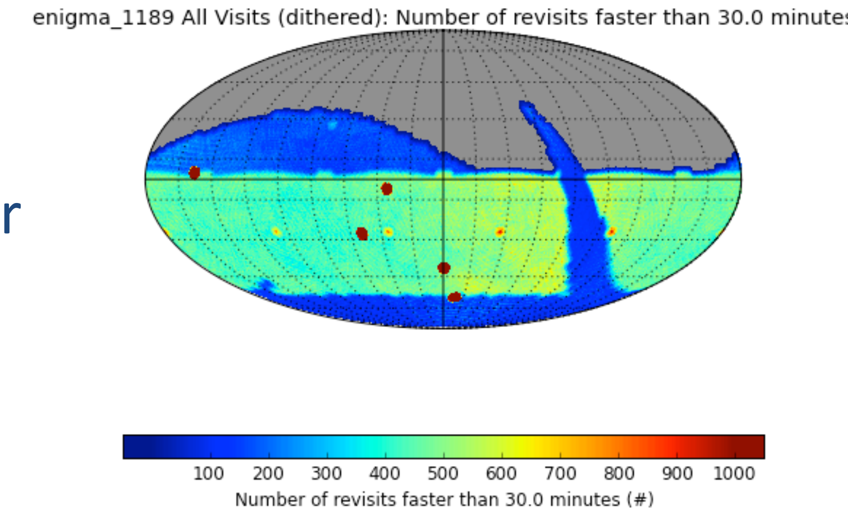
Median finSeeing

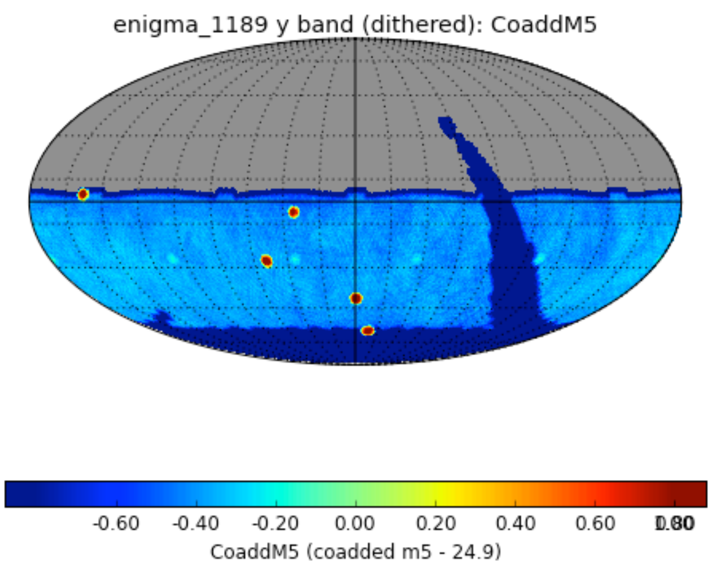


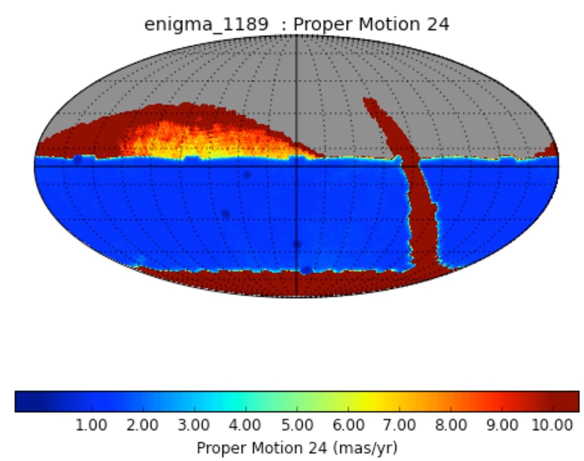

"YOUR SCIENTISTS WERE SO PREOCCUPIED WITH WHETHER THEY COULD THEY DIDN'T STOP TO THINK IF THEY SHOULD."
[Generate so many plots]
—DR. IAN MALCOLM

Science Performance

- SRD metrics for proper motion, parallax, and rapid revisits.
- Things like uniformity,
  coadded depth with dithering
- Computing with healpixels rather
  than per field
- Another 343 plots



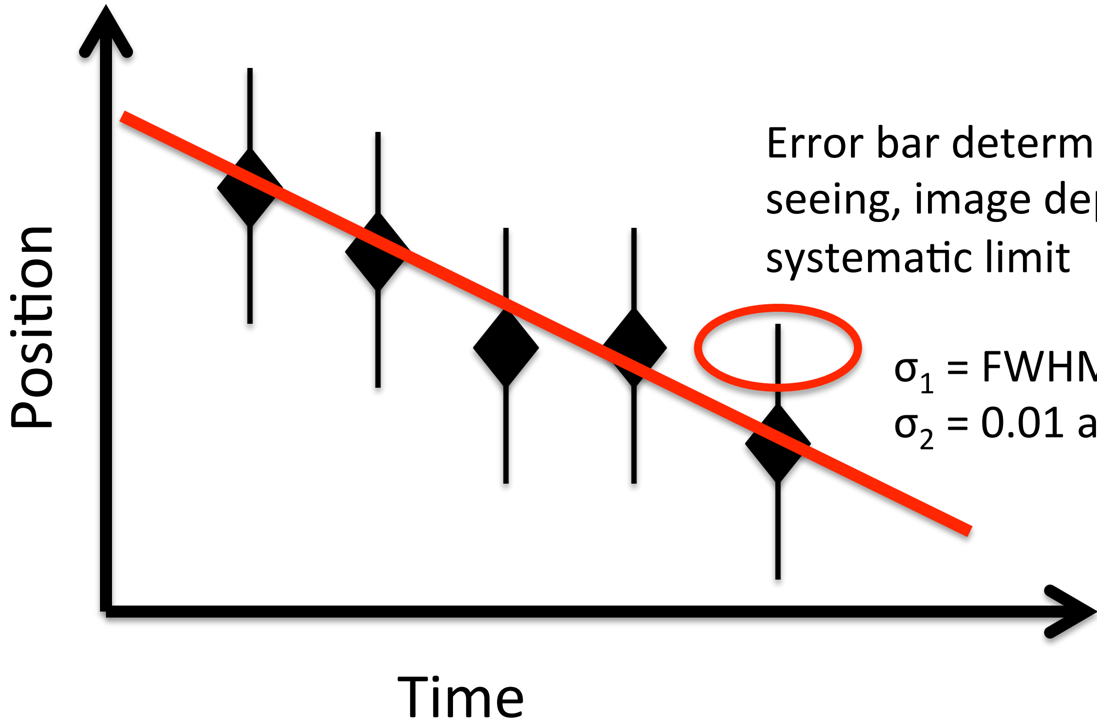




Details of Our Science Metrics

- The Scheduler Validation uses mostly simple metrics (mean, median, RMS, etc)
- Science metrics tend to be more complicated

Proper Motion and Parallax

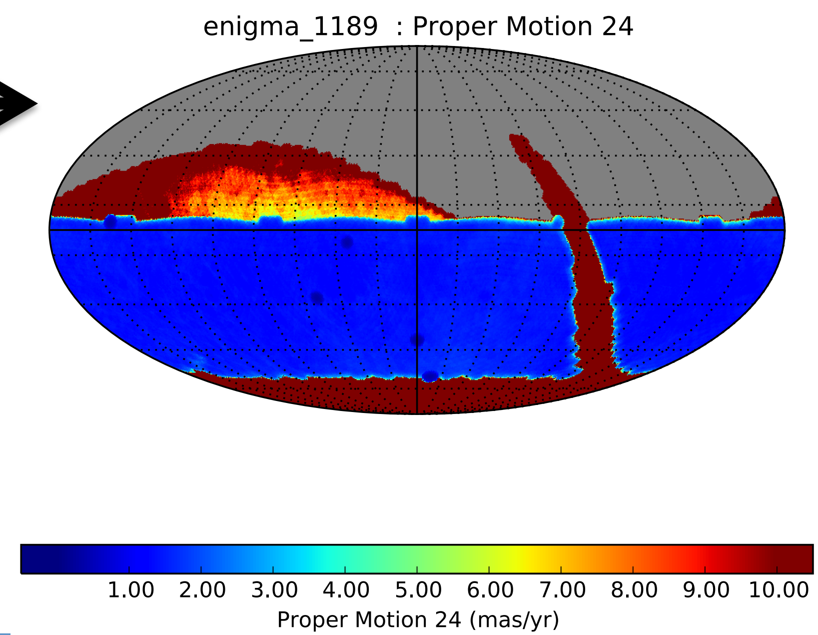


Error bar determined by seeing, image depth, systematic limit

σ_1 = FWHM/SNR
σ_2 = 0.01 arcsec

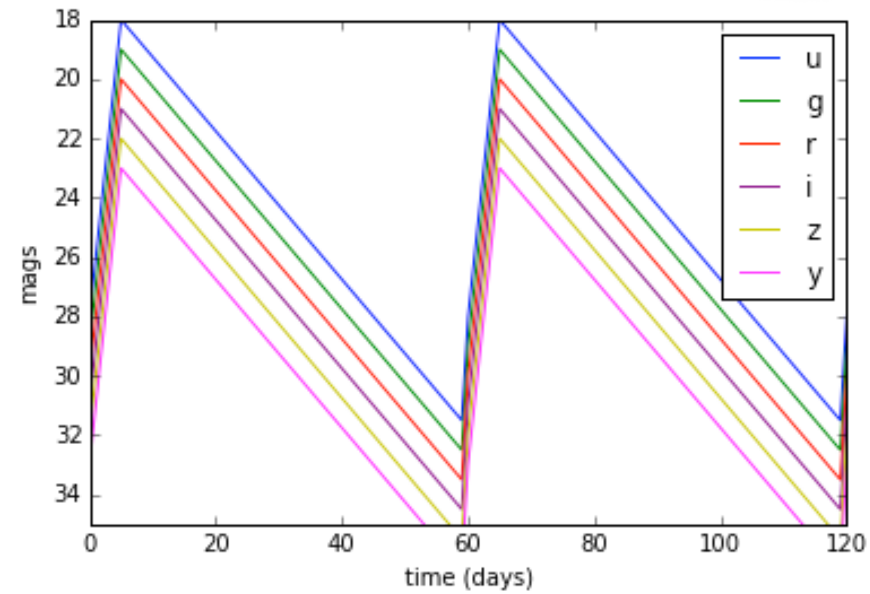
We can then compute the uncertainty on the slope of the best-fit line (only a function of times and uncertainties).

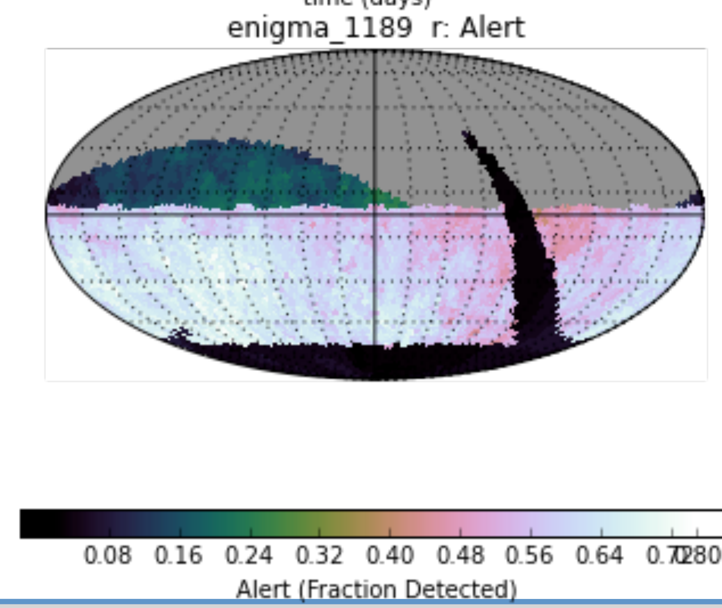
Put a 24th mag star at each healpixel, compute proper motion precision (defaults to flat SED, but can pick spectral type)



Transients

- Transient light curve as a simple saw-tooth
- Set detection criteria
  - Number of points before peak
  - Number of filters
  - How evenly the curve must be sampled
- Set off the transient continuously at each healpixel and find the fraction that meet detection criteria



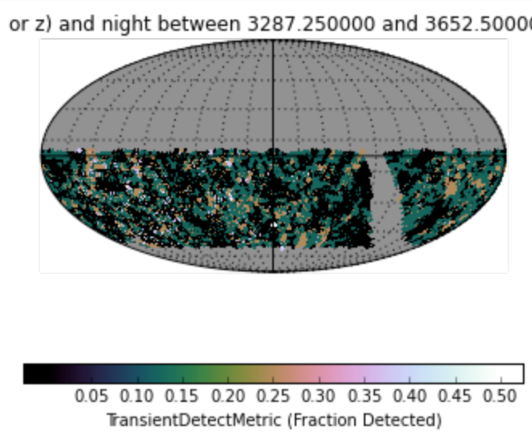


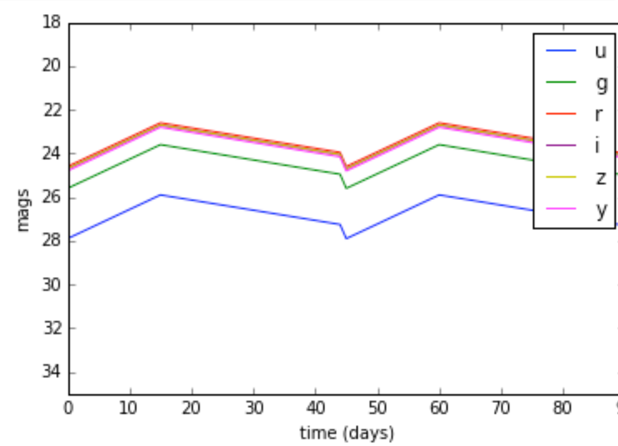
Transient Metric Configured for SNe Ia

– Transient metric configured for z=0.5 SN, different detection thresholds. **http://ls.st/10w**

What fraction are detected at least 6 times in one of g r i z, 3 in first half, 3 in second half

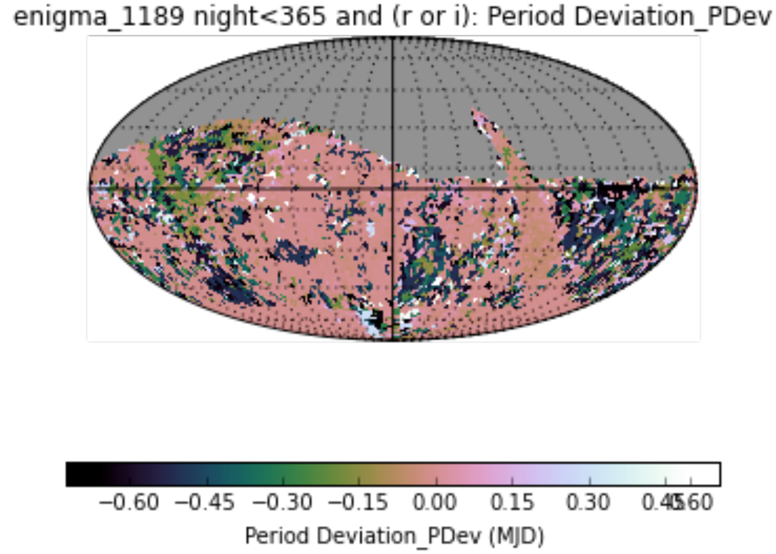
```
In [31]: transMetric = metrics.TransientMetric(riseSlope= -2./peakTime, declineSlope=1.4/30.,
                                               transDuration=transDuration, peakTime=peakTime, surve
         yDuration=1,
                                               nFilters=3, nPrePeak=3, nPerLC=2, **peaks)

         sqlconstraint = '(filter="r" or filter="g" or filter="i" or filter="z") and night between %
         f and %f' % (365.25*year,365.25*(year+1))
         transBundle = metricBundles.MetricBundle(transMetric, slicer, sqlconstraint,
                                                  runName=runName, summaryMetrics=summaryMetrics)
```

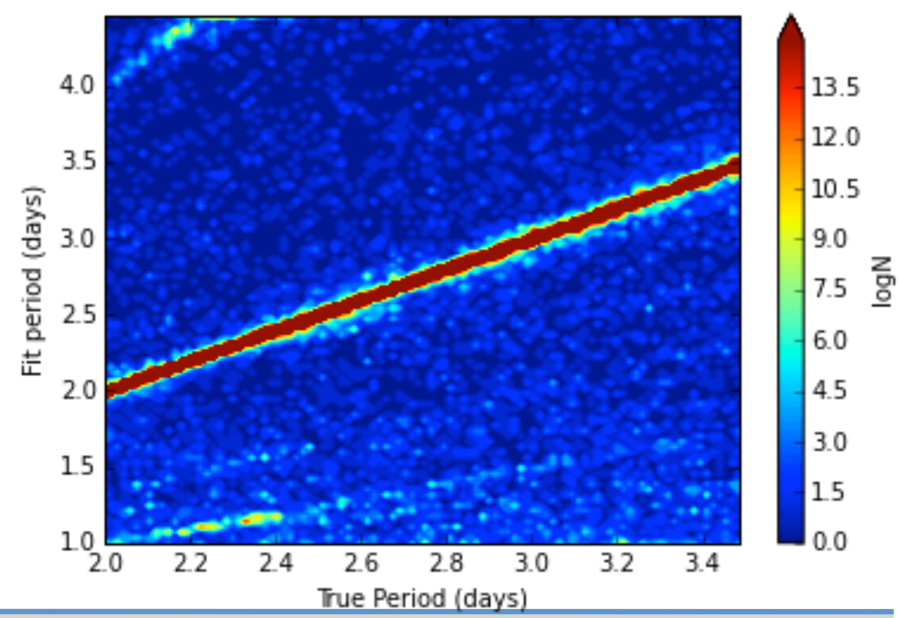




Period fitting

- Use scipy's Lomb–Scargle periodogram to check how well a period can be fit





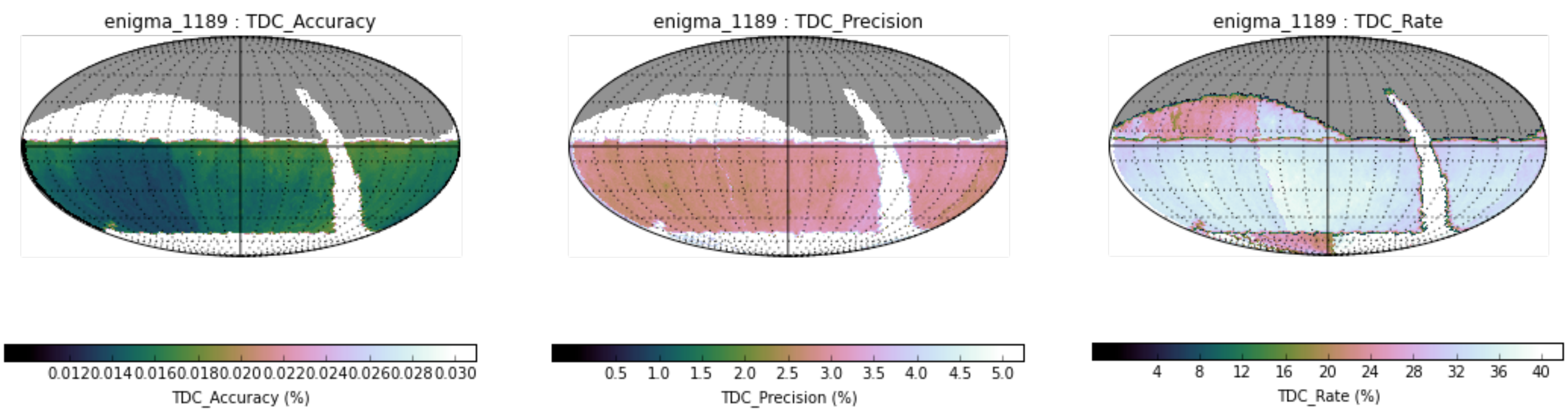
Time Delay Metric

– Strong lens time delay, accuracy, precision and success fraction

$$|A|_{\rm model} \approx 0.06\% \left(\frac{\rm cad}{\rm 3days}\right)^{0.0} \left(\frac{\rm sea}{\rm 4months}\right)^{-1.0} \left(\frac{\rm camp}{\rm 5years}\right)^{-1.1}$$

$$P_{\rm model} \approx 4.0\% \left(\frac{\rm cad}{\rm 3days}\right)^{0.7} \left(\frac{\rm sea}{\rm 4months}\right)^{-0.3} \left(\frac{\rm camp}{\rm 5years}\right)^{-0.6}$$

$$f_{\rm model} \approx 30\% \left(\frac{\rm cad}{\rm 3days}\right)^{-0.4} \left(\frac{\rm sea}{\rm 4months}\right)^{0.8} \left(\frac{\rm camp}{\rm 5years}\right)^{-0.2}$$



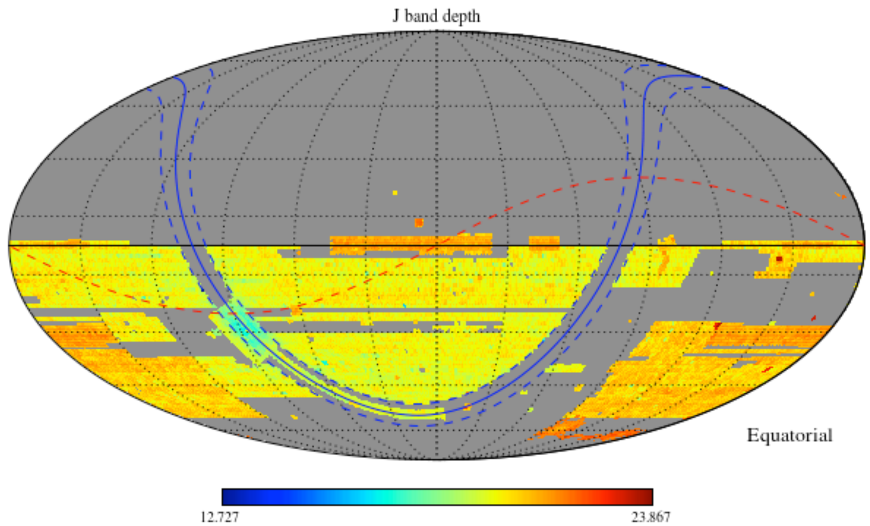
Metric contributed by Phil Marshall

Some obvious things we could use

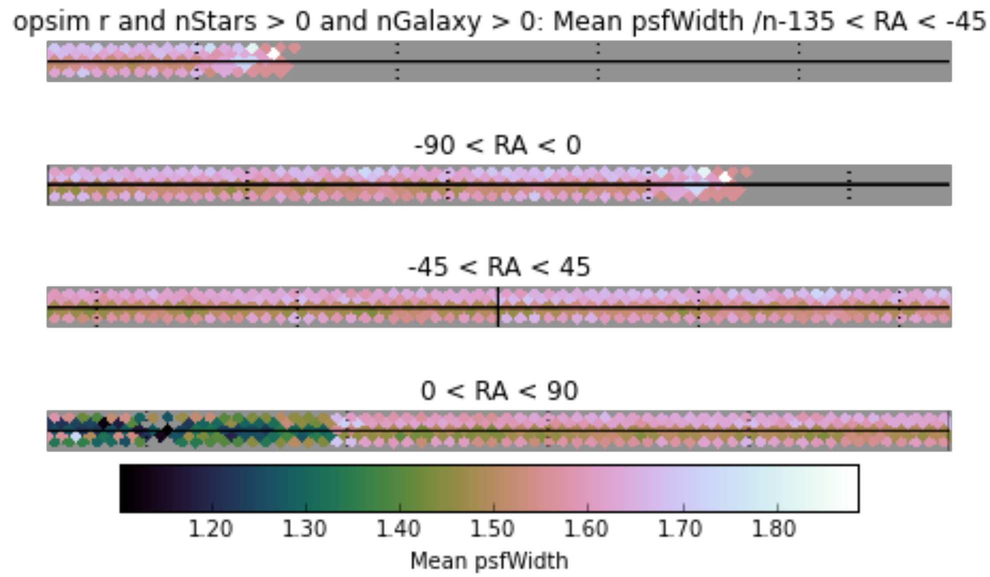
- Configure the period finding metric:
  - Can we get Cepheid distances to all the Local Group galaxies
  - Can we get RR Lyrae periods in the halo?
- Check for degeneracies in the astrometry observations (e.g., parallax and diffraction)
- Metrics for the Deep Drilling Fields
- Metrics for the LMC/SMC

MAF on other surveys

– We've used MAF to look at SDSS Stripe 82, VISTA, and CFHT



VISTA J-band depth

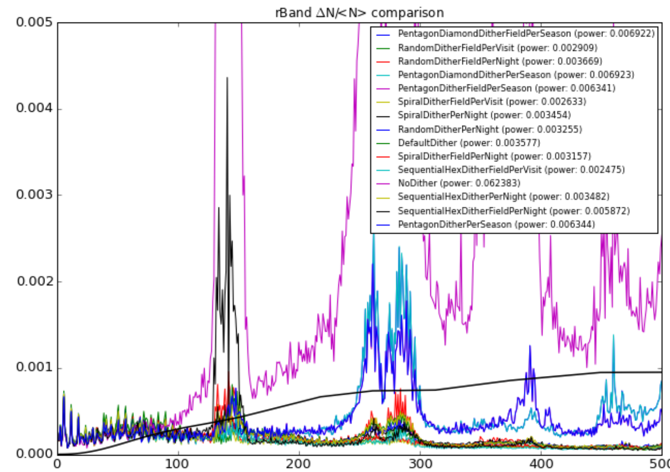


Stripe 82 mean seeing

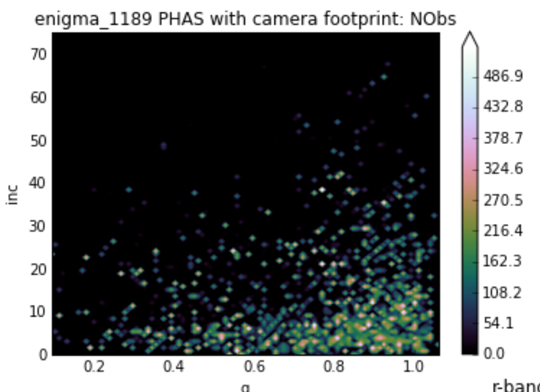
Take any pointing history, put it in a database, and MAF can start making plots!

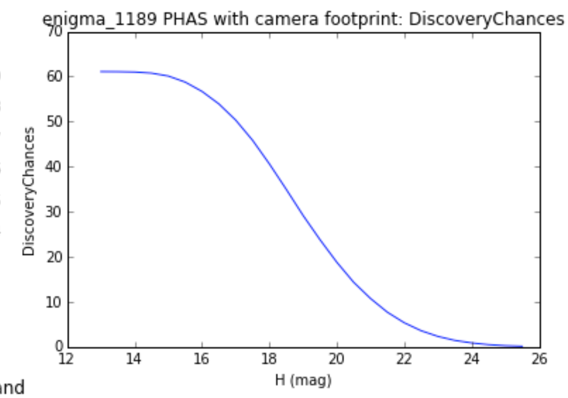
Active Research With MAF

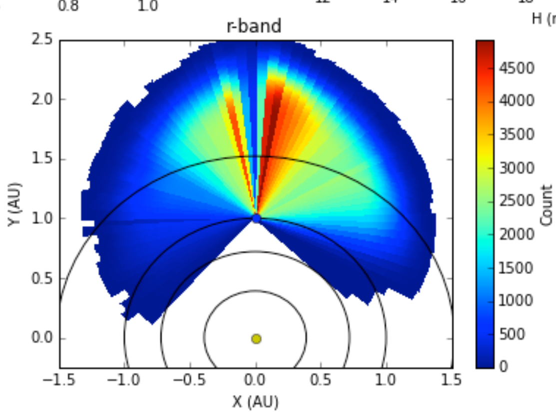
- Dither strategies on galaxy power spectrum
- Optimization for transient objects
- Including solar system objects

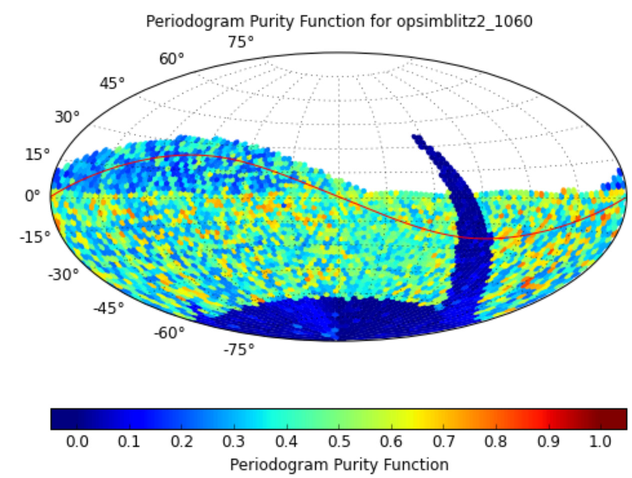


Awan et al, in prep





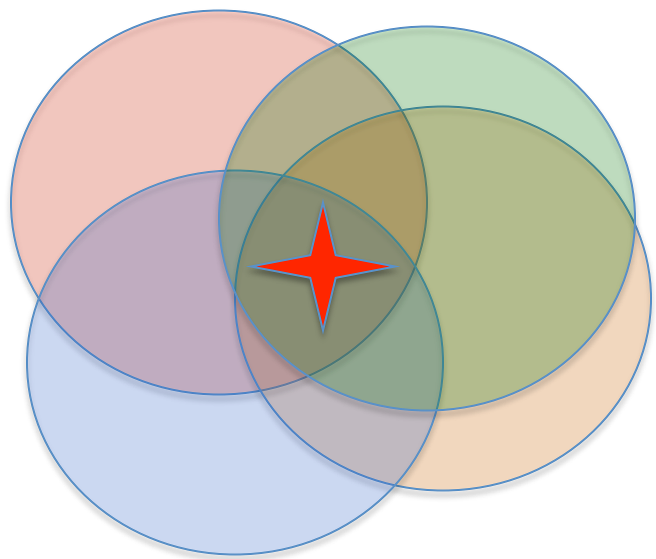




Lund et al., in prep

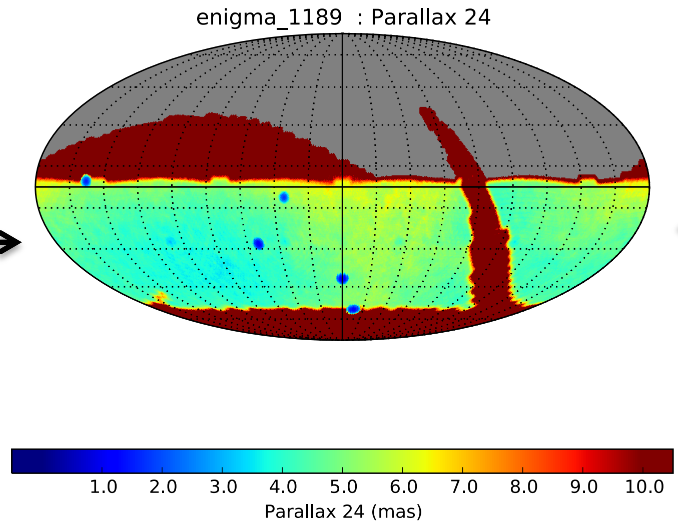
We want more metrics (and help configuring the metrics we have)!

Given a list of observations at a spot in the sky, how well can you do your science? **What is the algorithm?**



We need help connecting metric outputs to science results!

MAF



Given OpSim, we can use MAF to compute the parallax precision. Is that good enough to detect a new stream in the halo?

We can now say when one OpSim run is “better” or “worse” wrt some metric. Now we need to figure out if it’s “better enough to care”.

Conclusions

- MAF is a handy tool for analyzing survey performance
- We'd like more science metrics! Need to tie metrics to discovery thresholds.
- Go use it! Lots of example iPython notebooks in the git repo to help people get started:
  - sims_maf_contrib (includes tutorial iPython notebooks): https://github.com/LSST-nonproject/sims_maf_contrib **http://ls.st/zm9**
  - sims_maf: https://github.com/lsst/sims_maf **http://ls.st/k1f**

- Don't forget to cite MAF, OpSim, and LSST: **http://ls.st/ll9**

LSST Ecosystem

Data Management (aka DM): Take a bunch of raw images and process them to make catalogs

Sims:

- **Photon Simulator (aka PhoSim)**: Simulate LSST images by tracing photons
- **Catalog Simulator (aka CatSim)**: A realistic catalog of objects LSST could observe (Stars, galaxies, etc)
- **Calibration Simulation (aka CalSim)**: Uber-cal for LSST
- **Operations Simulator (aka OpSim)**: Simulate the operations of the telescope (motion of the motors) as well as the scheduling of observations.
- **Metric Analysis Framework (MAF)**: Analyze and visualize how well an astronomical survey performs.

The (LSST) Stack

A better name?

Survey Performance Analysis Tool

SPAT