Data Management Subsystem Status

Kian-Tat Lim
Data Management Project Engineer
State of DM

• People
• Software
• Hardware
• Tools and Processes
• Relationships
• Future Plans

whitehouse.gov
People
Staffing Up

• 1/3 increased time from existing staff
• 1/3 existing staff not on LSST
• 1/3 new hires
"Management" (Servants)

- Project Engineer
- Deputy System Architect: Tim Jenness

Project Engineer (starting mid-May) responsible for technical management to build Data Management System
Princeton -74.651620

- Vishal Kasliwal (Sep)
- Nate Lust
- Lauren MacArthur
- Peter Melchior (Nov)
- John Swinbank
NCSA - 88.2248645

- Jason Alt
- Matias Carrosco-Kind
- Hsin-Fang Chiang
- Margaret Gelman
- Don Petravick
- Paul Wefel

Need more middleware software hires; Matias only 1/4 time
Tucson -110.9479262

- Frossie Economou
- Angelo Fausti
- Josh Hoblitt
- David Nidever
- Jonathan Sick
- Contractors
IPAC -118.127601

- David Ciardi
- Tatiana Goldina
- Steve Groom
- Loy Li
- John Rector
- Trey Roby
- Xiuqin Wu
- Lijun Zhang
SLAC -122.203603

- John Gates
- Fabrice Jammes
- Fritz Mueller
- Nate Pease
- Andy Salnikov
- Brian Van Klaveren
UW -122.311400

- John Parejko (Oct)
- David Reiss (Sep)
- Colin Slater (Sep)
- Ian Sullivan

Still missing algorithms lead scientist
Worries and Responses

• Hiring has been slow, particularly for lead roles

• Reorganize work with delayed staffing, relieve current leads of less-critical tasks
Software
Overview

- Winter 2015 (10.1) and Summer 2015 (coming in Sep) releases
- Refactor design prototypes into maintainable production code
- Integrate components "end-to-end" including OCS
- Incorporate best-of-breed algorithms (most developed on HSC data and funding)

Technical debt from back-ports, but very soon will have HSC and LSST on same code base
Middleware and Control

- Alert Production simulation
- OCS communication demonstration
- Logging change from custom to standard
- Process Control investigations
Qserv and Dax

- Qserv: prototype → maintainable code base
- Qserv multi-node deployment and execution; reliably running queries at scale
- Data access web service prototypes
SUI

• Rethought requirements
• Adding features to Firefly visualization
• Demonstrated driving from (i)Python
• Integrated with data access Web services
• Working on further pipeline integration
Science Quality

• Hired Scientist
• Starting to generate Key Performance Metrics and understand other quality metrics
• Don't yet have datasets and integration tests in place
V10.1 Release Notes

- Measurement and Table rewrite
- Astrometry refactoring, distortion in WCS
- Camera geometry from FITS
- Edge rolloff effect modeling

Note that version numbering persists from Design Phase; this is PMCS release R5.0
Big rewrites to make code modular, modifiable, and maintainable.
V11.0 Release Notes (preview)

- Aperture correction
- PSFex model incorporated into stack
- NO_DATA mask plane instead of just EDGE
- CModel (linear combination of independent Sersic fits) fluxes for galaxies
- Chebyshev model for backgrounds
- Compound fields in catalogs
- FITS file handling improvements

PMCS release R5.1 in September
V11.0 Release Notes (preview)

• Coaddition improvements:
  – Multi-band processing for coadds: detect on each band, merge footprints, deblend and measure on each band, merge catalogs choosing reference band
  – Deblended "heavy" footprints for coadd forced photometry
  – Polygon masking for CoaddPSF, bad pixel tracking and masking
Difference Imaging
Resurrection

Francisco Forster
High Cadence Transient Survey
DECam images
Subtracted with LSST Stack
Worries and Responses

• Algorithmic progress hard to predict
• Architectures evolving rapidly:
  – Cloud, AstroPy, Hadoop ecosystem

• Focus on "end-to-end" system
• Document design path and replan
• Track new architectures, maintain portability
Hardware
Hardware

• Contract (finally) to buy computing at NCSA
• Networking contracts (nearly) in place
• Getting ready for network installation
  – Mountain to Base to Santiago
Tools and Processes
GitHub
Jenkins
**RFC Process** (emphasis added, truncated)

1. **Empowerment**
   
   You are empowered by the DM Project Manager (PM) and Project Scientist (PS) to make decisions on any DM-internal matter, including technical/algorithm issues, process improvements, tool choices, etc., when:

   A) you are willing and able to do the work to implement the decision yourself or with people who agree with you,

   B) you (collectively) are willing and able to fix any problems if it goes wrong, and

   C) you believe that all affected parties (including your immediate manager) would not seriously object to your decision and implementation.

2. **RFC Process**
   
   If the above three criteria are not met, perhaps because you don't know all the affected parties or because you don't know their positions, you should publish your proposed decision and implementation as a JIRA issue in the Request For Comments (RFC) project with a component of "DM". This will trigger postings to the dm-devel mailing list and the 'Bot: RFC' HipChat room.
Team Culture (emphasis added, truncated)

The **LSST DM community includes LSST-paid staff at multiple institutions and external contributors** from around the world. We have built a strong team that collaborates well. As we hire new people, we want to make sure we are maintaining a healthy, supportive, productive culture. While *culture is best transmitted by daily example*, having some formal standards for conduct can aid newcomers and reinforce good patterns.

Since LSST is an AURA center, the **AURA standards of workplace conduct** provide a starting point. As stated there, we in DM dedicate ourselves to fostering a *civil and inclusive community characterized by mutual respect for the contributions of all individuals*. As a community, we embrace the values in that document, in addition to any **local institutional standards**.
Other Tools and Processes

- JIRA planning
- Licensing and contributions
- Investigating Slack and Discourse
Worries and Responses

- Not yet developing at maximum productivity
- Focus on developer multipliers
Relationships
Within the Project

• Camera
  – CameraGeom
  – Visualization/Firefly
  – CD3 support

• OCS
  – Middleware interface test
  – Component, telemetry, configuration definition

• IN2P3
  – JCC

• Telescope & Site
  – Base Data Center design
  – Common IT needs/services/management
  – Networking design refinement and acquisition
  – Wavefront sensor processing
Outside the Project

- Subaru/HSC
- DESC
- VO Community
- AstroPy Community
- Euclid/WFIRST
Future Plans
2016

- Finish end-to-end, start refining components
- Developer multipliers
- Run easily on precursor data
Replanning

- Refine/revise FDR documents to reflect current knowledge
- Re-cost development roadmap reflecting personnel ramp-up
Roadmap

- **2017:** Reprocess DES, continuous simulated Alert Production
- **2018:** Ready for ComCam
- **2019:** Ready for full Camera
- **2020:** Ready for Science Verification