

The background of the slide is a dark blue technical drawing or blueprint. It features various geometric shapes, including circles, arcs, and lines, rendered in a lighter blue and yellowish-green color. The drawing appears to be a schematic of a large-scale engineering project, possibly related to the LSST telescope. The lines are thin and precise, typical of a technical drawing.

# DM Plans for Crowded Stellar Fields

Jim Bosch, Data Release Production Scientist

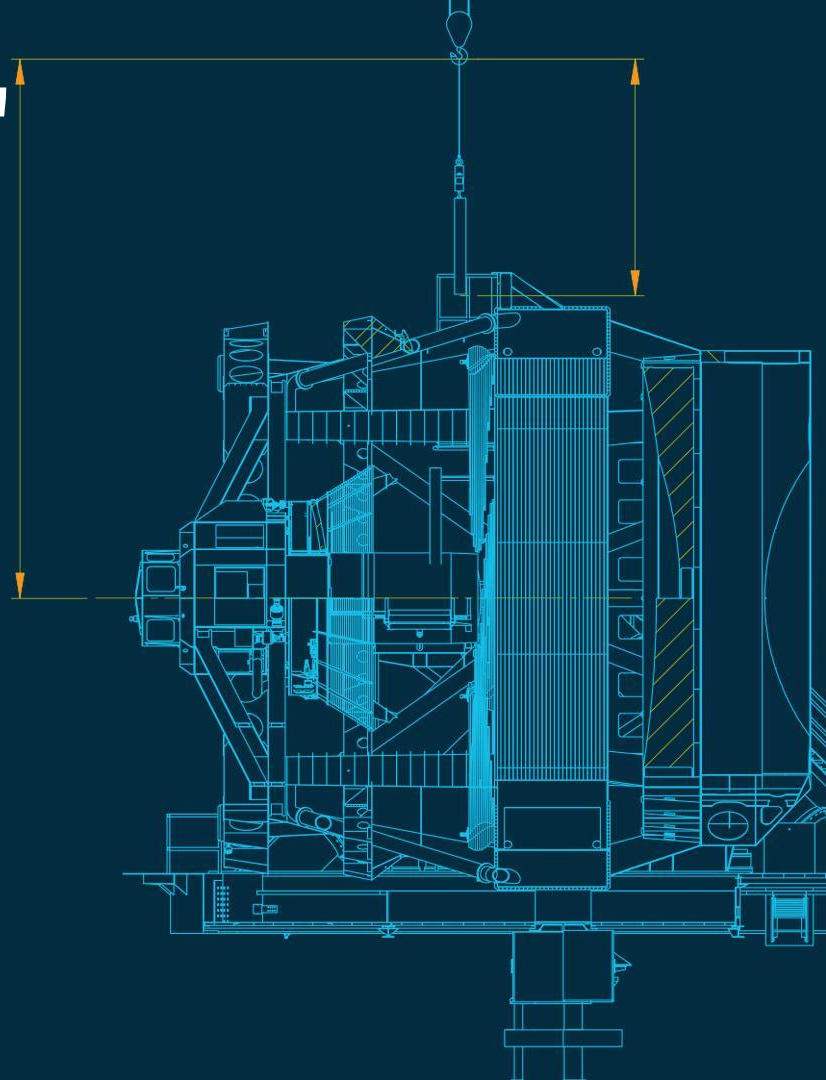


*Large Synoptic Survey Telescope*

# What does "crowded" mean?



*Large Synoptic Survey Telescope*



sources/arcsec<sup>2</sup> \* psf effective area ~ 0.044





$\text{sources/arcsec}^2 * \text{psf effective area} \sim 0.075$



sources/arcsec<sup>2</sup> \* psf effective area ~ 0.187







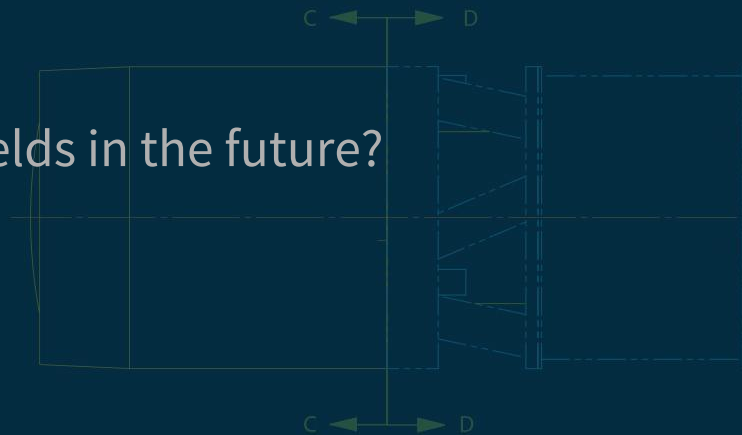
# Overview

Requirements and Plans: the Big Picture.

Testing, Goals, and Metrics: how well are we actually doing?

Algorithms:

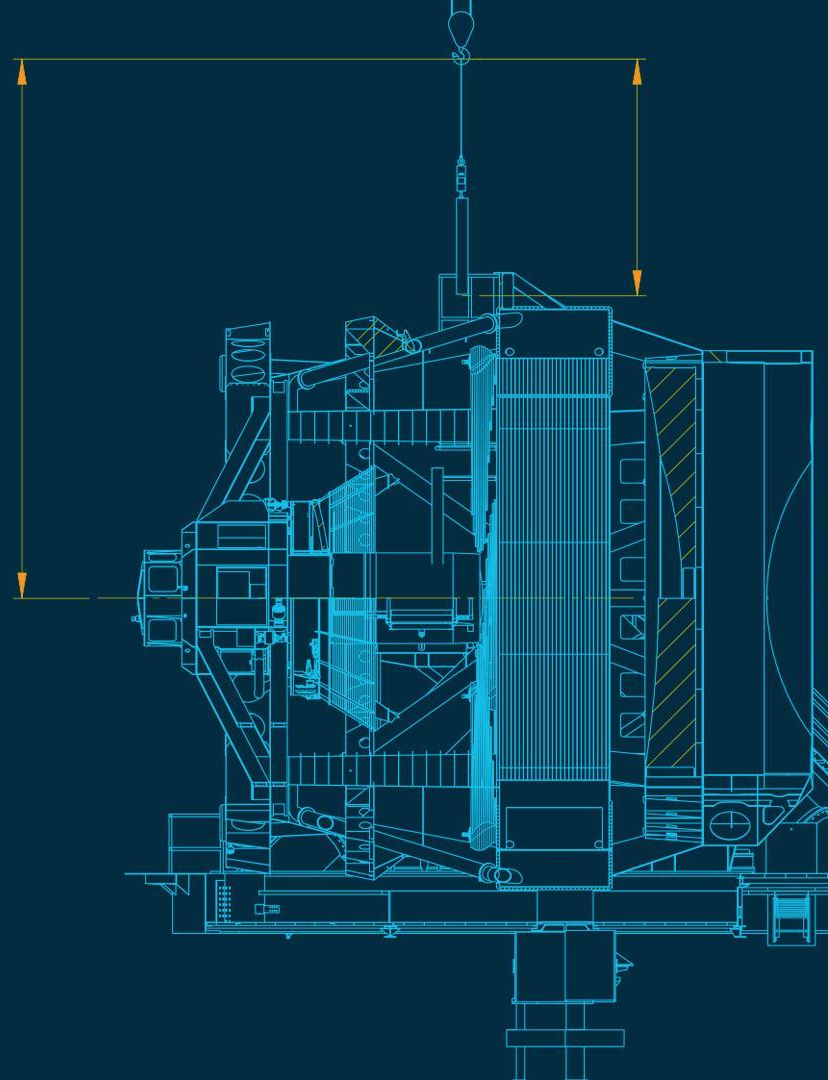
- what does our pipeline do now?
- how will we better support crowded fields in the future?



# Requirements



*Large Synoptic Survey Telescope*





# Requirements



All of DM's work on crowded fields is on a *best effort* basis.

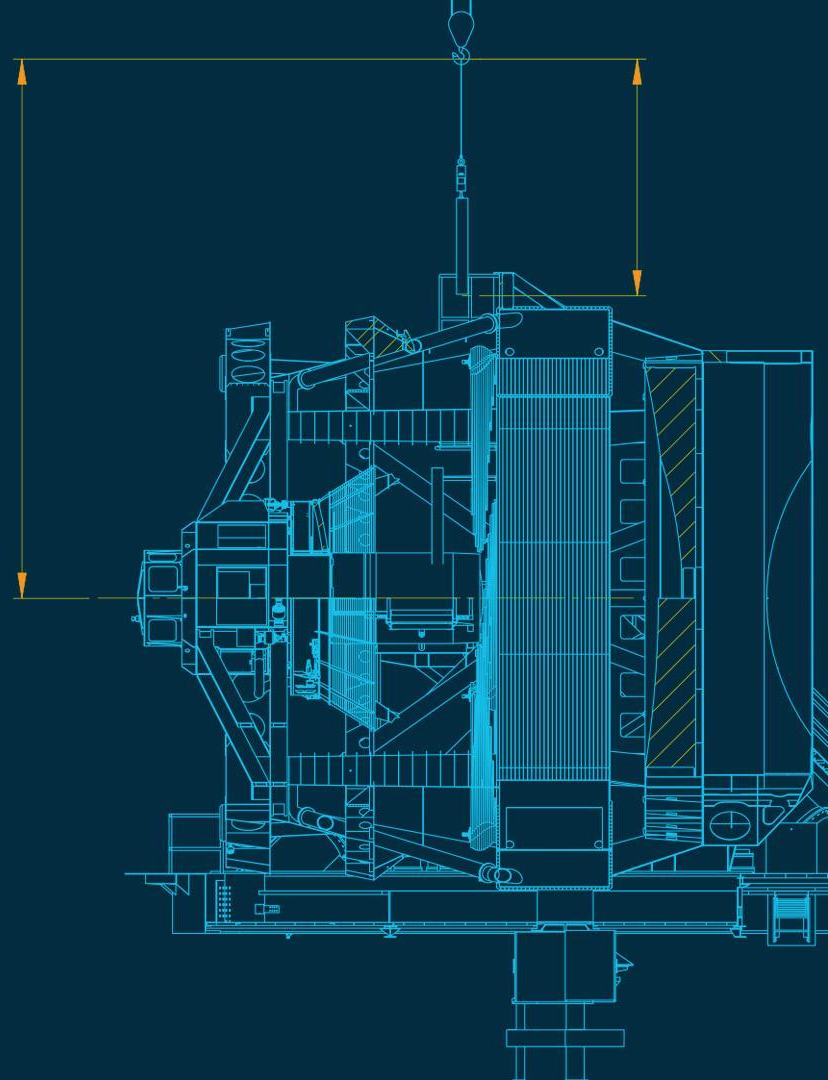
A major replan of the remaining DM construction work is close to being completed; one of open issues is the formulation of quantitative performance metrics and goals for crowded field processing.

Science Collaborations can greatly help with these metrics and goals.

# Plans



*Large Synoptic Survey Telescope*



# Plans



Our baseline design is to use essentially the same algorithms to process high-latitude fields and crowded stellar fields.

- It's at least plausible that this will let us process *very* crowded fields; the design combines concepts from existing pipelines for high-latitude images with concepts from existing crowded field codes.
- This is a novel approach, and hence it's impossible to guarantee it will work.
- With an infinite compute budget, I'm pretty sure it would work. With our finite one, it's much less clear.



# Plans

We are also evaluating processing crowded stellar fields with specialized code.

- There is plenty of prior art if we don't have to worry about galaxies at all.
- Can we really say we don't need to worry about galaxies at all? They don't get less common just because there are more stars, and LSST will be a lot deeper than previous surveys.
- Specialized code means processing intermediate regions multiple times, and that means bigger compute costs and more complicated databases.

# Plans



We do *not* plan to use an existing third-party code even if we do specialized processing of crowded fields.

- The LSST pipelines already have many of the algorithmic components we need: PSF models, background subtraction, detection algorithms, model-fitting. Retrofitting a third-party code to use these components may be more difficult than using them to reimplement the same algorithm.
- We do not believe existing third-party codes can run effectively at this scale without human intervention.
- We do not know if existing third-party codes will be fast enough.

# Testing, Goals, and Metrics

Since we can't promise anything about how well we *will* do, help us learn, track, and report how well we *are* doing.



*Large Synoptic Survey Telescope*





# Test Data

Part of the reason DM's stellar field plans are vague is that we don't regularly run our pipelines on stellar field data.

- We're aware of some datasets (e.g. Schlafly et al DECam plane survey, HSC satellite galaxy searches).
- We have not done the work to identify and package "bite size" subsets of the data for regular testing.
- We need test data that spans (and samples well) the levels of crowding LSST will see.
- We need test data that goes as deep as LSST.
- We need data we can process: DECam, HSC, or CFHT.

# Goals



DM will be trying to better define its goals and priorities for crowded fields over the next ~3 months, and we'd appreciate input from science collaborations. For example:

- What level of crowding should we focus our effort on? How do we weigh better processing of moderately crowded fields against minimal processing of extremely dense fields?
- How important are variability and astrometry compared to static-sky photometry?
- What metrics (and goals) should we have for completeness when it can't be described by a magnitude limit?

# Metrics

It's easiest for us if these goals are defined as metrics we can use to test our processing. Ideally, a metric includes:

- A test dataset.
- One or more numbers that can be measured from the output catalog that relate to the quality of the processing (e.g. width of the main sequence in some color-color space).
- A sequence of goals for those sets of numbers.

It would be very interesting to define these goals from the outputs of existing crowded field codes.



# Contributing, Part 1

Help with any of these steps would be great.

- Identify and package up a test dataset.
- Try running the DM stack on a test dataset.
  - If it's more than a little crowded, it'll probably at least require some configuration-tweaks to get it running right now.
  - If it's very crowded, it may not run at all right now, or the results may be complete garbage. But that too is good for us to know.
- Run an existing crowded field code on a test dataset.
- Try to define some specific metrics to test the quality of the processing.

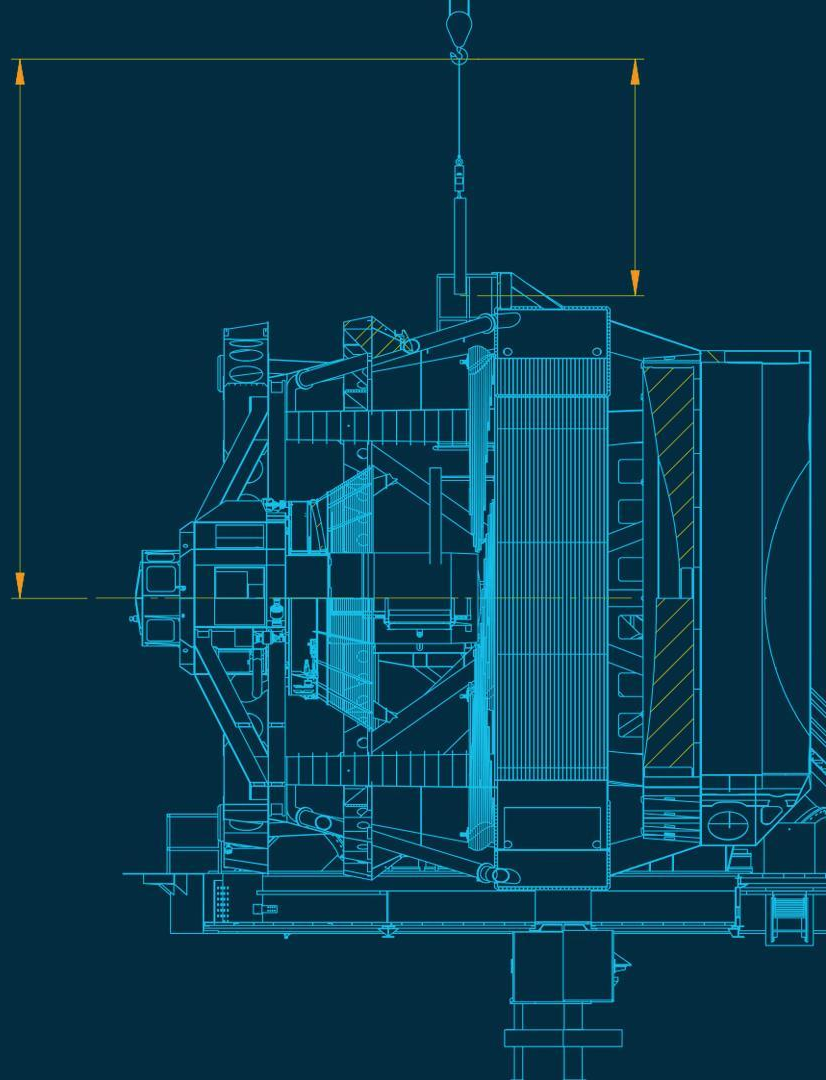
# Communicating



DM developers will be doing this kind of work, too.

- We can help (via [community.lsst.org](https://community.lsst.org) and Slack) with running the DM stack.
- We'd like to hear what you're planning to work on before you spend too much time on it, so we don't duplicate work. We'll try to find a way to make our testing plans public as well.
- We'll be identifying a point person or group to receive more structured input on metrics.

# Algorithms



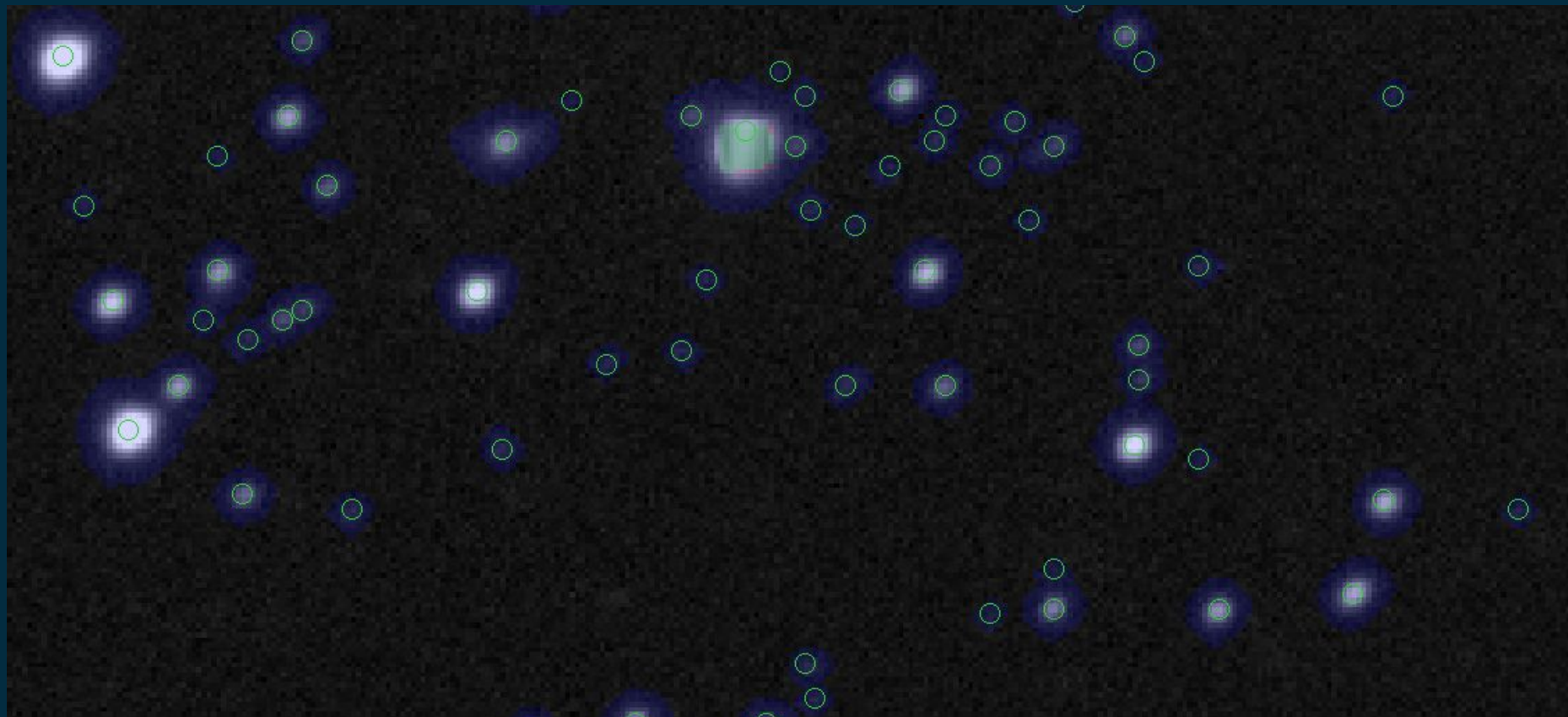
# The LSST Pipeline Today

Very roughly, the steps of the pipeline are:

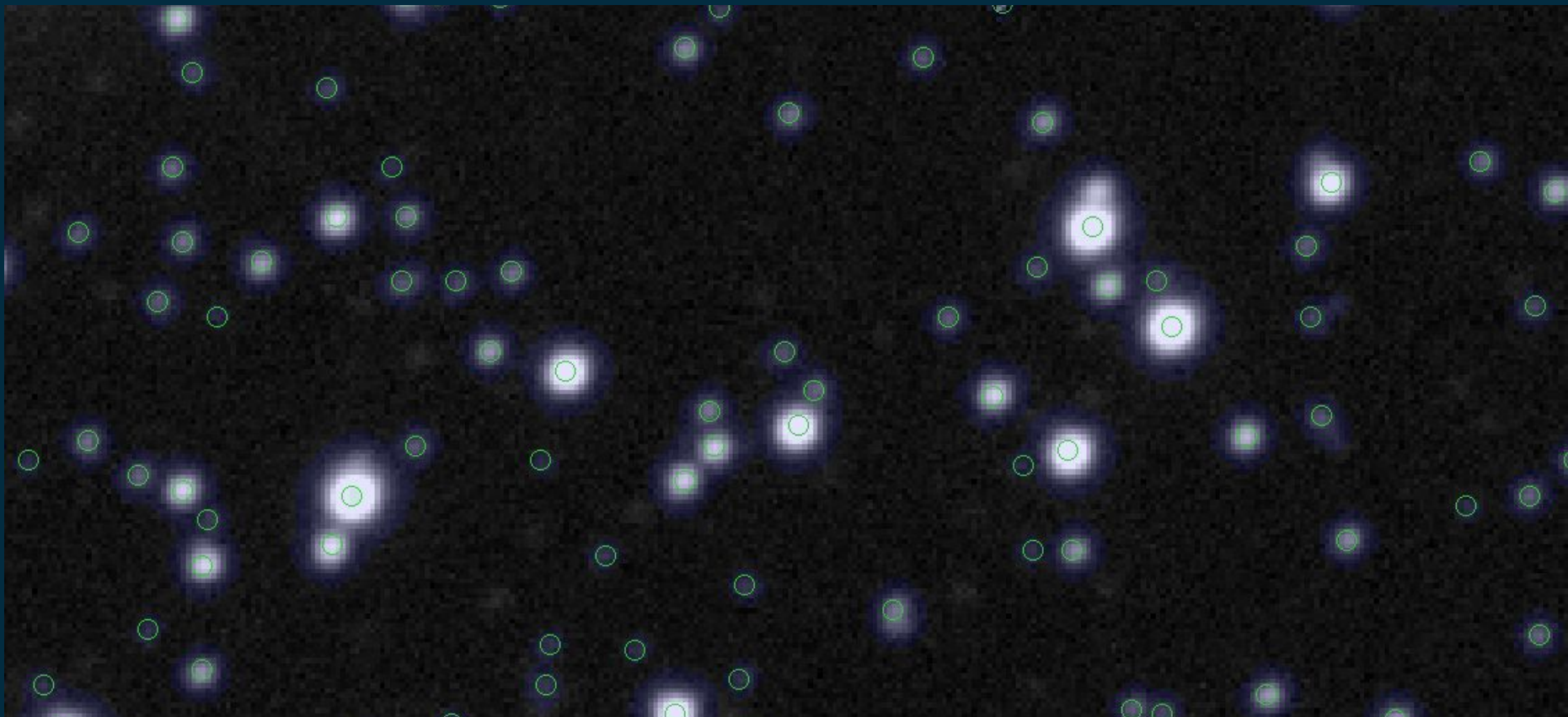
- We process individual CCD images to detrend, measure PSF models, fit initial WCS and photometric calibration.
- We do a joint fit to all of the per-CCD catalogs in a region of sky to improve the WCSs and photometric calibrations.
- We build coadds.
- We detect, deblend, and perform direct and forced measurements on the coadds.



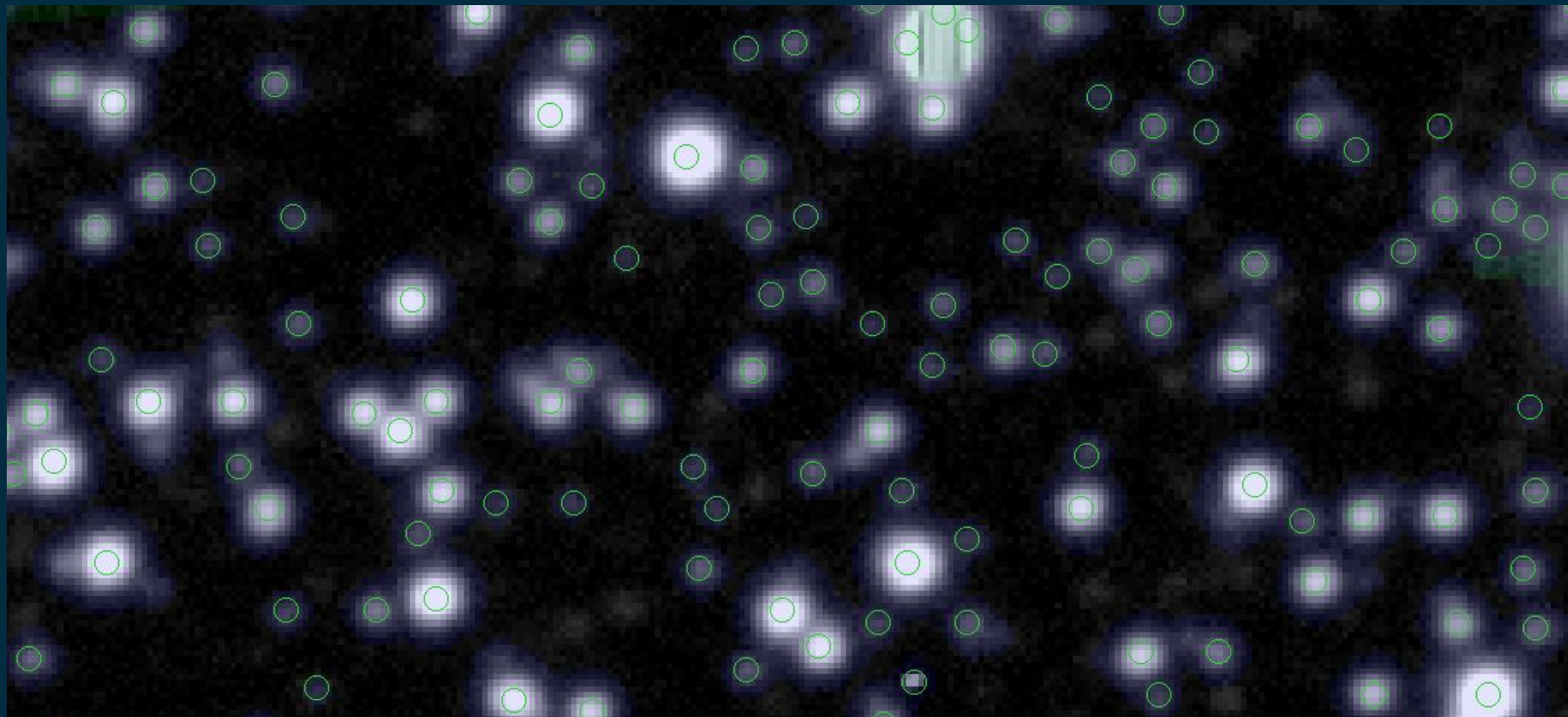
$\text{sources/arcsec}^2 * \text{psf effective area} \sim 0.044$



sources/arcsec<sup>2</sup> \* psf effective area ~ 0.075



$\text{sources/arcsec}^2 * \text{psf effective area} \sim 0.187$





# The LSST Pipeline Today

Very roughly, the steps of the pipeline are:

- We process individual CCD images to detrend, measure PSF models, fit initial WCS and photometric calibration.
- We do a joint fit to all of the per-CCD catalogs in a region of sky to improve the WCSs and photometric calibrations.
- We build coadds.
- We detect, deblend, and perform direct and forced measurements on the coadds.



# The LSST Pipeline in the Future

Very roughly, the steps of the pipeline are:

- We process individual CCD images to detrend, measure PSF models, fit initial WCS and photometric calibration.
- We do a joint fit to all of the per-CCD catalogs in a region of sky to improve the WCSs and photometric calibrations.
- We build coadds.
- We detect, deblend, and perform direct and forced measurements on the coadd.
- **We simultaneously fit one model for each object to all of the CCD images it appears on.**

# Coadd Processing Today

- Detection: find above-threshold regions and peaks within them on the coadd.
- Deblend: separate blended objects via a simultaneous fit with purely empirical models (which can just be the PSF).
- Measure: replace neighbors with noise using Deblend results, then measure each object separately.

# Coadd Processing in the Future



- Detection: find above-threshold regions and peaks within them on the coadd.
- Deblend: separate blended objects via a simultaneous fit with purely empirical models (which can **be encouraged to** just be the PSF).
- **Subtract the brightest stars from coadds, and go back to Detection until we converge.**
- Measure: replace neighbors with noise using Deblend results, then measure each object separately.
- **Fit blended objects simultaneously with PSF and galaxy models (maybe).**

# Big Algorithmic Questions



- Can we do PSF modeling and astrometric registration in severe crowding?
  - We can probably use image differencing techniques to iteratively improve the PSF.
  - We can probably select just the brighter (maybe even saturated) stars for astrometric registration.
- How does the deblender scale with the number of objects and pixels in the blend?
  - We already handle galaxy clusters with 50+ objects (not well, but galaxies are much harder than PSFs).
  - Prototype deblender in development already uses sparse matrices.



# Big Algorithmic Questions



- Can we do multifit in crowded fields?
  - We probably don't want to fit galaxy models at all, but we need to decide when to turn them off.
  - We're already considering fitting moving point-source models to all objects in small blends simultaneously; does that scale to large blends?

# Contributing, Part 2

Help us build a crowded field code from our algorithmic components. Starting with our PSF modeling and detection code, it'd be pretty easy to write Python scripts to:

- Detect objects.
- Simultaneously fit all detected peaks as point sources (with e.g. a SciPy sparse matrix solver).
- Iterate.

The learning curve for developing in the DM stack is steep; it might be worthwhile to spend a week at Princeton or UW.

# In Conclusion

There's a lot of uncertainty right now, but we will start putting some bounds on how well we'll do very soon.

Predicting how well we'll handle the hardest cases may always be impossible.

There are ways to contribute both priorities and algorithms right now, especially if you're willing to spend some time learning to use the DM stack.

