A Glorious Dawn – Observing the First Galaxies With Hubble

Matt Mechtley – Arizona State University
School of Earth and Space Exploration

1

A Glorious Dawn – Observing the First Galaxies With Hubble

Matt Mechtley – Arizona State University
School of Earth and Space Exploration

Wik:

Rogier Windhorst, Seth Cohen, Linhua Jiang, Rolf Jansen,
Evan Scannapieco (ASU)
Haojing Yan (U. Missouri)
Russell Ryan, Anton Koekemoer (STScI)
Glenn Schneider, Xiaohui Fan (U. Arizona)
Nimish Hathi (Carnegie Observatories)
Bill Keel (U. Alabama)
Huub Röttgering (Leiden Observatory)
Donald Schneider (Penn. State)
Michael Strauss (Princeton)
HIPPIES Team

Overview

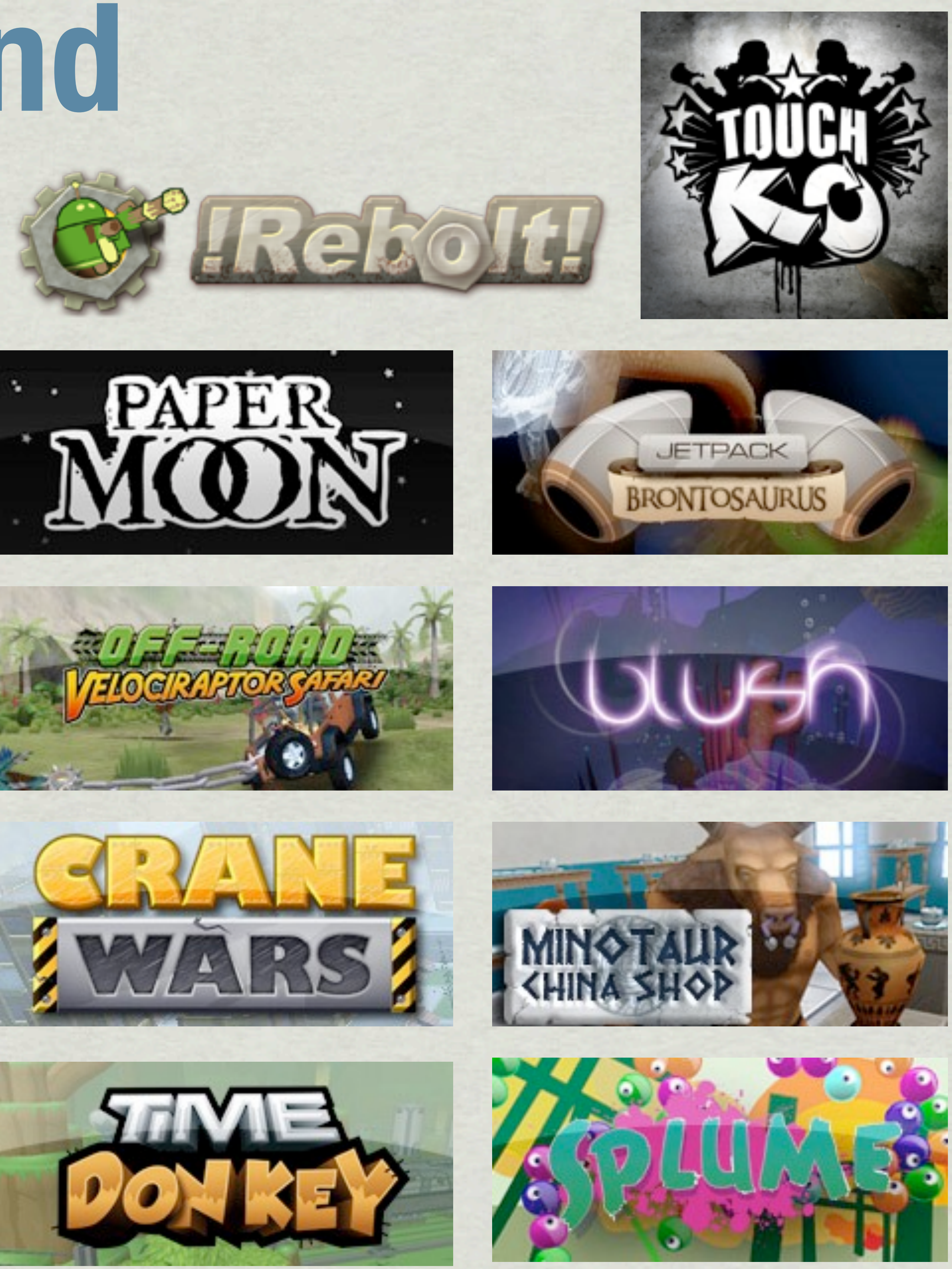
- My Background: Who is this dude?
- Nuts and Bolts: Python and why I use it
- Cosmology: Why study the earliest galaxies?
- HIPPIES: Searching for early galaxies
- My dissertation: Host galaxies of the earliest quasars

My Background

- B.S. Mathematics (Linear & Abstract Algebra)
- Professional game developer for 2 years

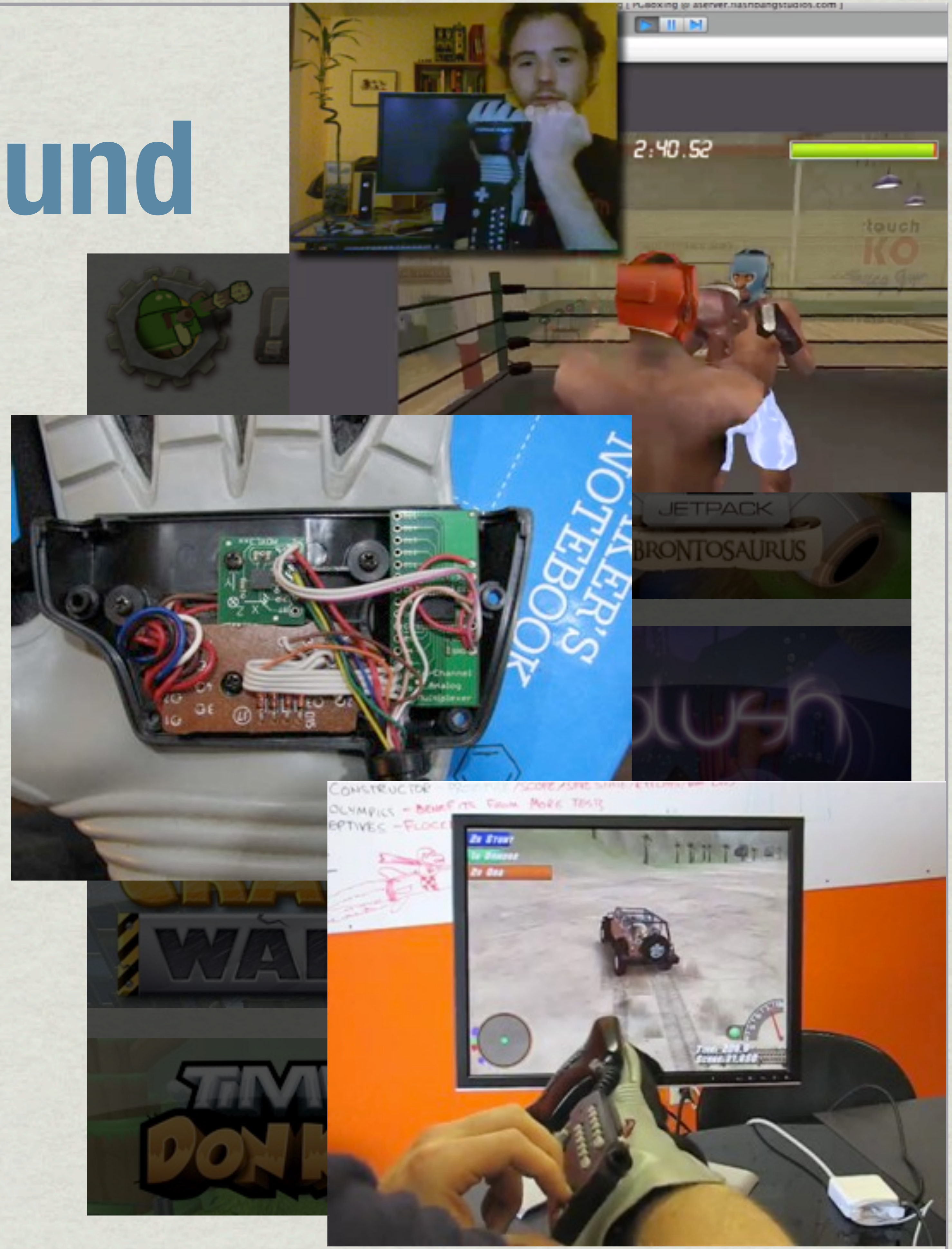
My Background

- B.S. Mathematics (Linear & Abstract Algebra)
- Professional game developer for 2 years
- Lots of games!



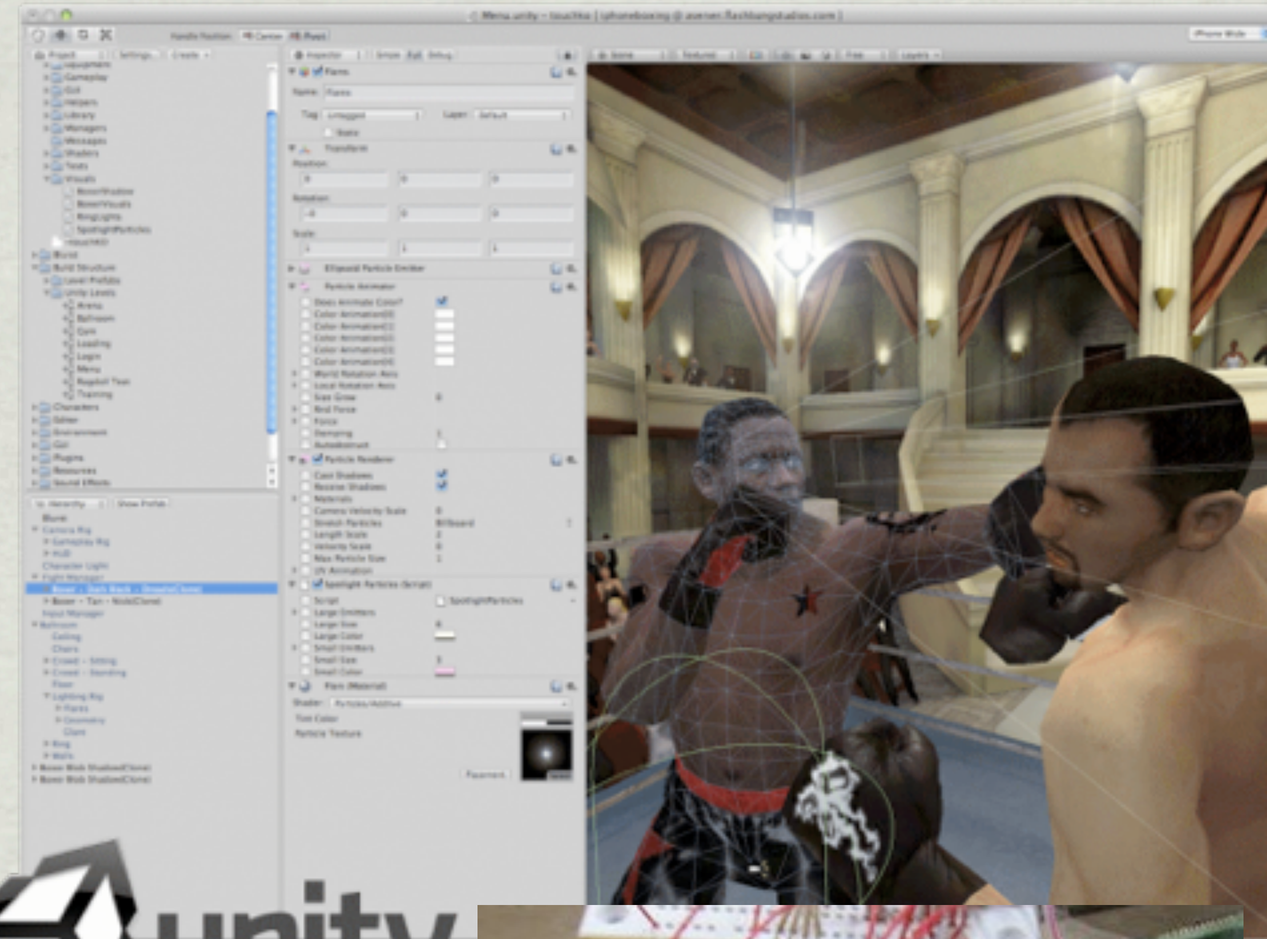
My Background

- B.S. Mathematics (Linear & Abstract Algebra)
- Professional game developer for 2 years
- Lots of games!
- Also some hardware hacking



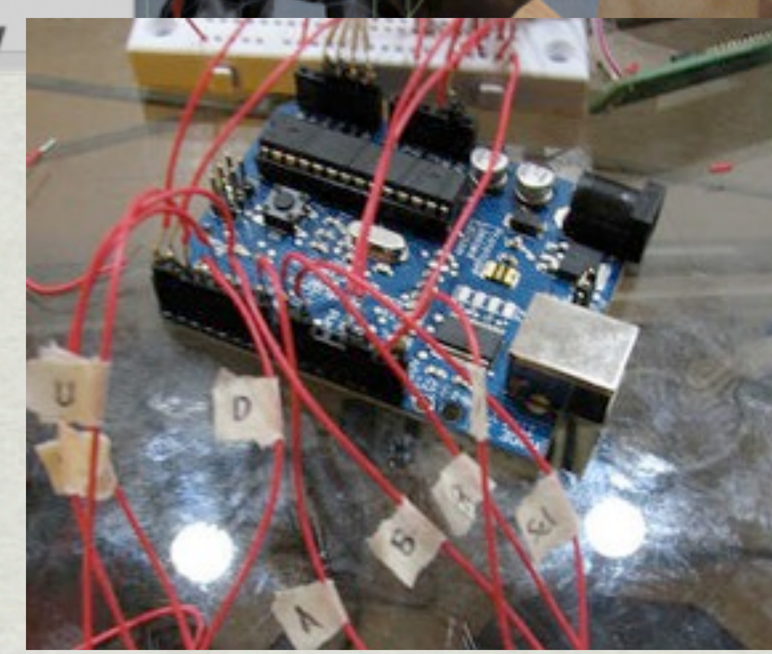
What I Learned From Games

- **PROTOTYPE**
- Good tools make happy developers
- Choose tools that enable rapid iteration (of the modify–build–test loop)
- Lets you fail *early*, avoiding wasted time



unity

ARDUINO



Optimization

* Most programmers learn how to optimize for memory and execution efficiency
* It’s also important to optimize for programmer time
* Saving a day of compute time is great! But what if the optimization took a week to write?
* Computers are constantly getting faster, your day is not constantly getting longer

Rapid Iteration With Python

- Interpreted language – no compiling (modules compiled to bytecode)
- Comprehensive standard library (file system manipulation, strings, subprocesses, sockets, sqlite databases, urls, etc.)
- Installing additional modules/libraries is easy:
  > python setup.py install
- Interactive console – excellent for prototyping
- Vast community/ecosystem – scientists of all flavors, web developers, game developers, robotics folks
  (Search StackOverflow, your problem has likely been solved)

Python's Unique Features

- Readability – "One of Guido [van Rossum]'s key insights is that code is read much more often than it is written." – PEP 8, Python Style Guide
- Iterators – Produce more readable loops, including flattening nested loops
- List Comprehensions – Compact syntax for simple loops

```python
squares = [x**2 for x in range(10)]

# Generally, any functions, nested functions, etc. can occur in the generator
names = ['Snake Plissken', 'Patrick Bateman', 'Dirty Harry', 'Debbie Harry']
lastNames = [name.split()[1] for name in names]
```

Readability: A Sample Program

- Despite only a single comment, most non-Python programmers can tell you what this is doing
- Reduces "WTF was I doing?" time when you have to come back to old code

```python
from math import sin, pi

def sinc(x):
    """Compute the sinc function:
    sin(pi*x)/(pi*x)"""
    try:
        val = (x*pi)
        return sin(val)/val
    except ZeroDivisionError:
        return 1.0

input = [0.0,0.25,0.5,0.75,1.0,
         1.25,1.5,1.75,2.0]
output = [sinc(x) for x in input]
```

From T. Oliphant 2007

“But interpreted languages are too slow”

- Numpy/Scipy are compiled C/Fortran routines with Python bindings. Their algorithms have already been optimized by smart people!

- In many cases, they’re the *same* routines you would have used anyway (e.g. lapack). The only difference in speed comes from the external call.

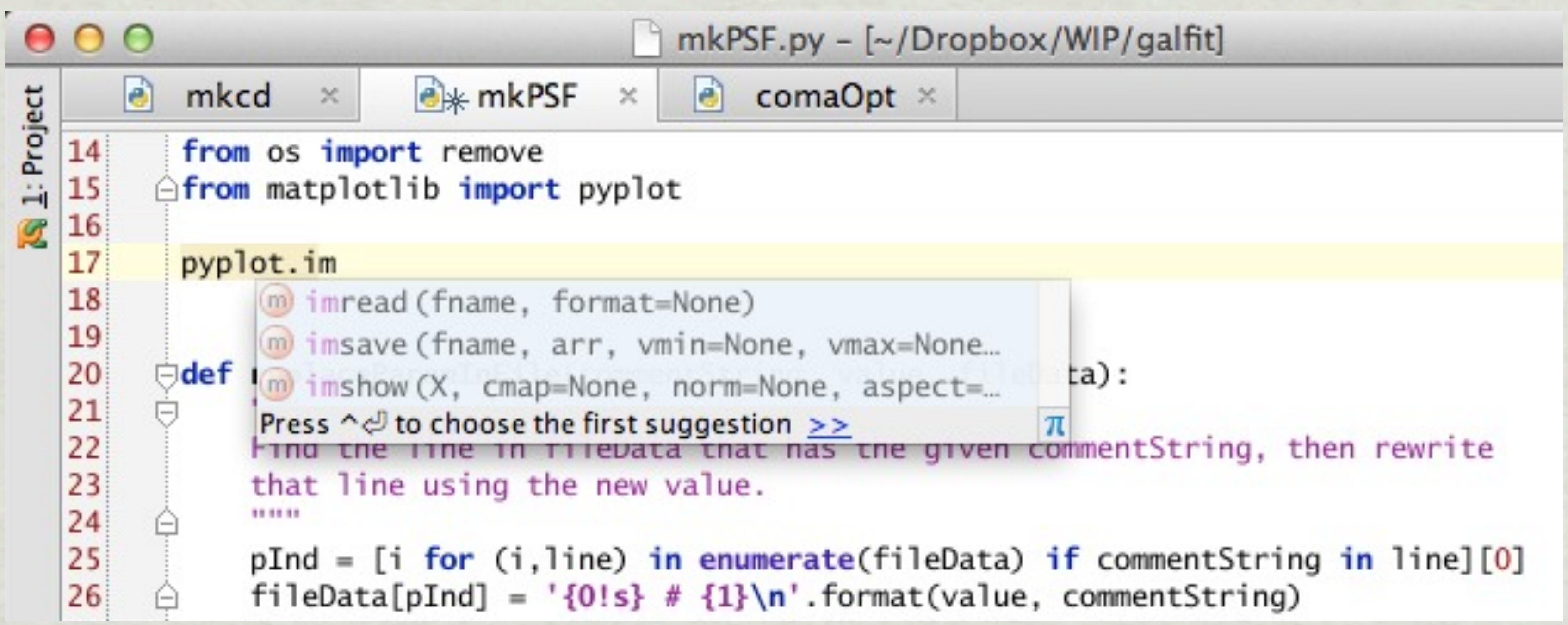
- Vectorized functions/operations for compact syntax, including common linear algebra operations

- Can also write your own compiled extension modules – Slick tools for this too, like Cython, f2py, etc.

10

PSA: A Text Editor Can be a Rapid Iteration Tool, Too

- Keep trusty vi/emacs as the hammer in your toolbox, but there are nailguns out there

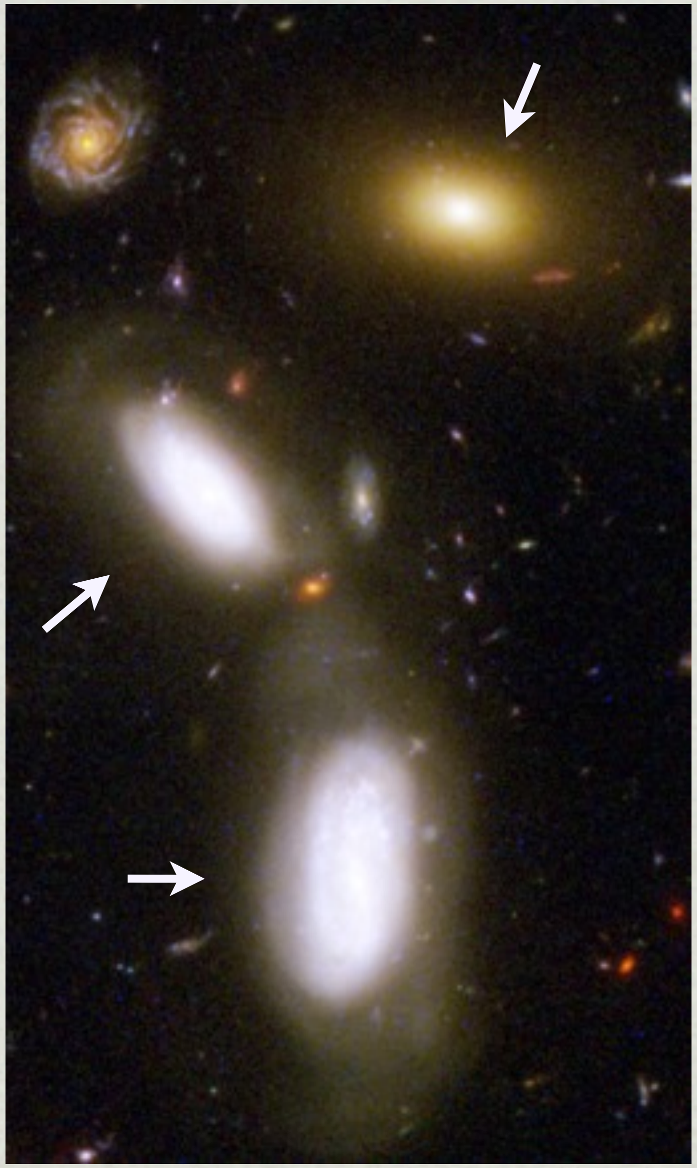


11

OK, ONTO THE SCIENCE

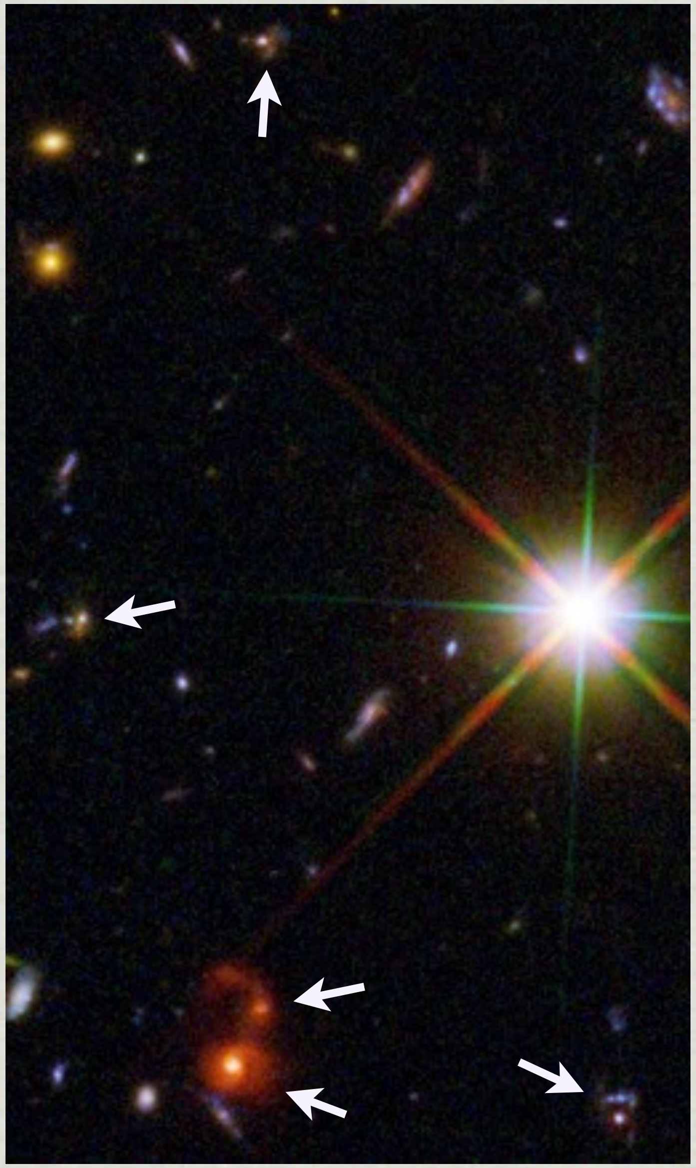
1 Zwicky 18, Extremely Metal-Poor Galaxy – A local analog to galaxies in the early Universe?

Evolution of Galaxy Morphology



Redshift z=0.1
1–2 Gyr Ago

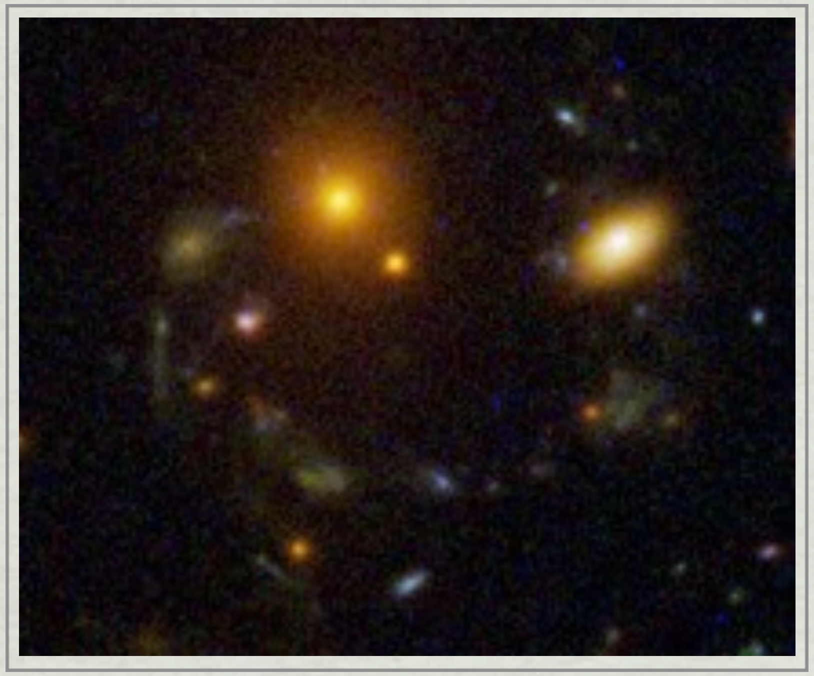
Redshift z=1.0
7–8 Gyr Ago



Redshift z>2
>10 Gyr Ago

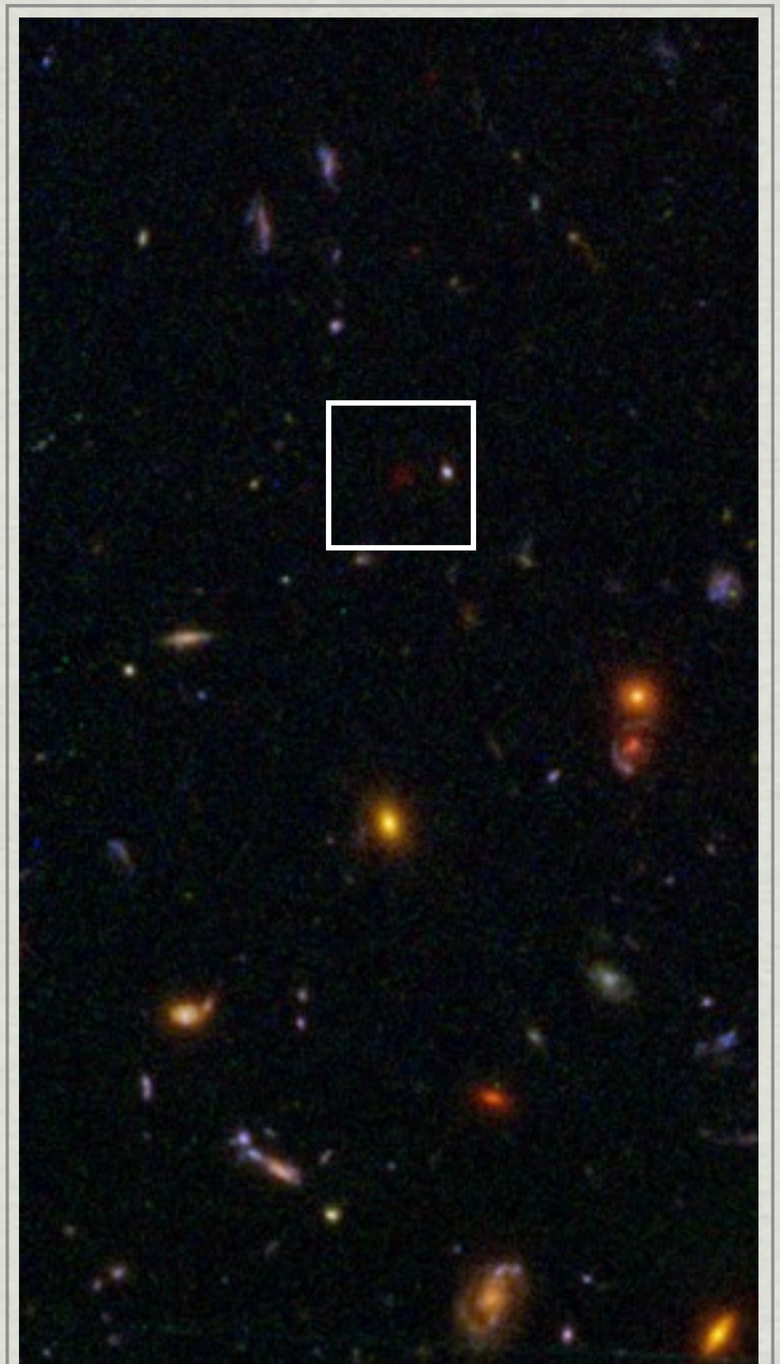
13

The Universe Smiles at You



14

The Most Distant Galaxies



Redshift z~8
12.9 Gyr Ago

15

Galaxy Assembly – Growth via Mergers*

*And probably some cold-mode accretion, too

Tadpole Galaxy
(Minor Merger)

Antennae Galaxies
(Major Merger)

Images: Hubblesite

History of the Universe

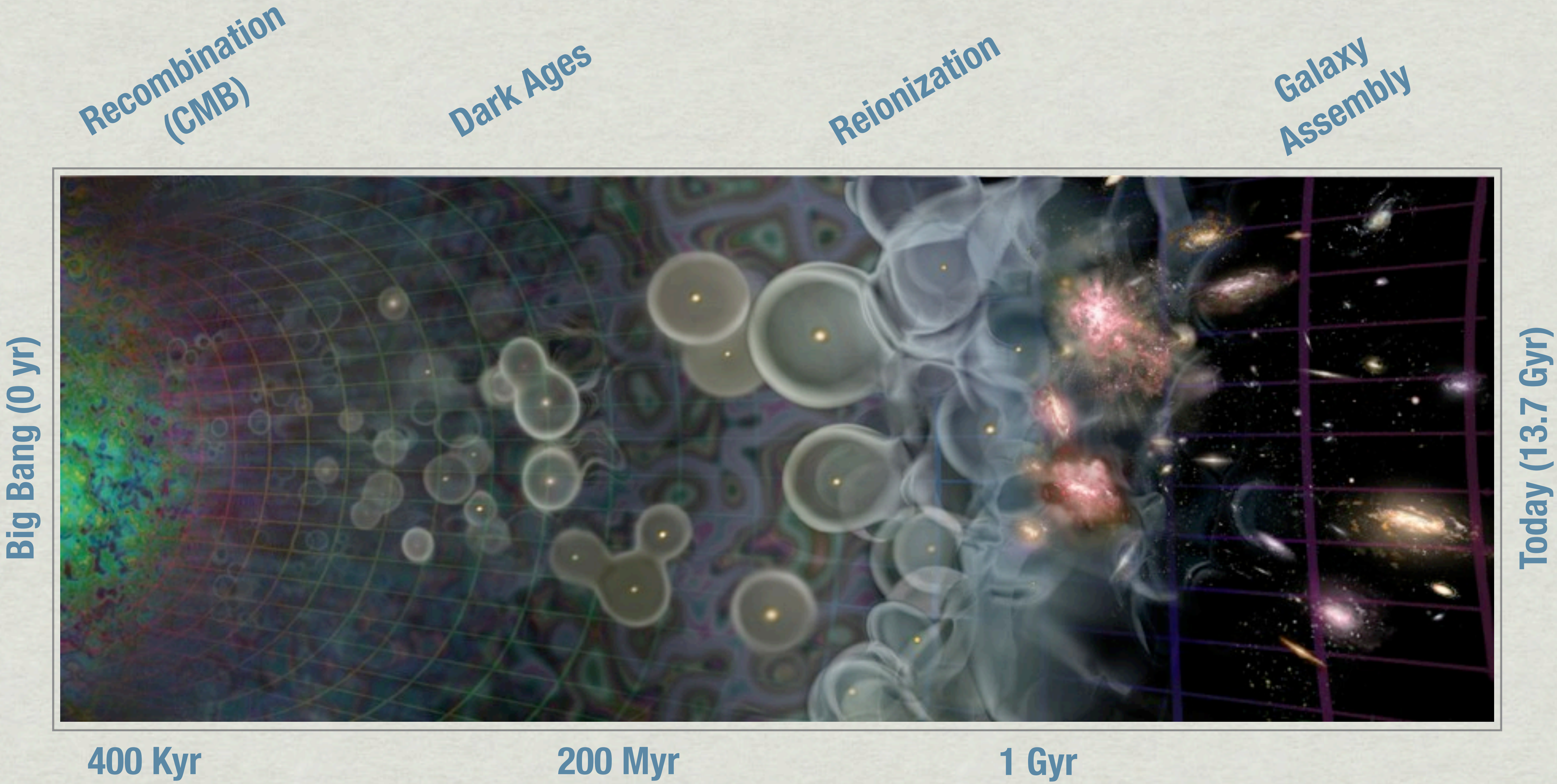
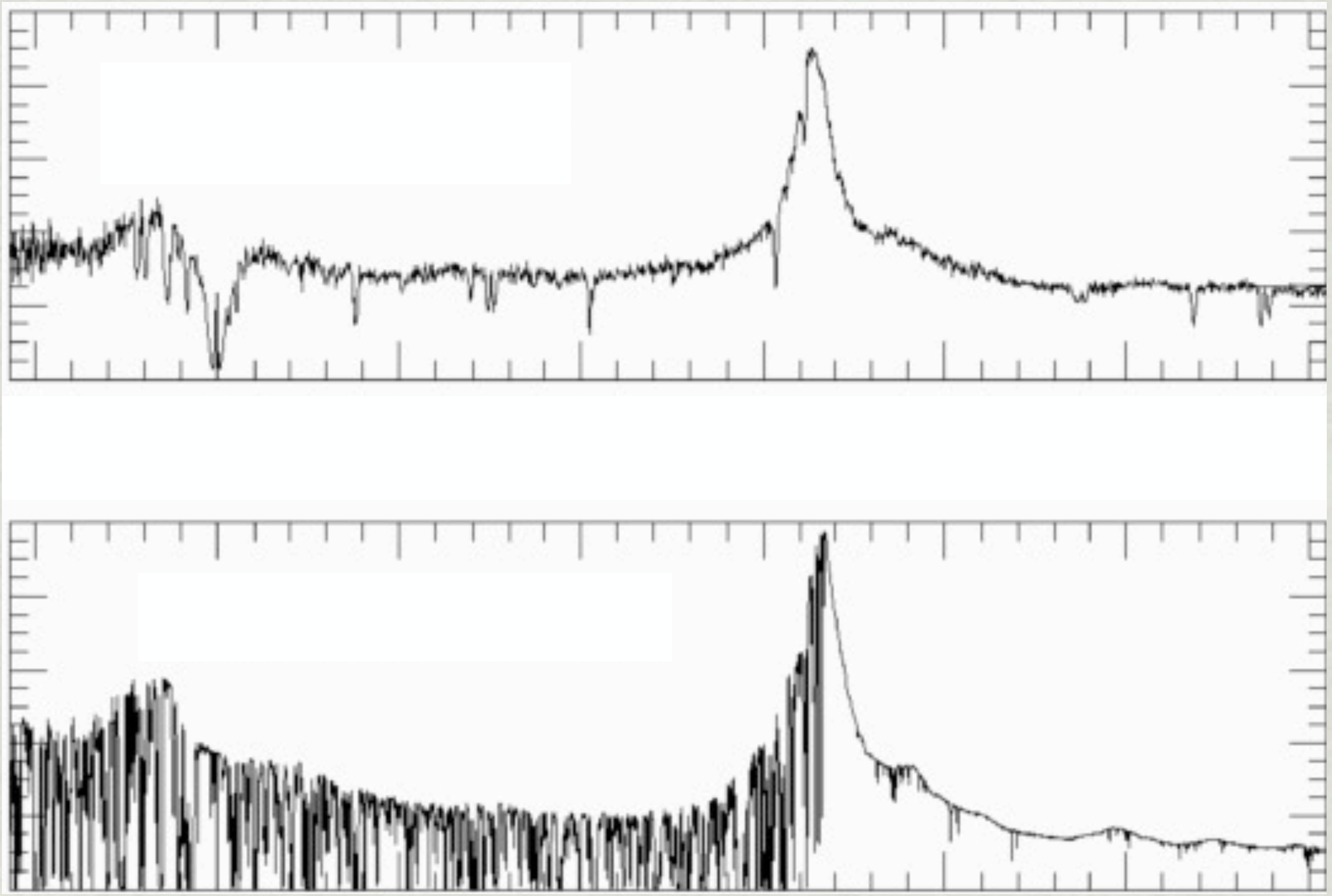


Image: Scientific American

Evidence for Reionization: Lyman Alpha Forest

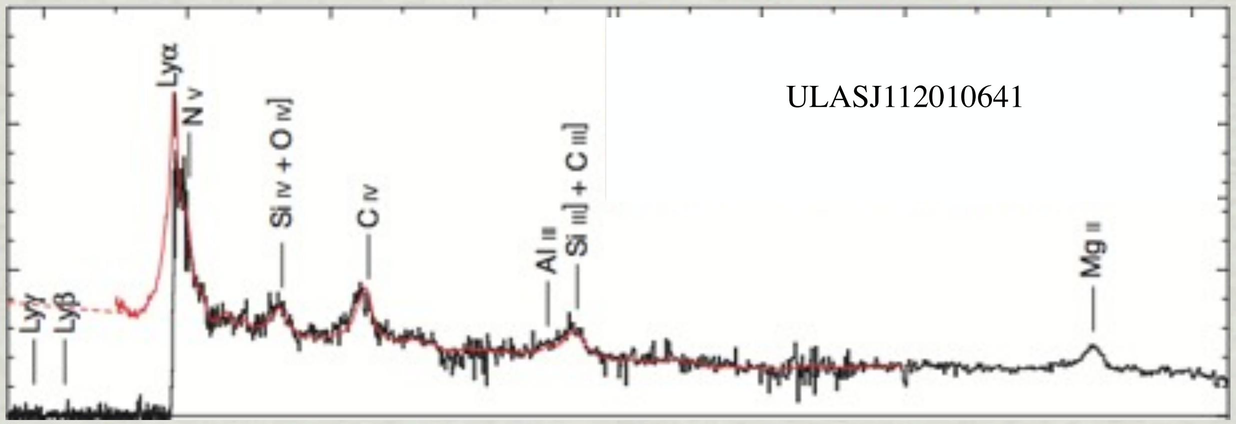
3C 273
z=0.16

Q1422+2309
z=3.6



W. Keel / E.L. Wright

ULAS J1120+0641
z=7.1

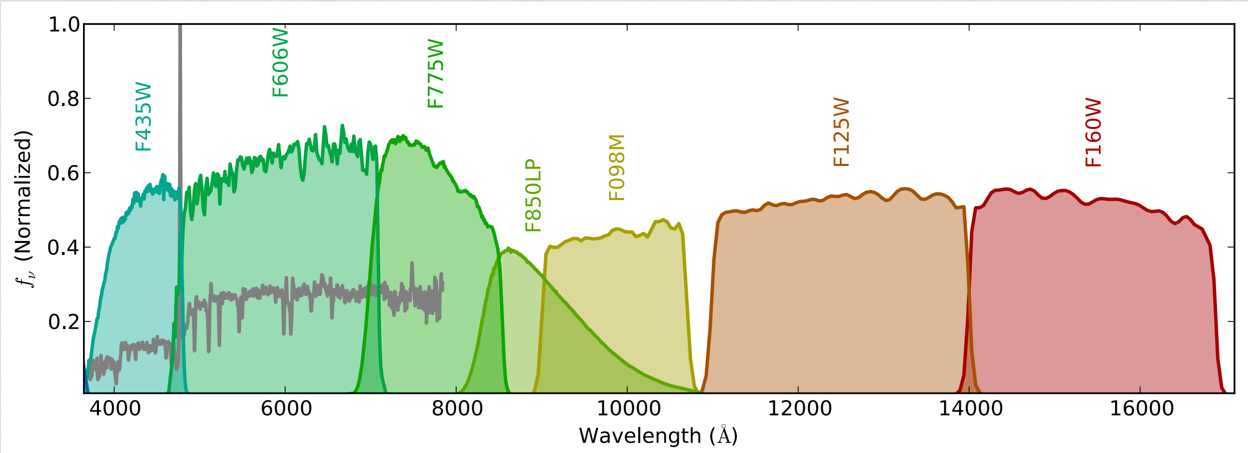


Mortlock et al. 2011

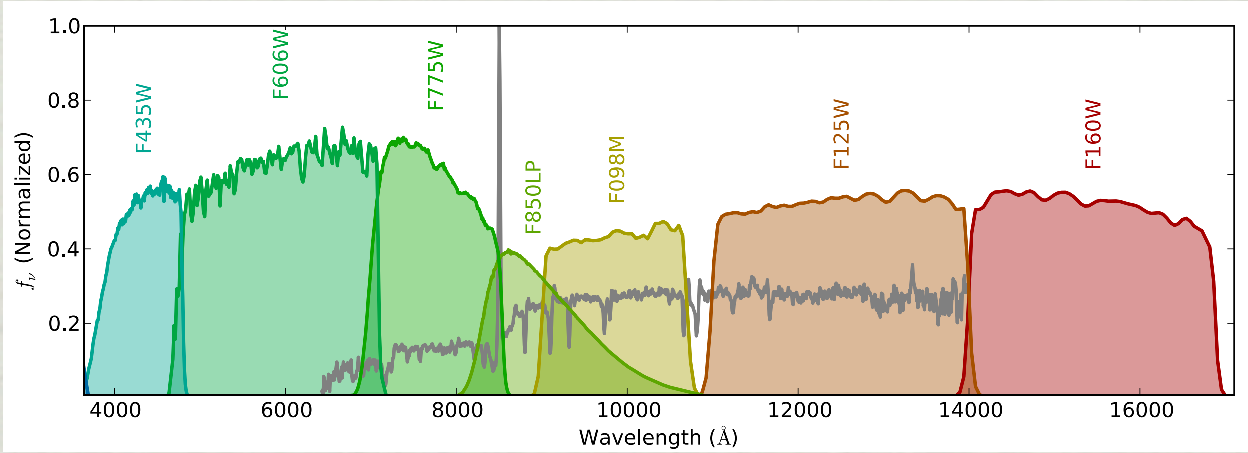
How to find distant galaxies

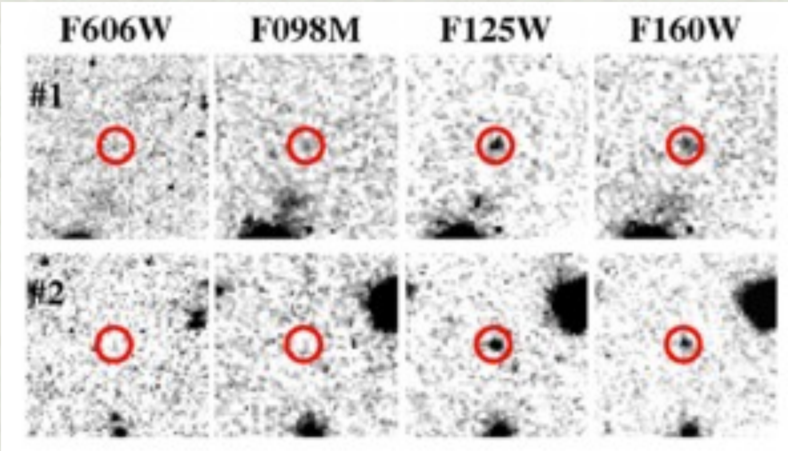
Lyman break or "dropout" technique: Look for objects that are visible in red filters, but not visible in blue ones

Redshift z=3
11.5 Gyr ago



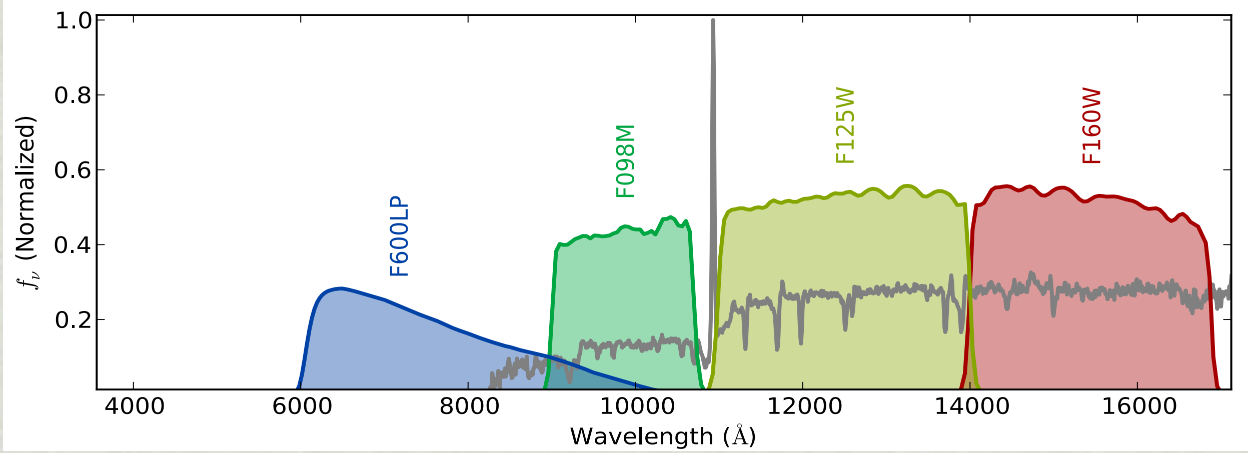
Redshift z=6
12.7 Gyr ago





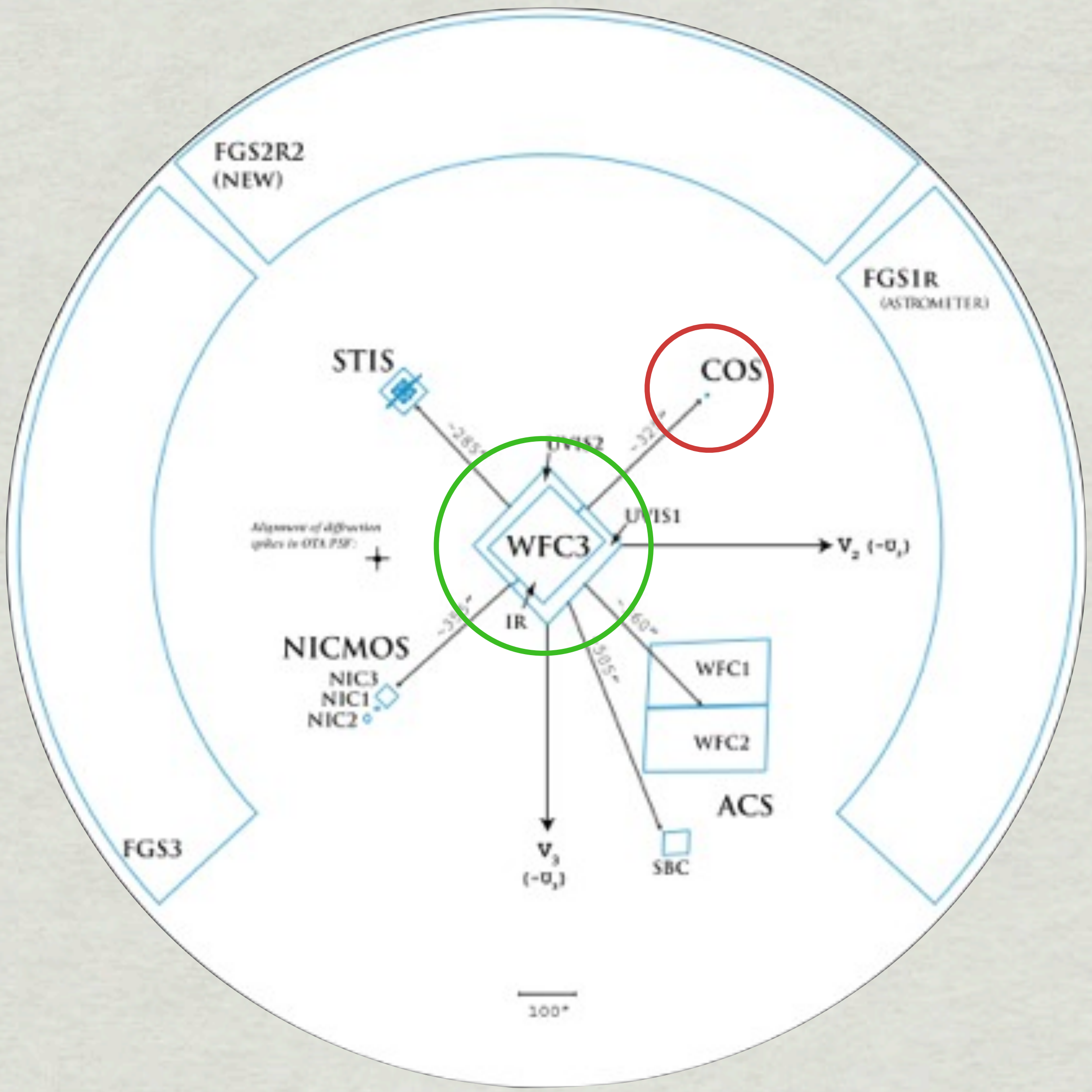
Yan et al. 2011

Redshift z=8
13.0 Gyr ago



Spectrum: Shapley et al. 2003

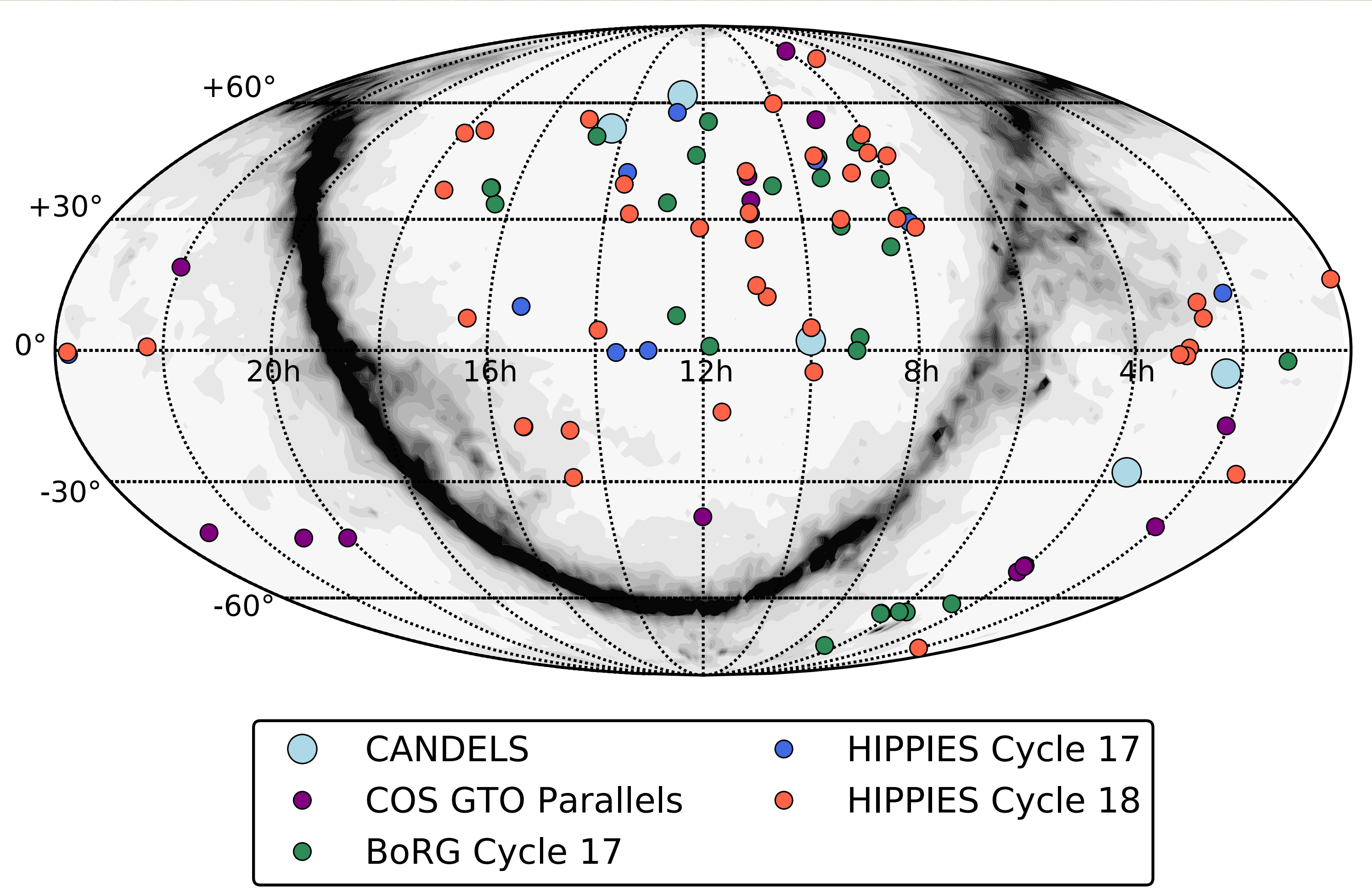
HIPPIES: Hubble Infrared Pure Parallel Imaging Extragalactic Survey



Hubble Space Telescope Focal Plane

- While other observers use the Cosmic Origins Spectrograph, we image a random patch of sky
- “Pure Parallel” since we don’t get to pick where we look
- Bonus data!
- Random fields avoid biases associated with existing well-studied fields

HIPPIES Fields



- 91 random fields
- 4–5 filters each
- 1141 individual exposures

- Need to automate:
  Interpolate over bad pixels
  Resample images
  Align images
  Measure correlated noise
  Detect objects

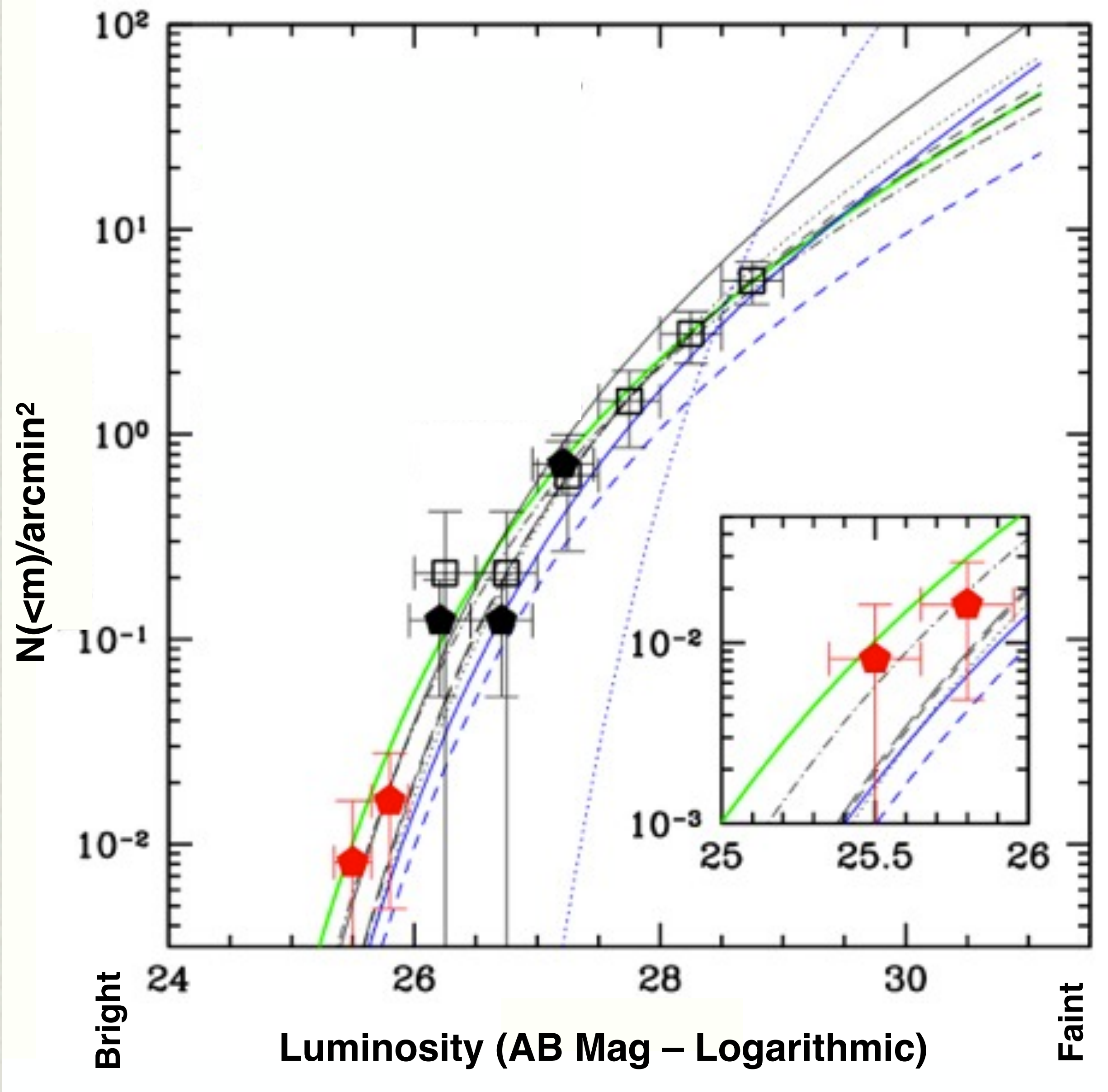
HIPPIES Python Pipeline

* Fully automated (just type go and leave for lunch)
* Replaced an IRAF task for calculating correlated noise and scaling the pixel-to-pixel uncertainty map. IRAF writes all sub-steps to disk, the Python version works in memory.
* Uses Numpy/Scipy for tasks like median filtering and autocorrelation. Speed is actually faster than IRAF.
* ~1300 lines of code and ~1.5 months of work, it's a beast mostly because of error-checking

HIPPIES Primary Science Goal: Measuring the z>7 Luminosity Function

- Essentially a luminosity histogram
- Useful for determining if there are enough galaxies to reionize the Universe
- HIPPIES gets the bright end (needs area)
  HUDF gets fainter bits (needs depth)
- HIPPIES avoids cosmic variance biases

Redshift z=7

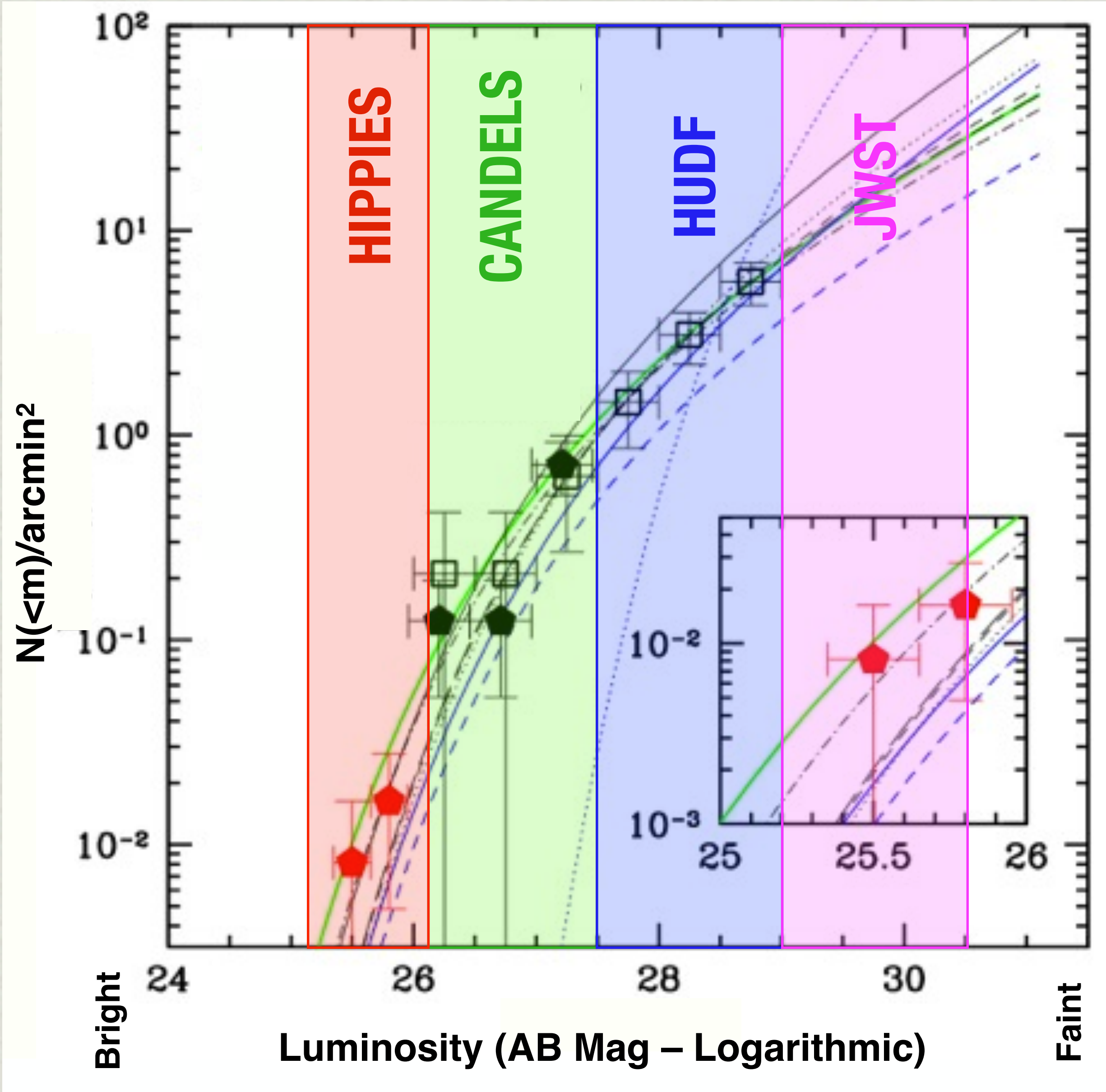


Yan et al. 2011

HIPPIES Primary Science Goal: Measuring the z>7 Luminosity Function

- Essentially a luminosity histogram
- Useful for determining if there are enough galaxies to reionize the Universe
- HIPPIES gets the bright end (needs area)
  HUDF gets fainter bits (needs depth)
- HIPPIES avoids cosmic variance biases

Redshift z=7



Yan et al. 2011

QUASARS

AGN Unification Model: Quasars are Active Galactic Nuclei (like Centaurus A here), pointed at your face

Quasars

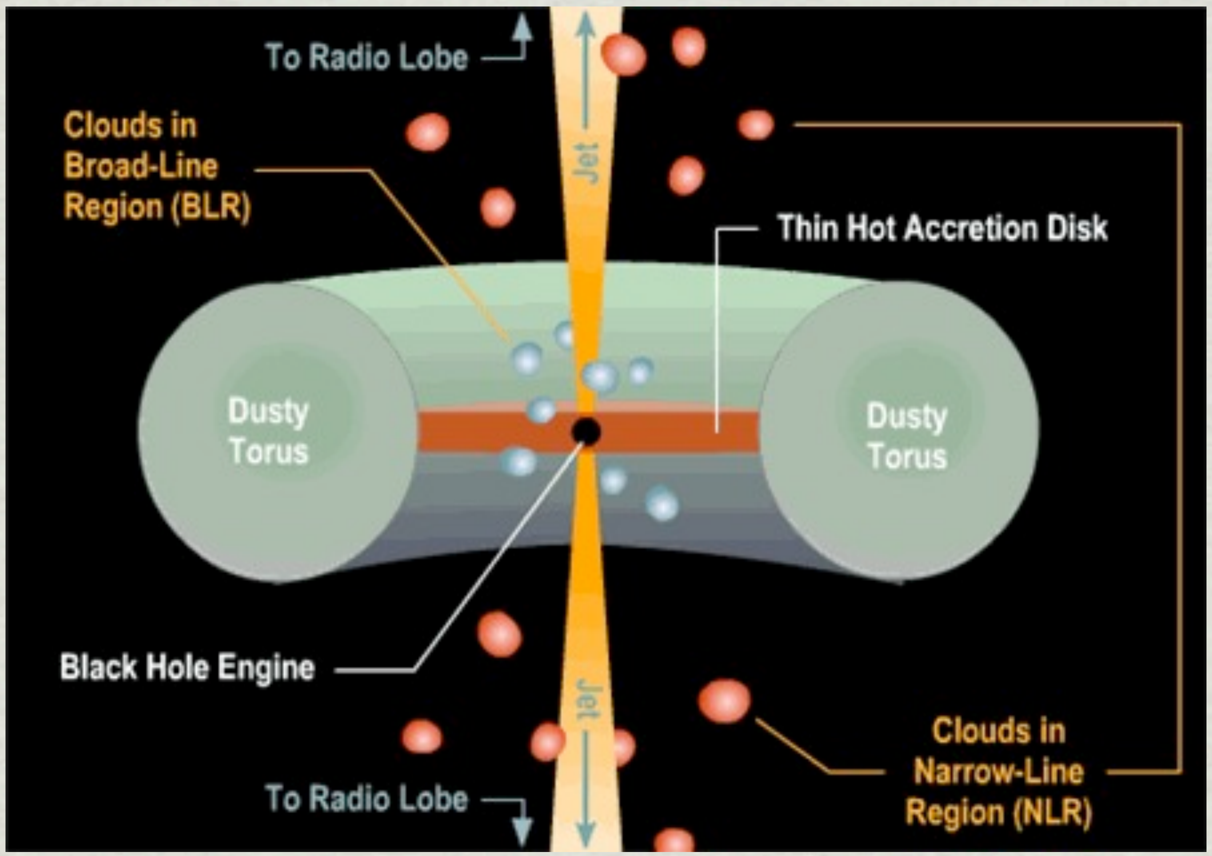


Illustration: Brooks/Cole Thomson Learning

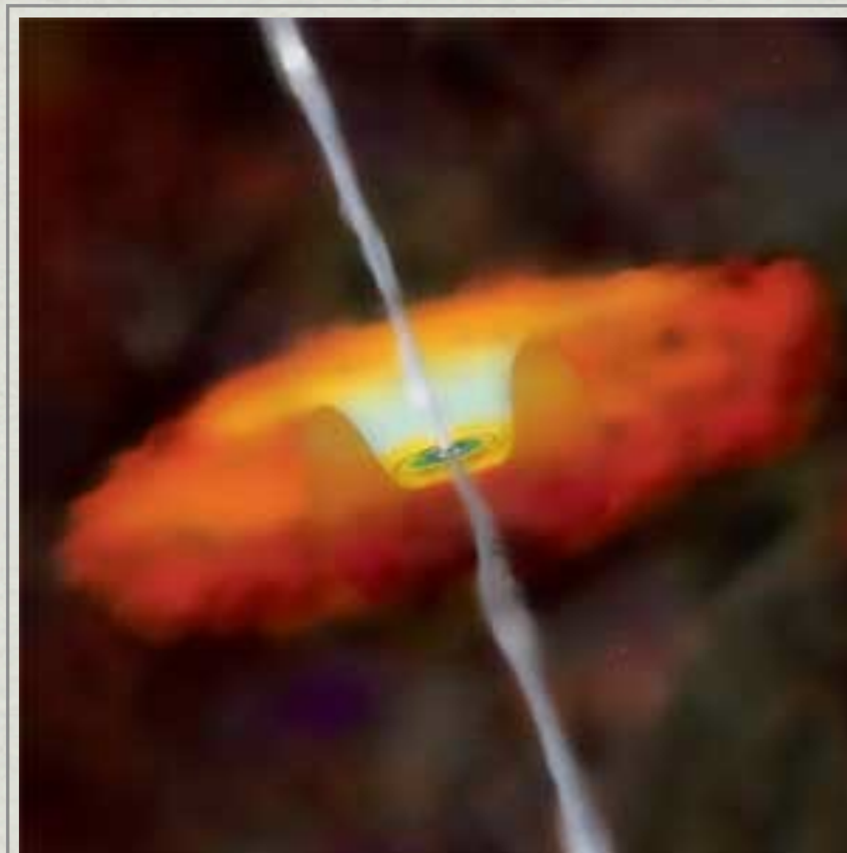
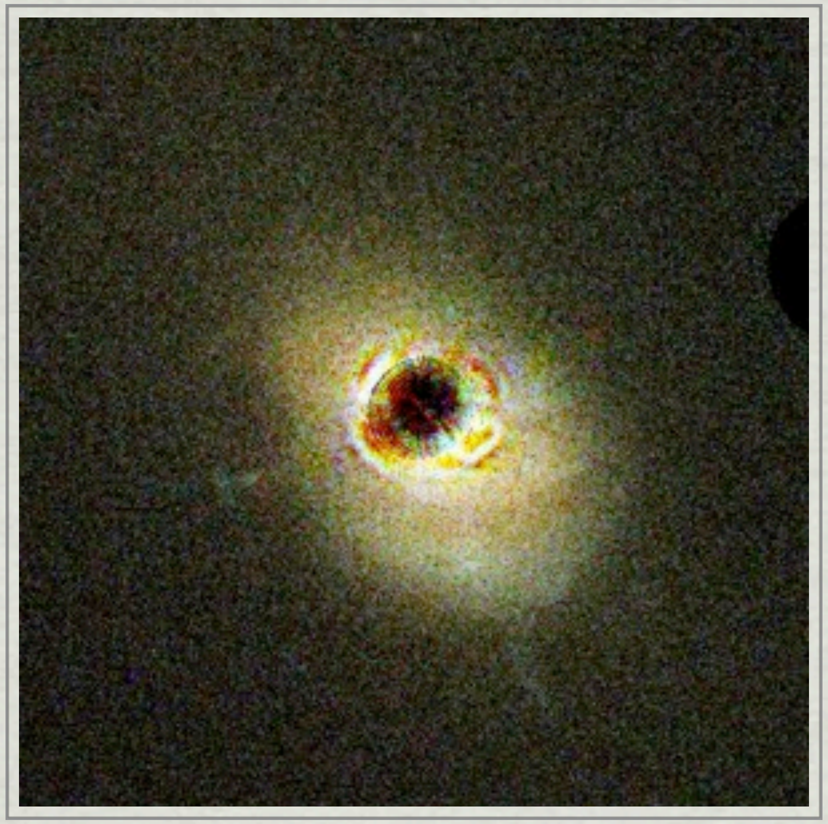


Illustration: Chandra X-Ray Observatory

- Torus is small, several parsecs in diameter. Surrounding galaxy is several *kilo*parsecs
- Can only image this directly for nearby galaxies

Point Source Subtraction

Original Hubble image

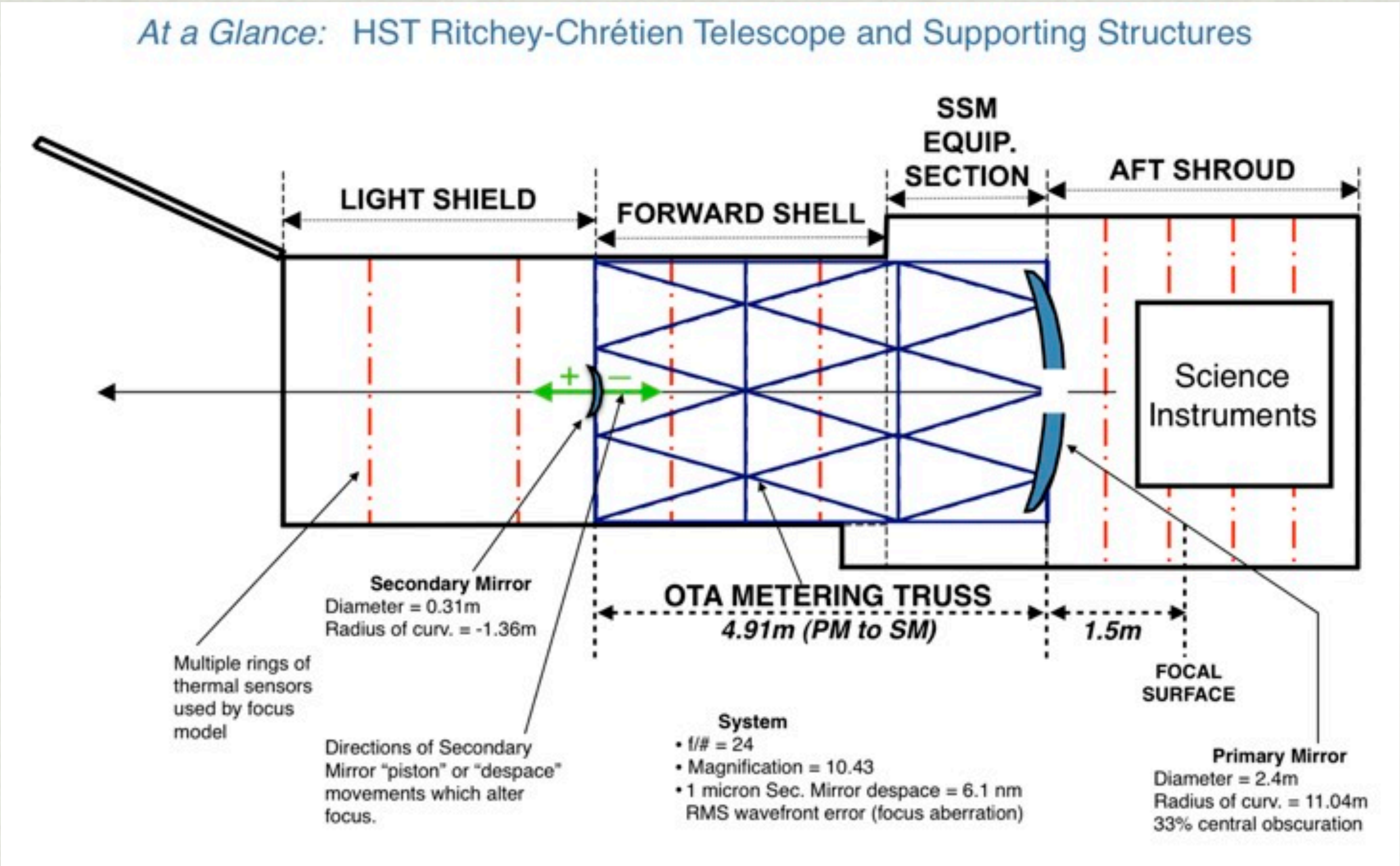


Quasar point source subtracted

Images: Hubblesite

- Lets us disentangle quasar and host galaxy light
- Can use empirical point source (star) or model
- Much harder for distant galaxies! Distant galaxies are small and would fit in the over-subtracted hole

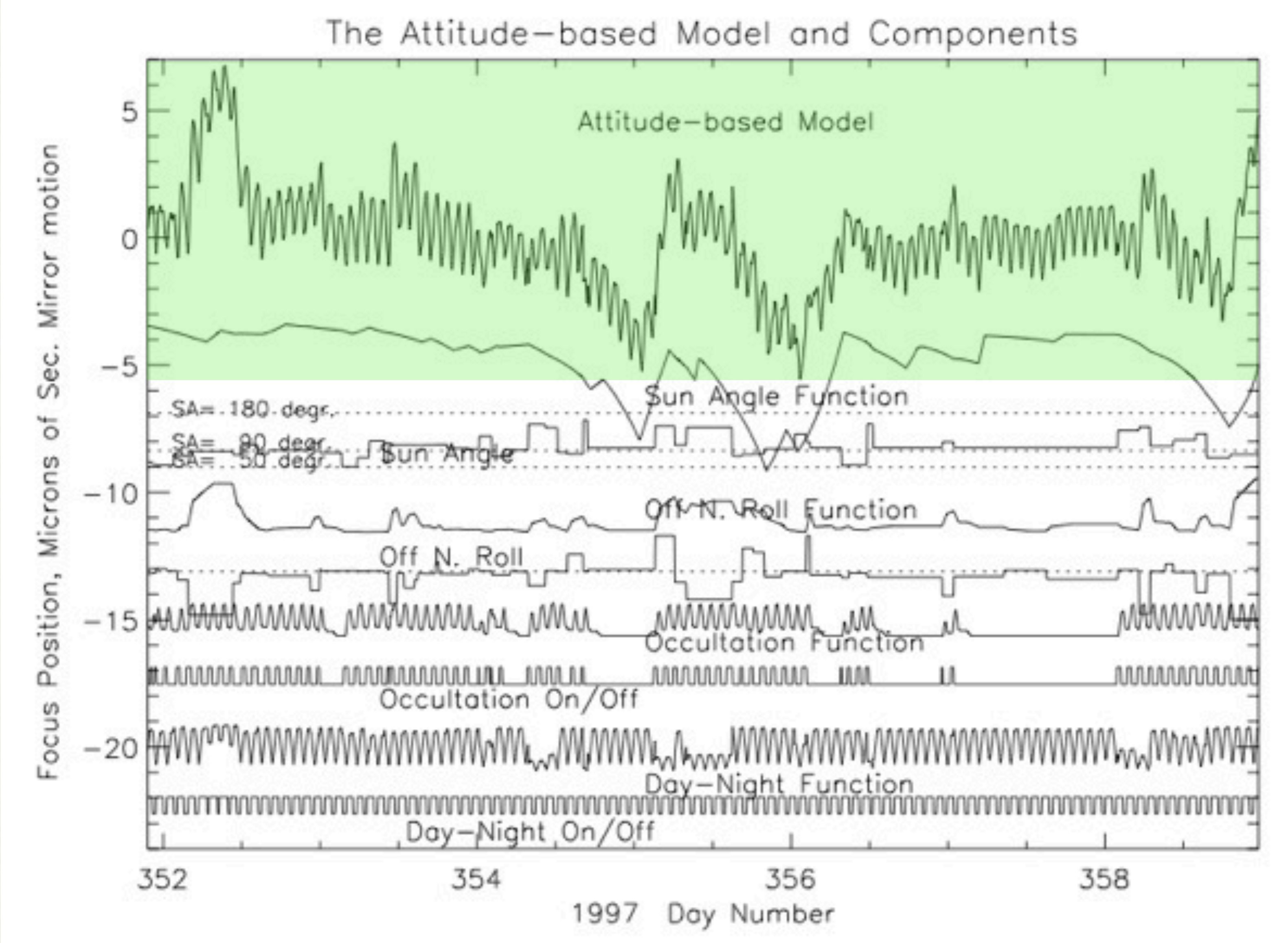
Spacecraft Breathing



- Secondary mirror truss moves, caused by changes in temperature

Spacecraft Breathing

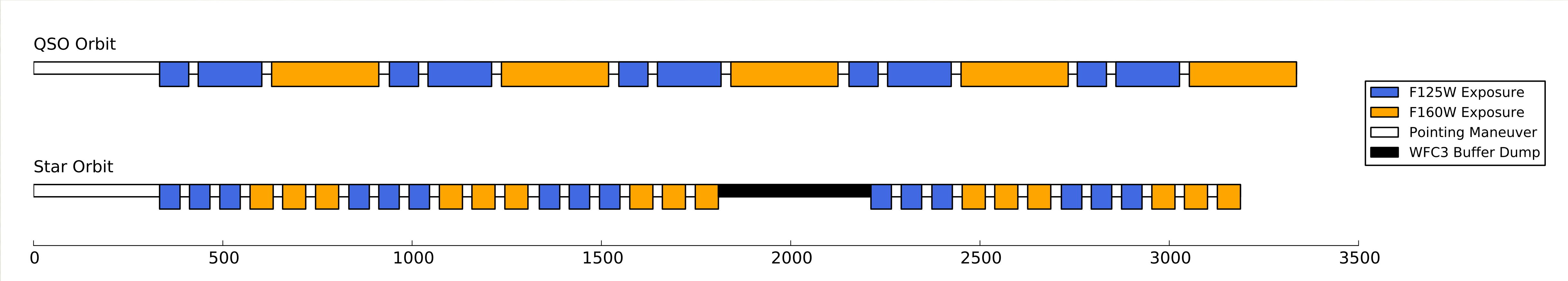
- Telescope focus changes with attitude and orbital day/night cycle
- Hershey (1998) and Cox & Niemi (2011) have modeled this quite well



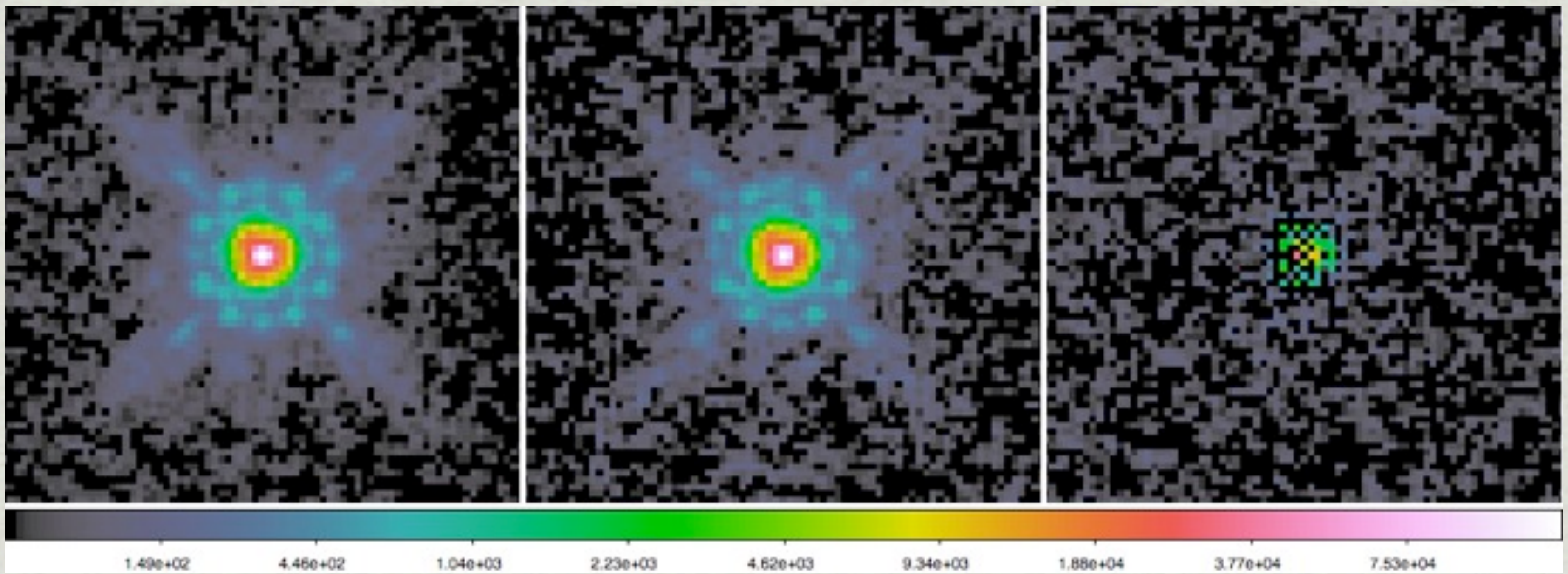
Hershey 1998

Ensuring Thermal Stability: A Solution!

- Pick a star nearby in the sky (to ensure similar telescope attitude)
- Observe in adjacent orbits
- Take matched exposures at the same point in orbital phase, to measure the PSF difference from day/night cycle



Let's Try Subtracting the Empirical PSF (Star)



Quasar **Star** **Residual**

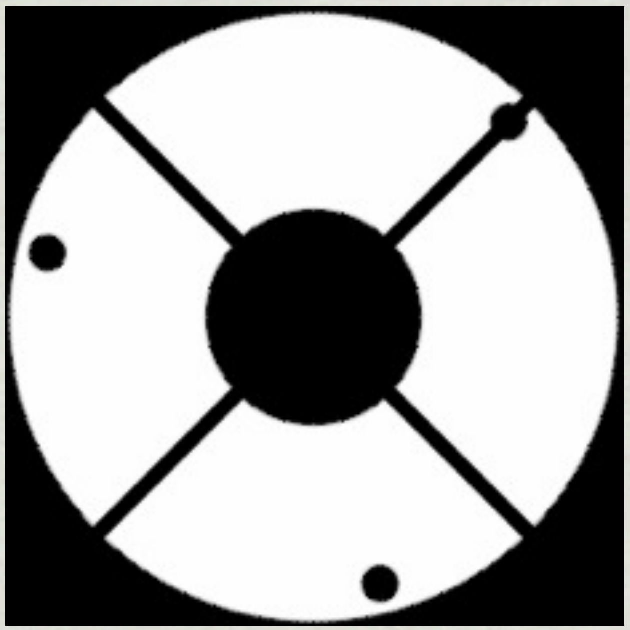
- We used GalFit (Peng et al. 2006) to fit the empirical PSF (x,y shift and normalization)
- Tantalizing numeric residual, but cosmetically hideous
- Would prefer a spatially oversampled PSF, without adding more pixel noise

30

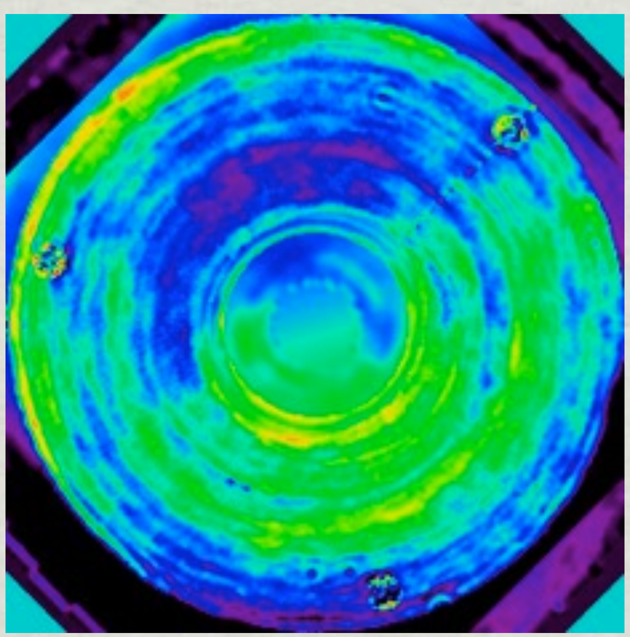
Theory of Fourier Optics (60 Second Version)

Wide Field Camera 3 Entrance Pupil

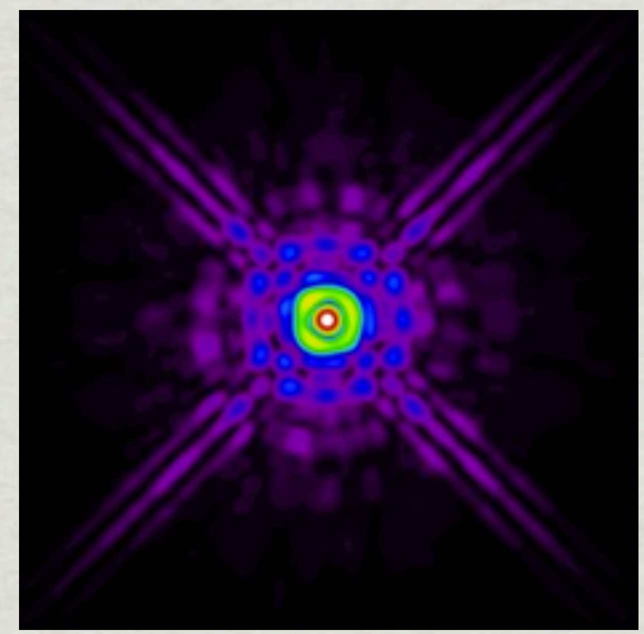
$\Re$ Part – Intensity



$\Im$ Part – Phase



Fourier Transform

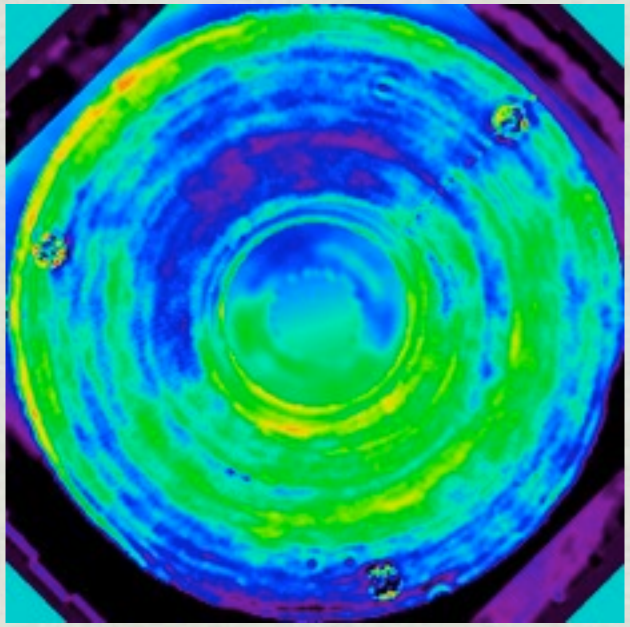


$\Re^2$ – Point Spread Function

TinyTim software (Krist et al. 2011) does this to model the Hubble PSF

31

Optimize by Varying Zernike Coefficients



* Wavefront error (phase difference) can be adjusted using these polynomials

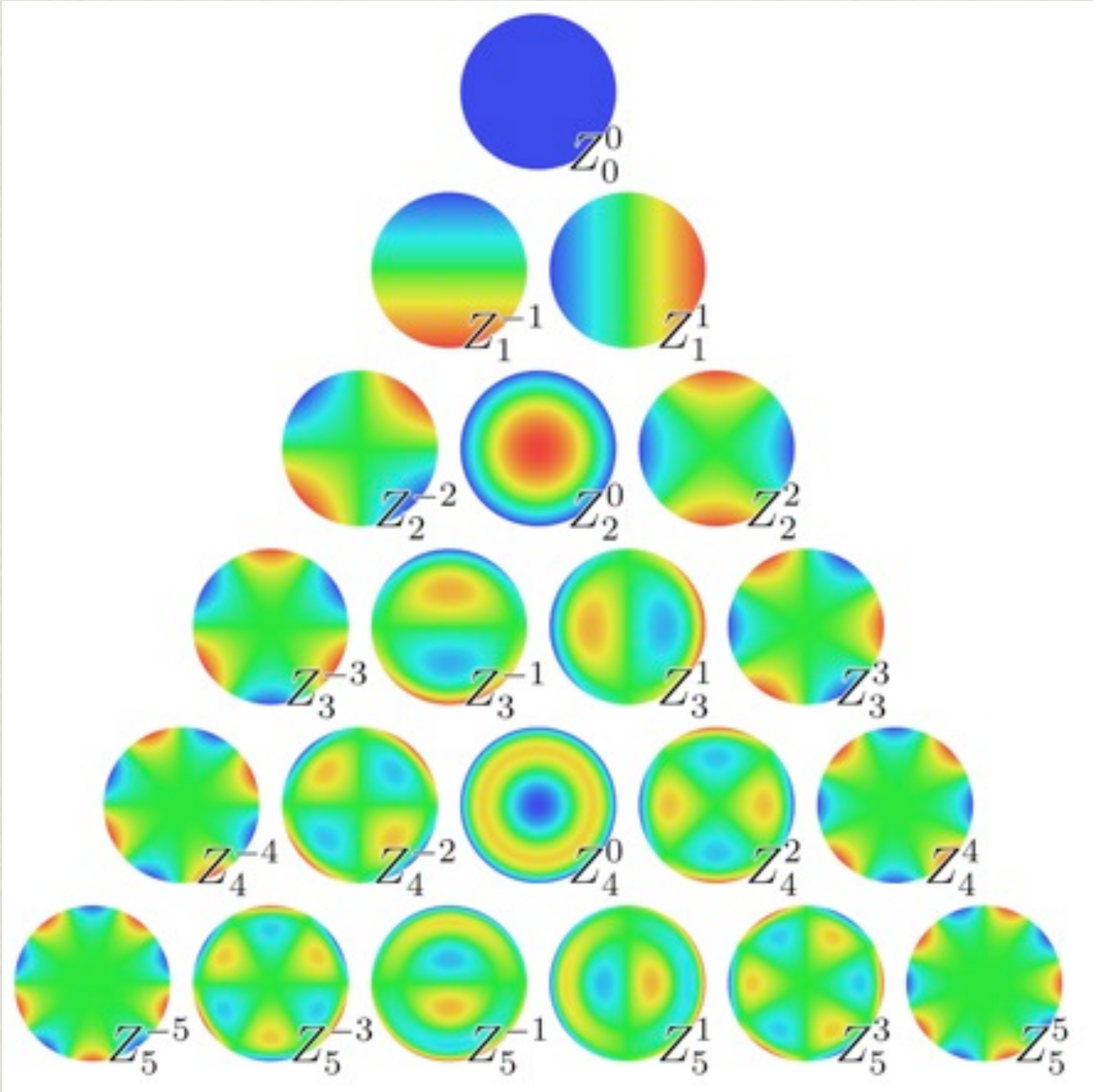


Image: Wikipedia

General Plan

1. Make model PSF that matches star image as well as possible

2. Subtract this model from the quasar

- For step 1, we need to generate many PSF models with varied aberration parameters, and minimize the fit residuals.

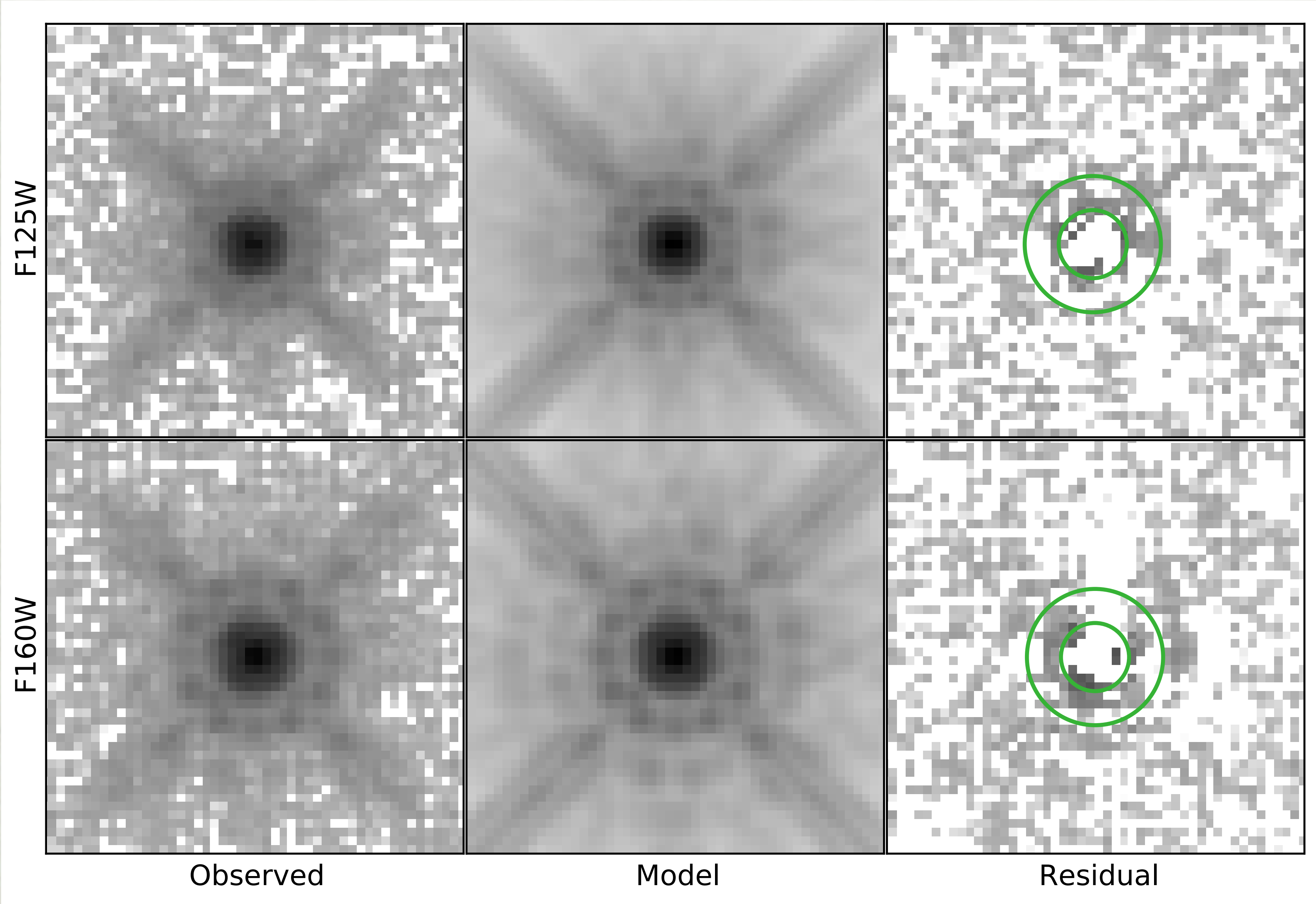
- What is the glue that will hold together TinyTim for modeling and GalFit for fitting? Who could it be...

Could it be... PYTHON?

- Simple grid of parameter space, not at all fancy
- ~10 minutes of programming (including one file not shown)

```
#!/usr/bin/python

import pyfits
import subprocess
from numpy import linspace, abs, zeros
import shutil
import itertools

pDict = dict()
pDict['x_astig'] = linspace(-0.02,0.02,7)
pDict['y_astig'] = linspace(-0.02,0.02,7)
pDict['x_coma'] = linspace(-0.1,0.1,7)
pDict['y_coma'] = linspace(-0.1,0.1,7)
pDict['focus'] = linspace(-8.00,-8.00,7) # Offset from ACS, in microns

bestChi2 = 2e15 # Something way too big

with open('stats.txt','w') as stats:
    stats.write(' '.join(pDict.keys())+ ' cdsigma chi2nu\n')

for params in itertools.product(*pDict.values()):
    stepDict = dict(zip(pDict.keys(),params))
    # Turn dict into k=v string
    clargs = ' '.join(['{0!s}={1!s}'.format(k,v) for k,v in stepDict.items()])
    # Generate PSF for these values
    subprocess.call('python ../TinyTim/mkPSF.py hTempl.par hexpsStar.txt '+clargs, shell=True)
    shutil.move('hPSFStar.fits','psf.fits')

    # Run galfit to determine how well it fits the data
    subprocess.call('galfit fit.feedme', shell=True)

    newChi2 = pyfits.getval('starFit.fits', 'CHI2NU', ext=2)

    with open('stats.txt', 'a') as stats:
        stats.write(' '.join(['%0.4f' % v for v in params])+' %0.4f %0.4f\n' % (sig, newChi2))

    # If it beat the last one, replace best fit/psf files
    if abs(newChi2-1) < abs(bestChi2-1):
        bestChi2 = newChi2
        shutil.move('starFit.fits', 'bestFit.fits')
        shutil.move('psf.fits', 'bestPSF.fits')
```

Optimized PSF

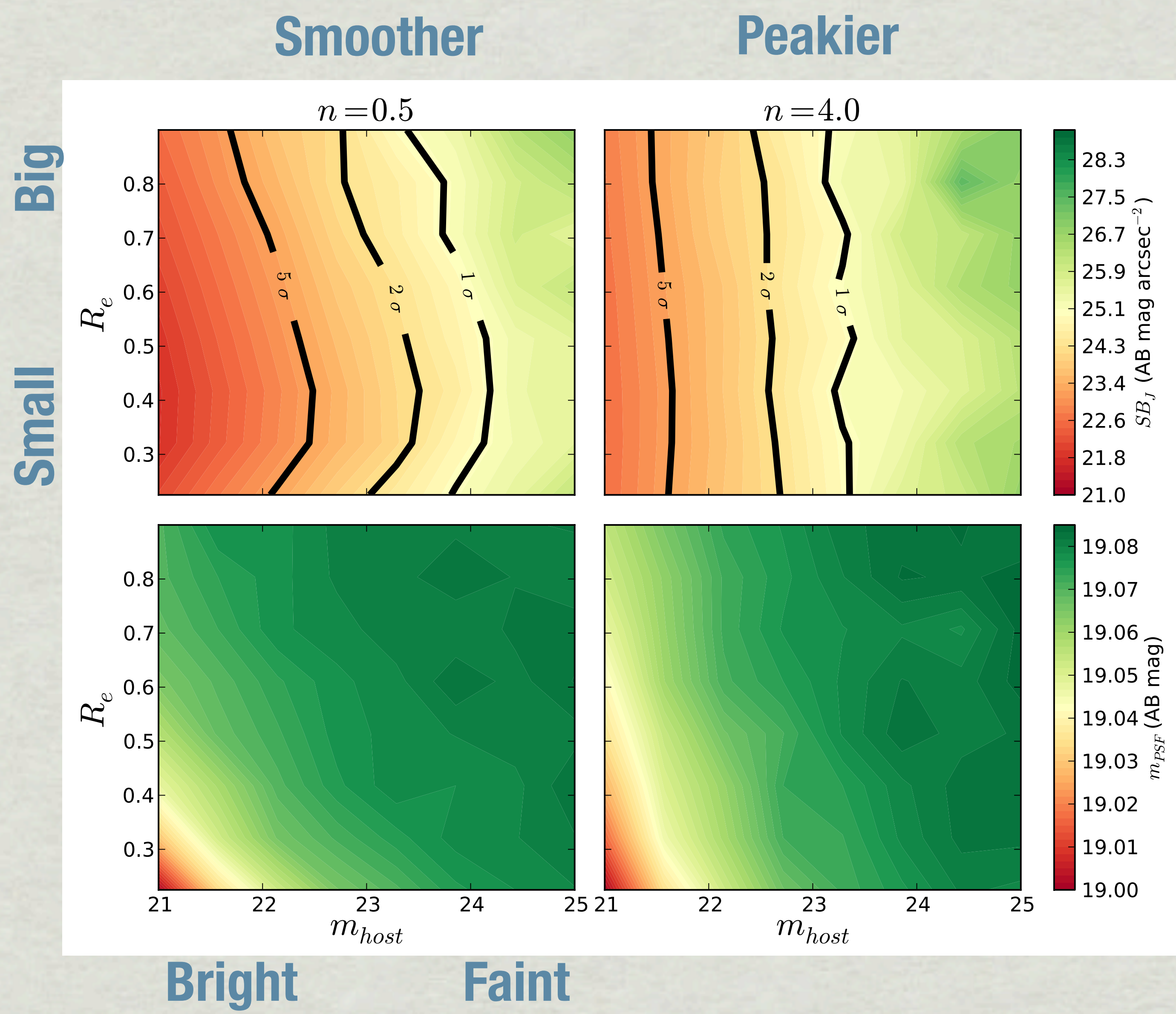


- A marginal detection (upper limit) like the empirical subtraction, but improved S/N (even after accounting for errors in the model)

35

Reliability Modeling

- Make model galaxies + point source, do our subtraction
- Determines our ability to recover host galaxy light, as a function of the host galaxy's parameters
- Varied radius, brightness, Sérsic index ("peakiness")

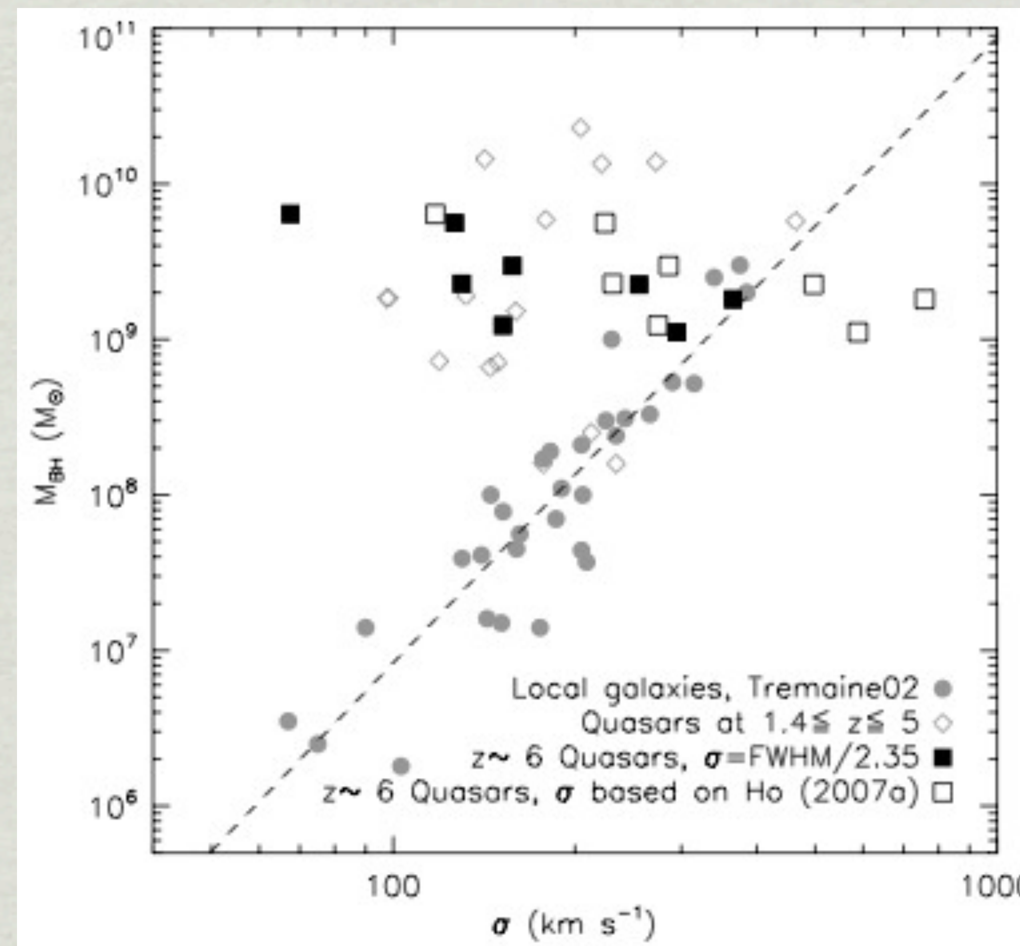


What Does This Tell Us?

- Galaxy seems too small for its black hole, if the local relation holds. Maybe black holes form first?

- Star formation rate consistent with that measured from heated dust – ~2000 M$_\odot$/yr (Milky Way is ~1 M$_\odot$/yr)

Black Hole Mass



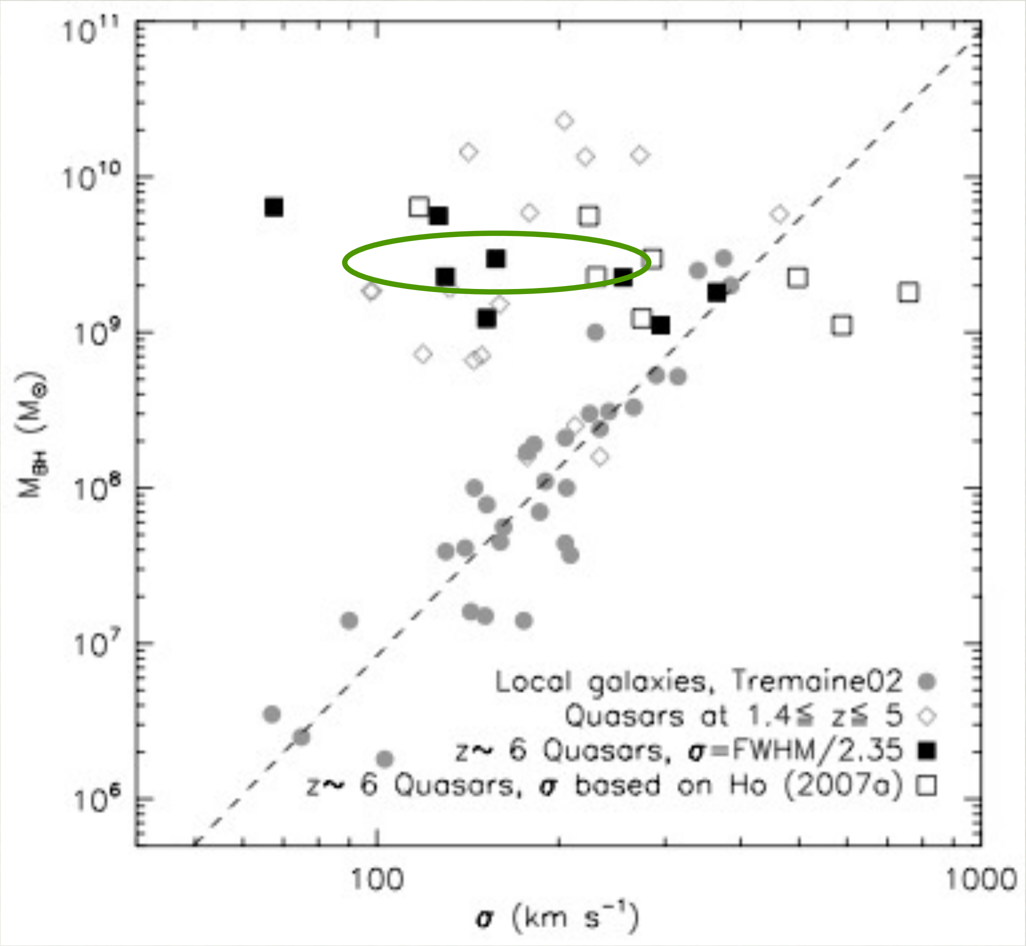
Wang et al. 2010

Velocity Dispersion
(Proxy for Galaxy Mass)

What Does This Tell Us?

- Galaxy seems too small for its black hole, if the local relation holds. Maybe black holes form first?
- Star formation rate consistent with that measured from heated dust – ~2000 M⊙/yr (Milky Way is ~1 M⊙/yr)

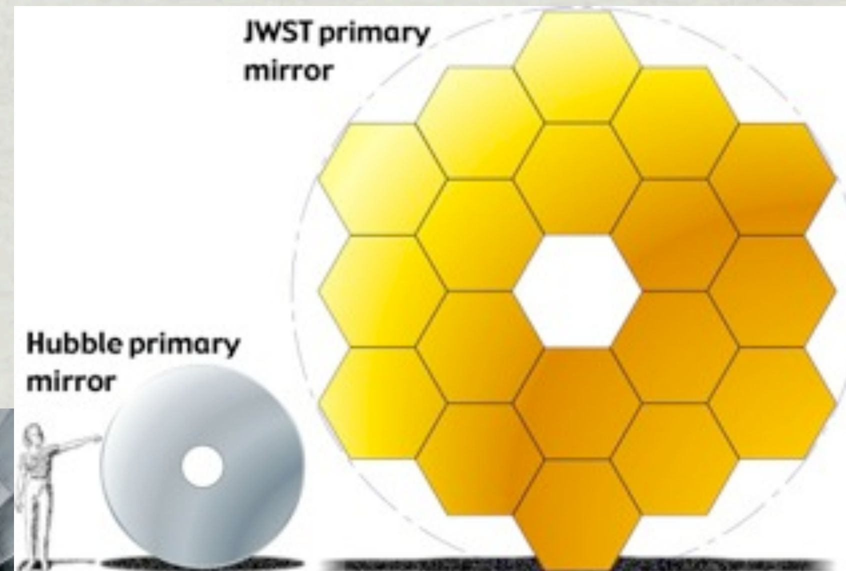
Black Hole Mass

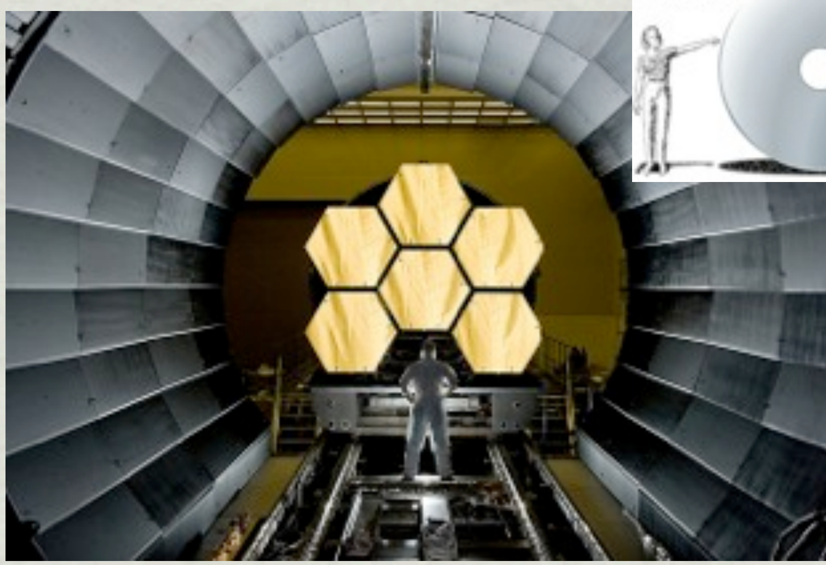


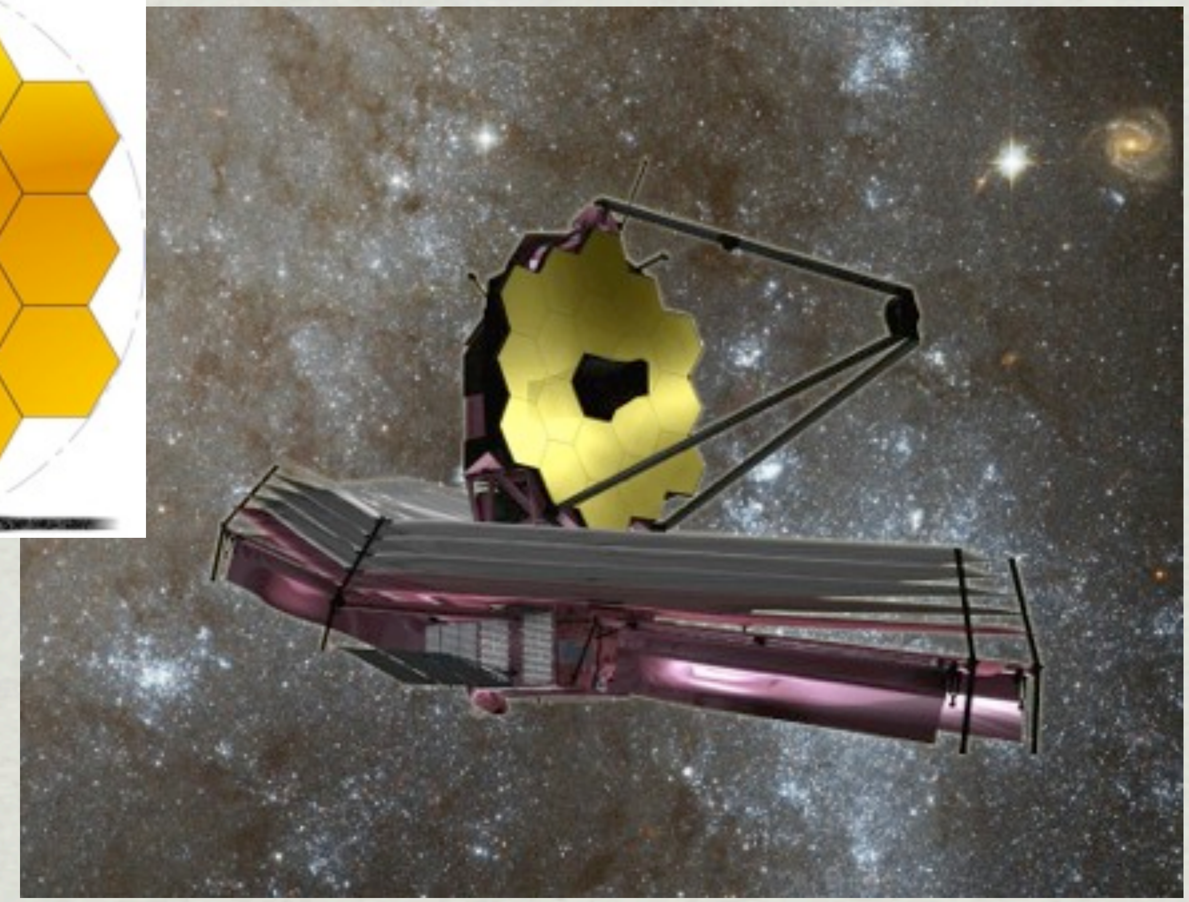
Wang et al. 2010

Velocity Dispersion
(Proxy for Galaxy Mass)

The Future:
James Webb Space Telescope



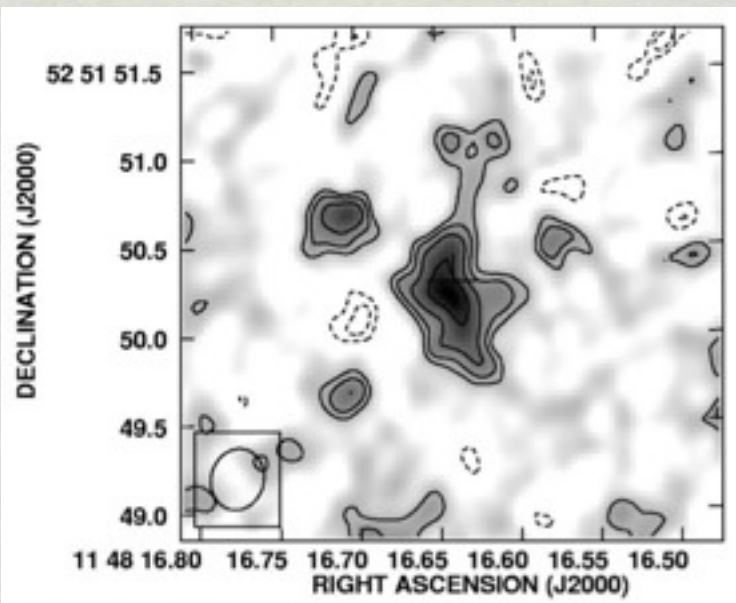




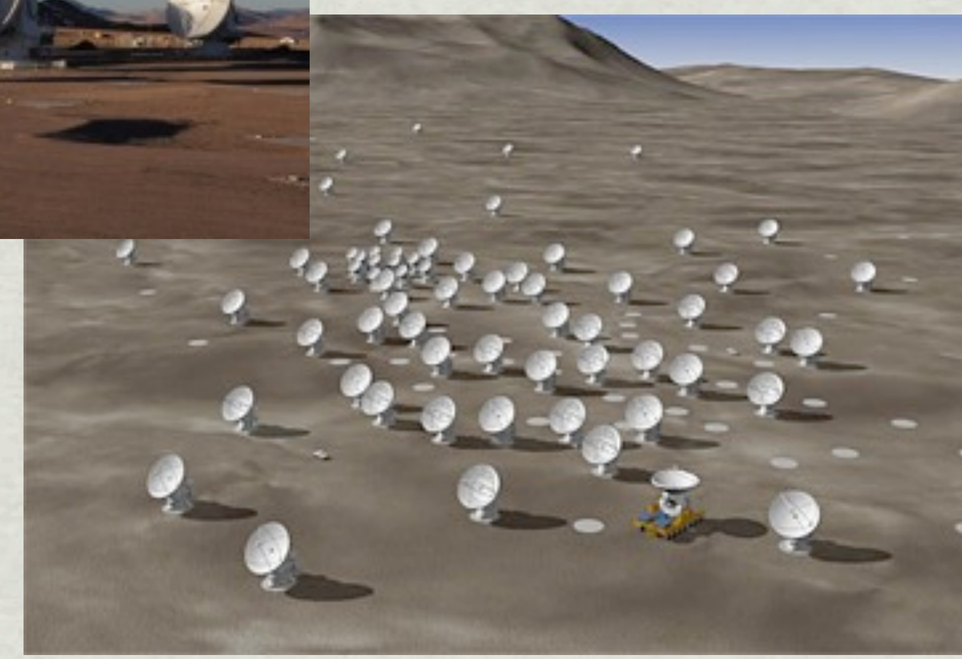
- 6.5 meter infrared-optimized space telescope
- Launch in ~2018

38

The Future: Atacama Large Millimeter Array



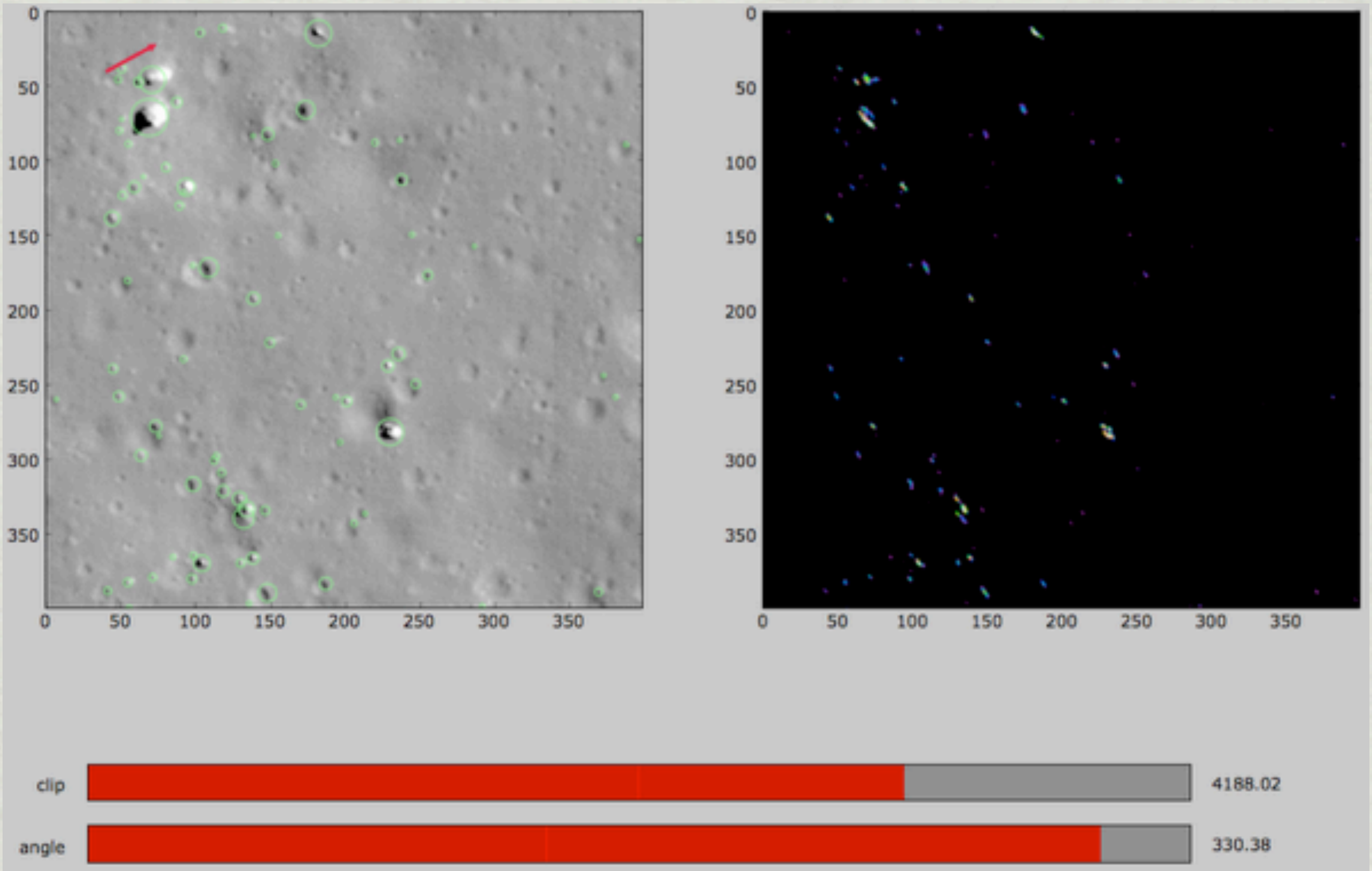
CO emission from quasar host galaxy Walter et al. 2004



- Millimeter-wave radio telescope array in Atacama Desert, Chile. ~50 12-m antennas when complete
- Better sensitivity and site than any other radio telescope

39

Bonus Pretzel: Finding Rocks on the Moon



- Uses scipy.ndimage for image thresholding and gradients. Matplotlib widgets for interactive parameter tuning!

Conclusions

- Use (or make) tools that enable rapid iteration!
- Optimize for your time programming, as well as the usual things
- HIPPIES: There may be more very bright galaxies than expected at redshifts z>7
- Quasars: The most distant quasar host galaxies may be fainter than expected, or *very* compact. Future observations with JWST and ALMA should tell us for sure!