Large Synoptic Survey Telescope

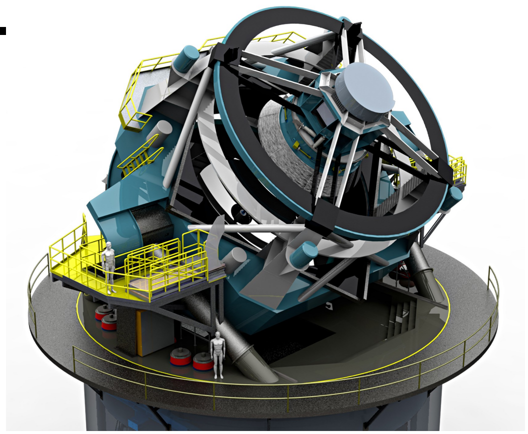
LSST Pipelines and Infrastructure for Probing Dark Energy

Raymond Plante
National Center for Supercomputing Applications

Computing and Astroparticle Physics
Aspera Workshop
IN2P3 – Lyon
8 October 2010

- LSST as a scientific instrument
- Some requirements driving architecture
- Hardware Infrastructure
- Processing Middleware
- Issues & Questions

The LSST scientific instrument

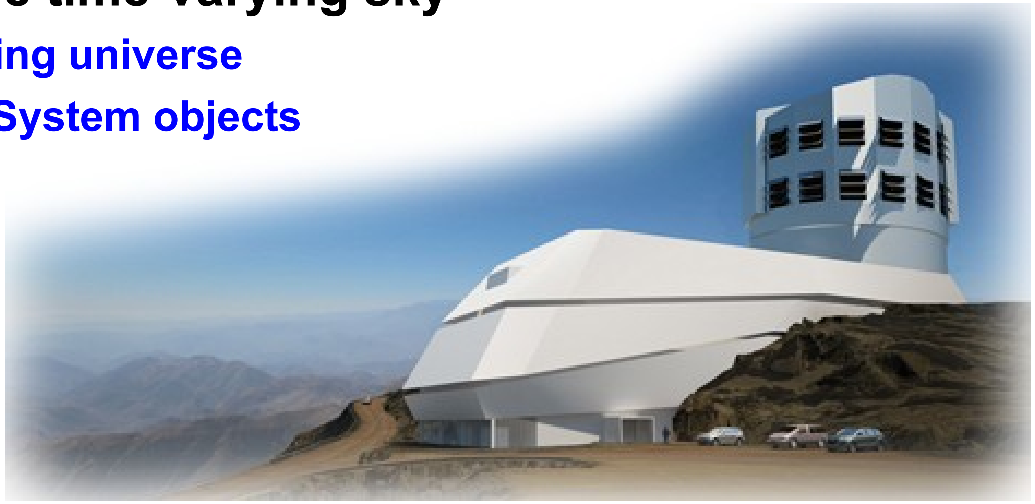


- A new telescope to be located on Cerra Pachon in Chile
  - 8.4m dia. Mirror, 10 sq. degrees FOV
  - 3 GPixel Camera, 6 filters
  - Image available sky every 3 days
  - 10-year survey begins in 2017
  - Sensitivity – per “visit”: 24.5 mag; survey: 27.5 mag
  - First computing hardware systems to be purchased in 2015

  “Fast, deep, wide”

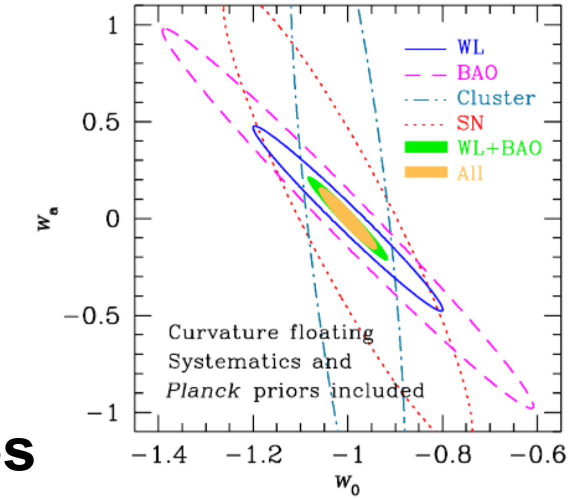
- Science Mission: observe the time-varying sky
  - Dark Energy and the accelerating universe
  - Comprehensive census Solar System objects
  - Study optical transients
  - Galactic Map

- Named top priority among large ground-based initiatives by NSF Astronomy Decadel Survey



A Probe for Dark Energy

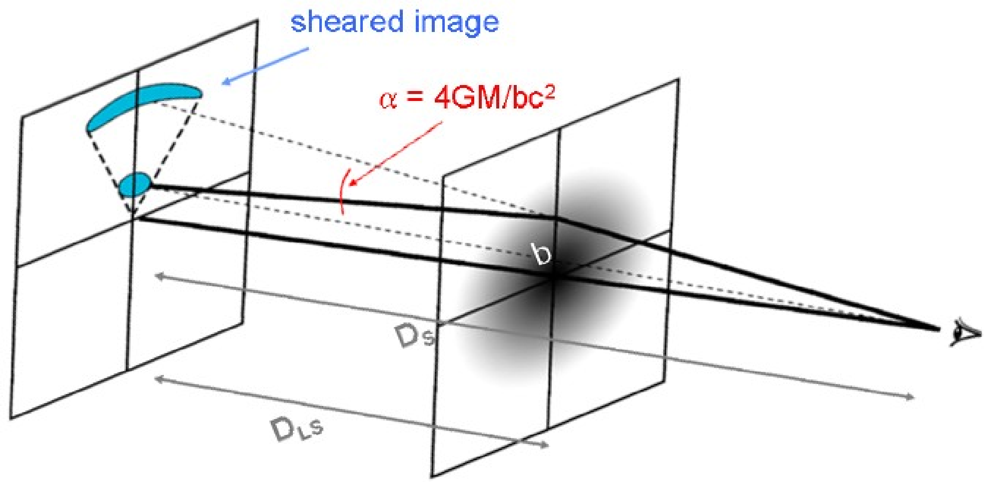
- **Dark Energy, as characterized by *w(z)*, can be measured via several observational techniques**
  - **Each place an independent constraint that can be combined to break degeneracies**

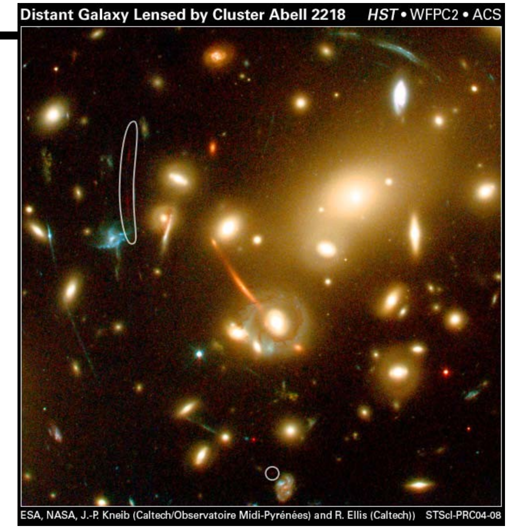


- **LSST can support four techniques**
  - **Weak lensing: cosmic shear as a function of z**
  - **Baryon Acoustic Oscillations via galaxy distribution power spectrum**
  - **Evolution of mass function of clusters**
  - **Redshifts and distances of Type Ia supernovae**

Ref: *LSST Science Book: http://www.lsst.org/lsst/scibook*

Computing Challenge: weak lensing

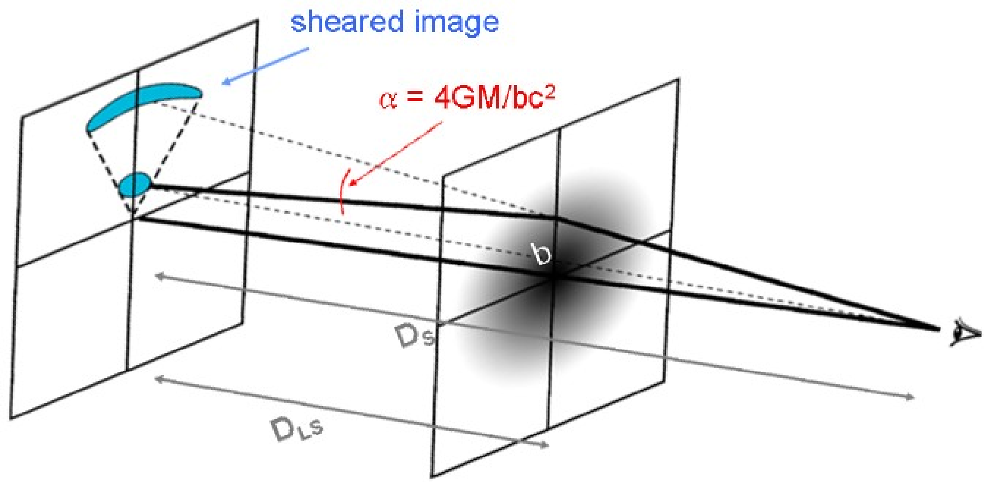


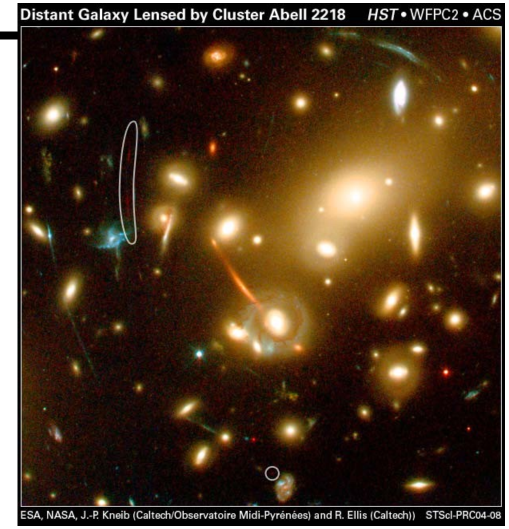


- **The weak lensing effect**
  - **Shape of a distant galaxy is distorted as its light passes through massive foreground clusters**
    - In extreme, stretches into arcs
  - **Using equations of general relativity, it is possible to derive the cluster mass distribution from the distortions on the background galaxies**

Computing Challenge: weak lensing

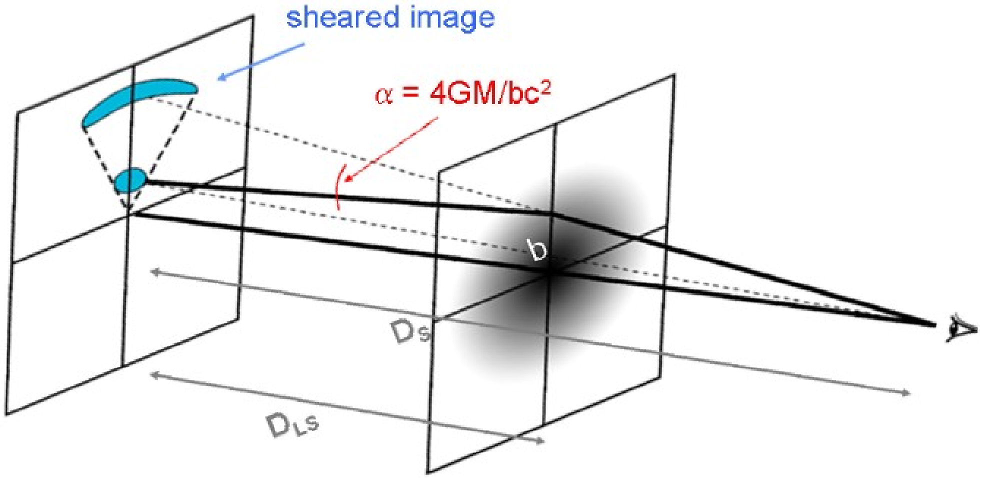
Large Synoptic Survey Telescope

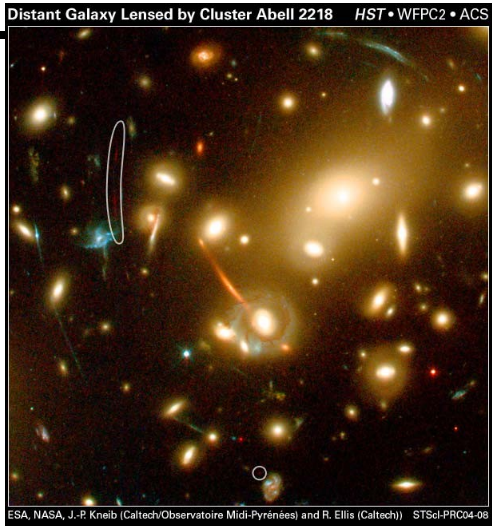




- **Challenge 1: handle systematic effects**
  - **Galaxy shape is distorted shape convolved with telescope PSF**
  - **PSF changes with observation & with position (in FOV)**
  - **Requires new algorithms in PSF fitting**
  - **Multifit: fit shapes in individual observations, not in coadd**
    - Use coadd to locate galaxies; use individual frames to fit
    - Data orchestration challenge

Computing Challenge: weak lensing





- Challenge 2: deriving mass distribution
  - Iterative, multi-parameter model fitting
  - Computationally intensive

Some requirements driving architecture

Large Synoptic Survey Telescope

- **Short exposures**
  - **Each field observed with 2 30-second exposures**
  - **Data Rate: ~13 TB/night**
- **Real-time Processing**
  - **Process new fields within 1 minute of exposure**
  - **Detect new sources, variable sources, and moving objects**
    - Via image subtraction
    - Filter out known sources, update orbits of known solar system objects
  - **Issue a *VOEvent* to interested users announcing new objects**
    - Follow-up by spectroscopic telescope can measure redshift for Ia SNs
- **Data-Release Processing**
  - **Nightly processing repeated, completed a night within 24 hours**
  - **Additional processing to create higher level products**
    - Co-added sky, Object Catalog, “Source” Catalog, Variable Object Catalog...

    Released on a yearly basis
  - **For each year’s release, all previous years’ data will be reprocessed**
    - To take advantage of the latest algorithms, ensure uniform data products

    Computing power needs to grow with time

    Archive Center: 60 TF (Yr 1) – 270 TF (Yr 10)

Some requirements driving architecture

- **Science Product Production**
  - **Produced by Community, not by observatory**
  - **Most science produced directly from analysis of catalogs**
    - Requires community search capabilities
    - Categorized in to four performance classes
      - With associated populations: many short queries, few power queries
    - Implies overall database performance requirements
    - Access to all releases of catalogs
  - **Some science will require new pipeline processing**
    - Image-based processing
    - Reprocessing using different parameters, user-provided algorithms
    - “Power problems” will require substantial resources
  - **Where do computing resources come from**
    - LSST will provide (small) fraction of needed storage and processing for science product production by community
      - Computing: 18 TF (Y1) – 50 TF (Y10)
      - Storage: 3 PB (Y1) – 12 PB (Y10)
    - Place Data near community (grid) computing platforms and networks
      - Place Archive Center at NCSA

Cost Constraints

- **Data Management Hardware budget for first light system: ~$20M USD**
- **Data Volumes:**
  - **Images: 13 TB/night raw, uncomp. → 47 TB/night calibrated, comp.**
  - **Database:**
    - Year 1: 19 Billion stars & galaxies from 290 Billion detections: **170 TB**
    - Year 10: 50 Billion stars & galaxies from 2.8 Trillion detections: **~ 9 PB**
  - **Users will require access to prior releases**
- **Constraining the cost of storage**
  - **Permanent Storage restricted to:**
    - Raw images
    - Released Co-added Sky
    - Released Catalogs

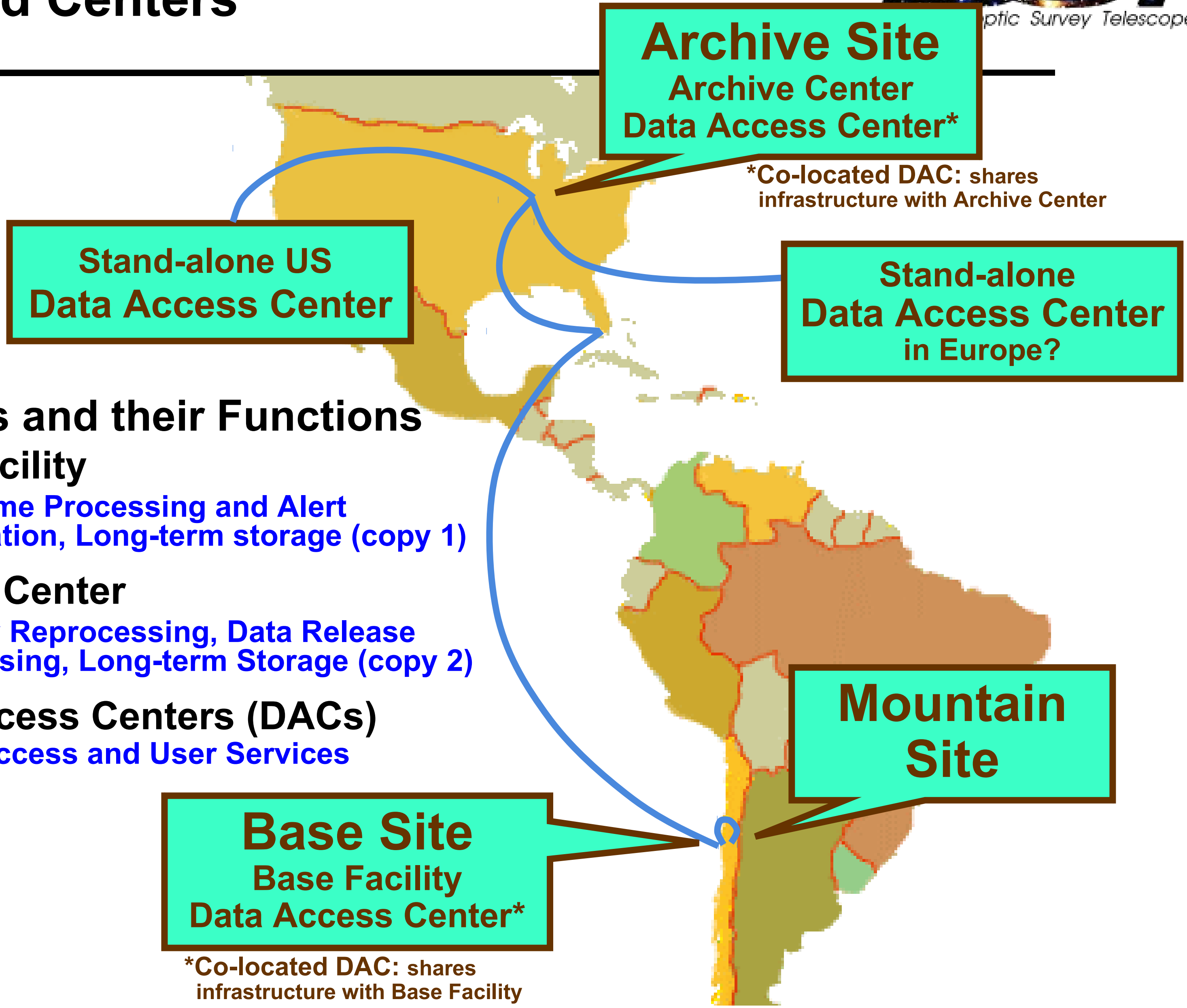
    Calibrated images stored temporarily, regenerated on demand
  - **Technology Choices:**
    - Long-term storage: disk versus tape; **current baseline: tape**
    - Fast storage: disk versus solid state; **current baseline: disk**
      - Hybrid systems for high performance (Greywulf research by Szalay, JHU)

Baseline Design Strategy

Large Synoptic Survey Telescope

- Based on known technologies on vendor roadmaps
- Baseline revised each year
- Built from commodity technologies at time of purchase
- Plan for ongoing growth, replacement through l0-year life of the observatory
  - Expect a complete replacement halfway through
  - Periods of mixed technologies
- Develop model for costs evolving over time
  - Based on vendor roadmaps, past experience
  - Construction carries ~40% contingency

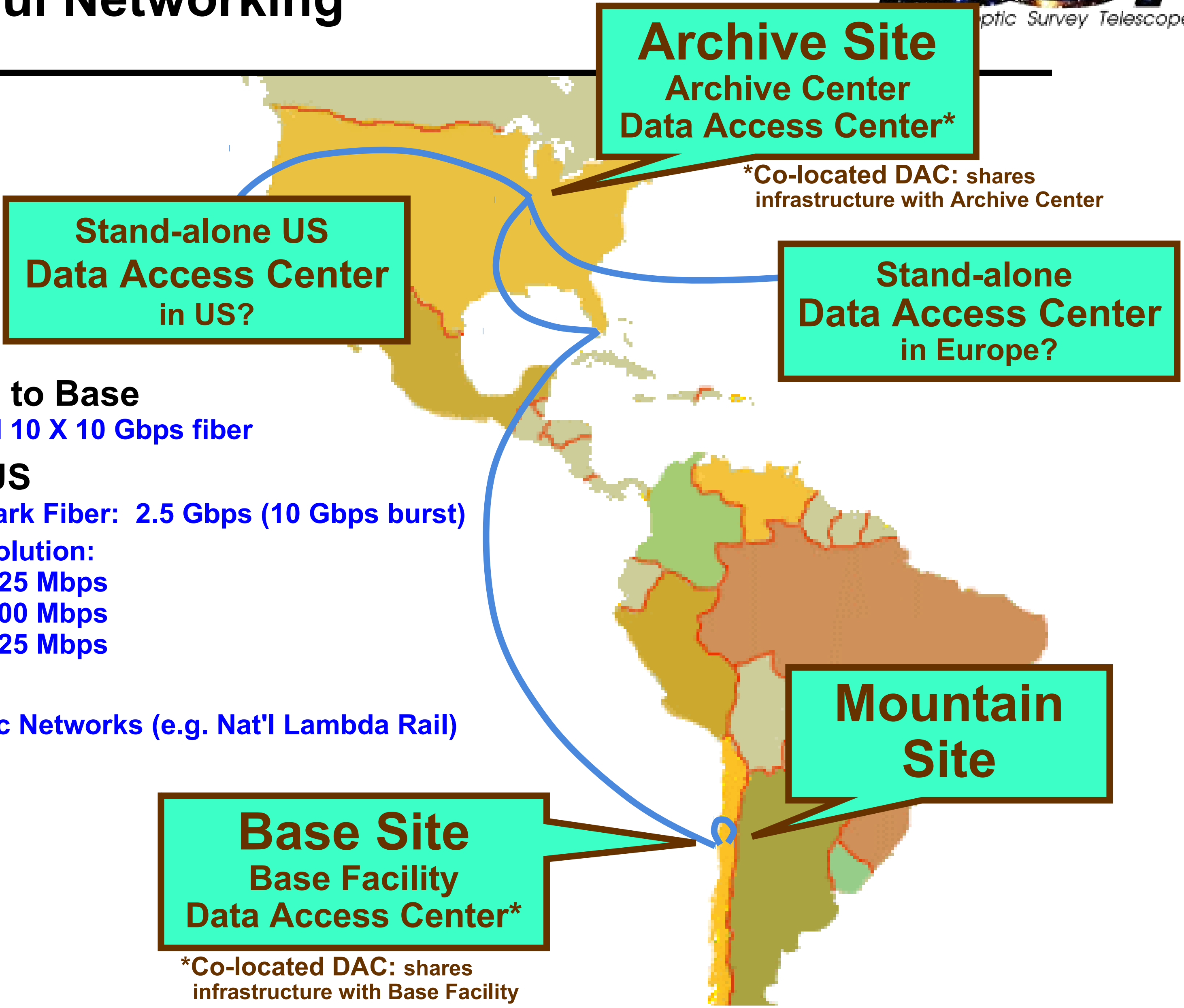
Sites and Centers



Site Roles and their Functions

- **Base Facility**
  Real-time Processing and Alert Generation, Long-term storage (copy 1)
- **Archive Center**
  Nightly Reprocessing, Data Release Processing, Long-term Storage (copy 2)
- **Data Access Centers (DACs)**
  Data Access and User Services

Long-haul Networking



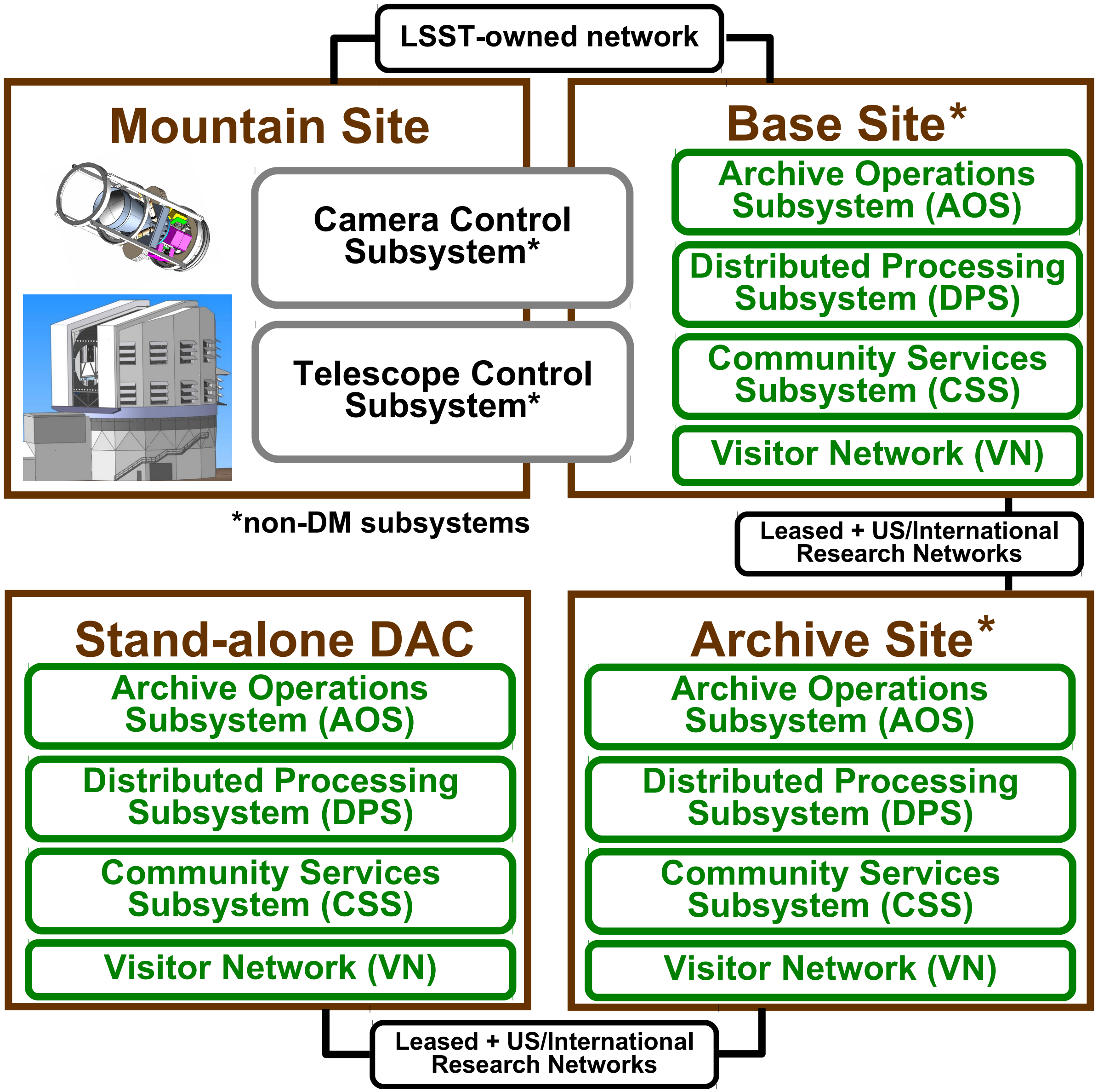
- **Mountain to Base**
  - Dedicated 10 X 10 Gbps fiber
- **Base to US**
  - Leased Dark Fiber: 2.5 Gbps (10 Gbps burst)
  - NOAO Evolution:
    - 2010: 325 Mbps
    - 2011: 500 Mbps
    - 2012: 625 Mbps
- **Intra-US**
  - Scientific Networks (e.g. Nat'l Lambda Rail)

Sites and Subsystems

- **Site:** a physical space in a building that hosts LSST DM subsystems
- **Subsystem:** a defined combination of…
  - configured hardware,
  - software stack
  - running processes
  - responsible personnel

* **Co-located DAC Sites**
  - Includes Community Services Subsystem (CSS)
  - Shares infrastructure with DM facility

- Sites run with a common set of Subsystems
  - Configured differently for the role of the site



Hardware Summary

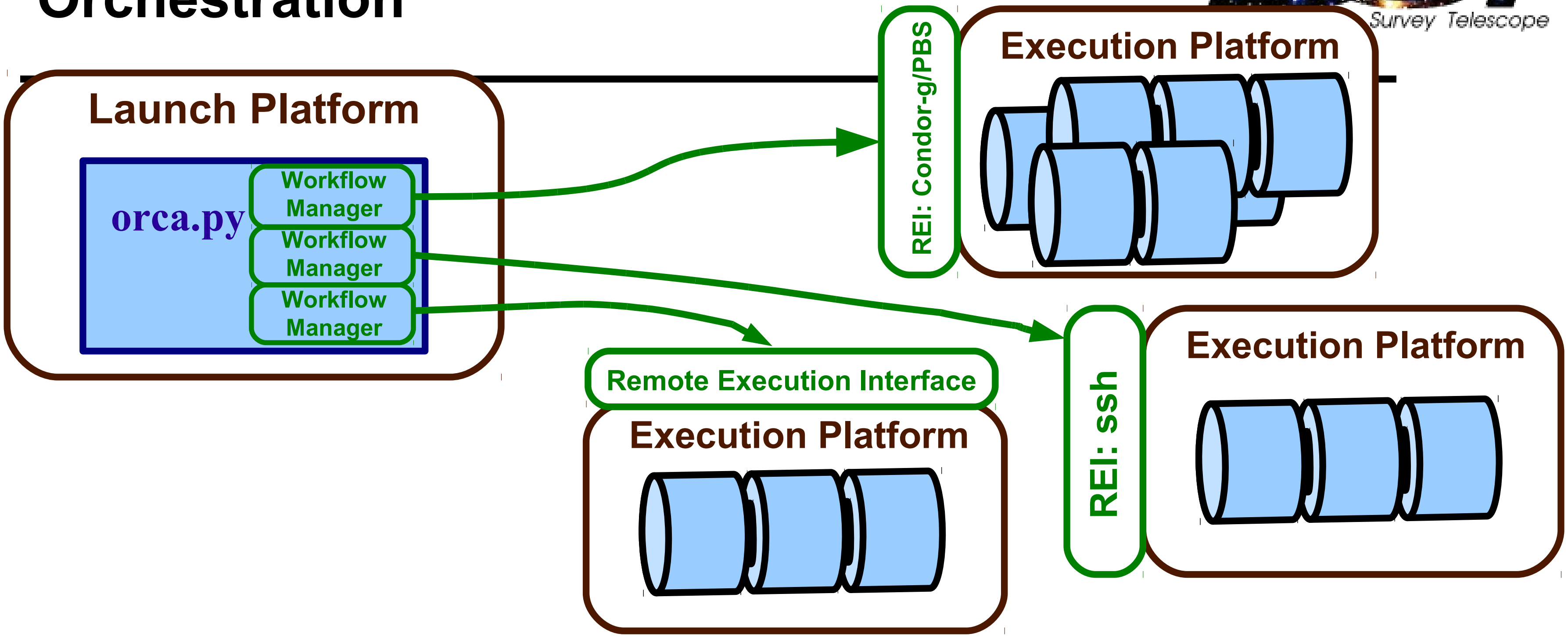
- **Storage**
  - **Three-tier Model**
    - Slow, deep storage
      - Used for permanent copies of data products
      - Tape library + “slow” disk cache
      - 24 (Y1) – 90 (Y10) PB
    - Medium, “near-line” storage
      - Primarily to provide temporary storage for virtual data products
      - Archive Site: 8 – 15 PB
    - Fast storage – 2 types
      - Scaled for capacity, support HP access to files by processing pipelines
      - Scaled for data bandwidth, support HP access to database
        - » Scales with number of disk spindles
        - » Implies 3x capacity at commodity disk sizes (3 – 30 PB)
- **Computing**
  - **Commodity cluster nodes (baseline: 16-core, 32 GB memory min.)**
  - **Capacity:**
    - Real-time processing at Base Site: ~10k cores (600 nodes), 60 TF
    - Archive Center: 90 – 270 TF
    - Data Access Centers: 25 – 58 TF
  - **Variants under consideration: GPU-enhanced, SSD-enhanced**

Middleware View of LSST processing

Large Synoptic Survey Telescope

- Two categories of processing => two strategies
  - Alert Production: Real-time Processing
    - Executed nightly
    - Minimize I/O by keeping data in memory
      - Stress data-parallelism; requires consistent routing of data
      - Isolate and minimize parallel process cross-talk
    - Parallelism implemented using MPI
  - Data Release Production: High-volume Processing
    - Executed yearly but continuously
    - We can trade performance for robustness
    - Processing is more complex
      - with changing axes of parallelism
    - Parallelism implemented using Condor
- Categories include some common needs
  - Provenance tracking
  - Logging under high levels of parallelism
  - Encapsulated data access via logical identifiers

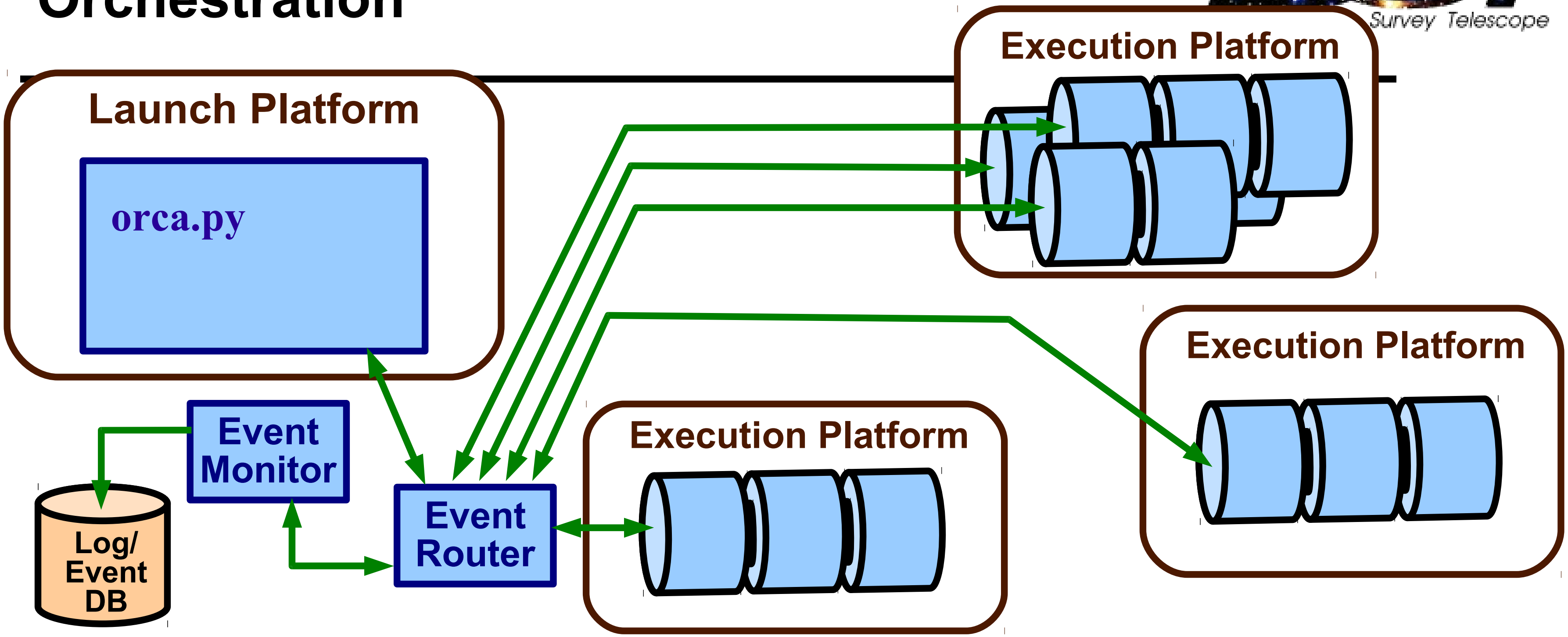
Orchestration



Orchestration layer launches pipelines on remote execution platforms

- **Adapts to different types of platforms and the way they run applications**
- **LSST designed to run on own dedicated platforms or community platforms (e.g. NCSA public resources)**
  - Basis for preferring grid solutions
- **Launch mechanisms currently supported as plug-ins**
  - Ssh
  - Condor-g: generic interface to local batch system (e.g. PBS)
  - Condor/Glide-in
- **Agnostic about form of pipeline: (Wrapped) black box app or app using LSST Pipeline Framework**

Orchestration

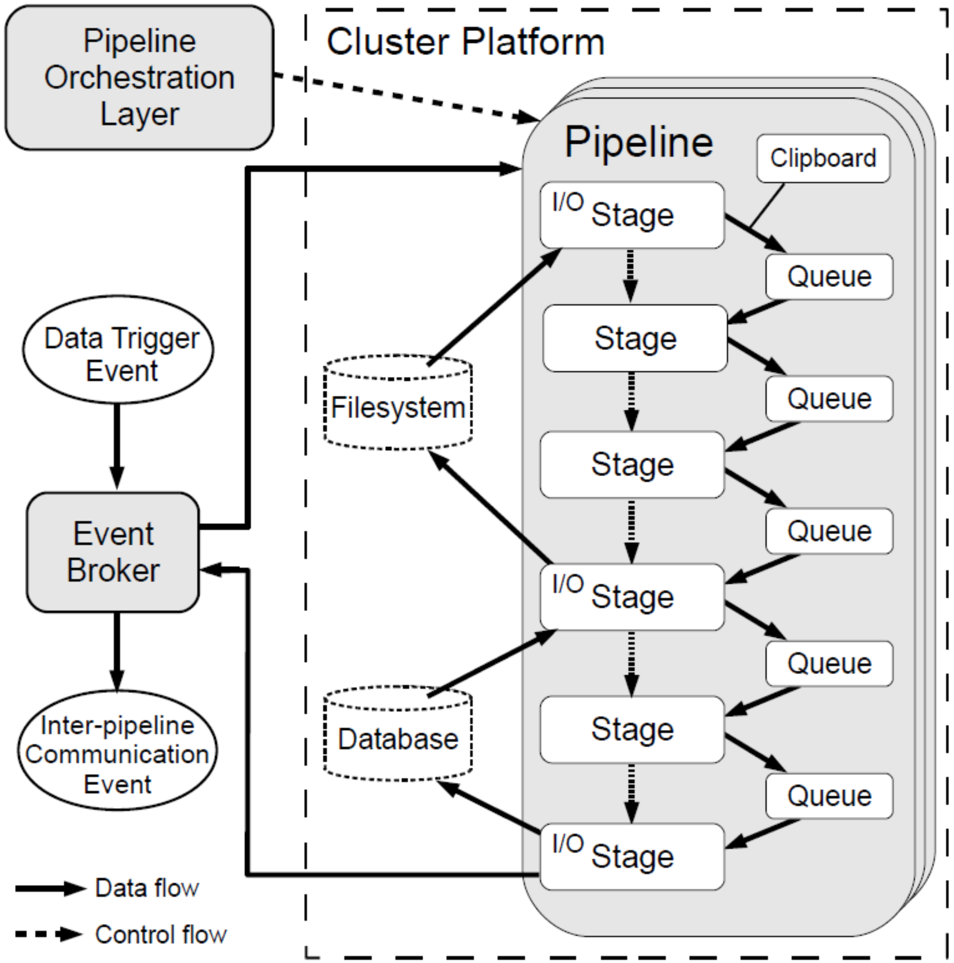


Pipelines communicate with each other and with Orchestration Layer via Events

- **Event system based on the Java Messaging Framework (JMF)**
- **Pipeline log messages sent out as events for remote recording**
- **An Event Monitor analyses progress to detect possible problems**
  - Node failure, Runaway processes, ...
  - Can signal orchestration layer to relaunch failed processing
- **Inter-pipline communication via Events**
  - One pipeline may “wait” until an expected event with needed information arrives
  - Event payloads are light: one pipeline may tell another where to look for data

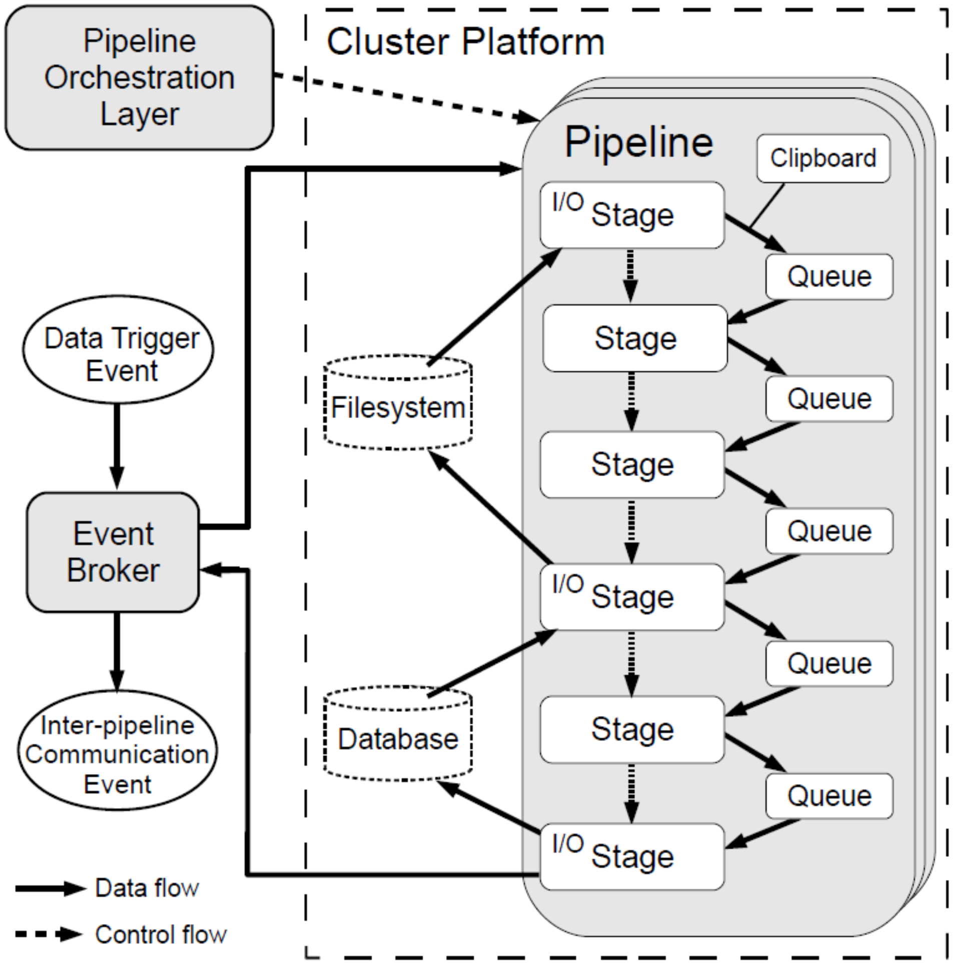
The LSST Pipeline Framework

- **Pipeline = a sequence of Stages**
  - **"Harness" = container for stitching together stages**
- **Pipeline has N+1 threads**
  - **Pipeline thread is the master controller**
  - **Slice threads process a data-parallel unit of data**
    - e.g. CCD segment
  - **All threads have same basic structure**
- **Sequence of Stages**
  - **Data queues sit b/w each stage**
  - **Data is passed from one stage to another through queues via a "Clipboard"**
    - Clipboard = hierarchical dictionary
    - Stages look for input data on clipboard and place new data items on it.



The Pipeline Harness

- **I/O is done via dedicated I/O Stages**
  - Algorithm stages are isolated from I/O
  - Output Stages can write products to disk at any time
- **Processing is Event driven**
  - A data trigger event signals the first stage which exposure to process next
  - Harness puts event data on clipboard
  - Each stage is run in sequence
  - After last stage is executed, pipeline returns to first stage, waiting for next event.

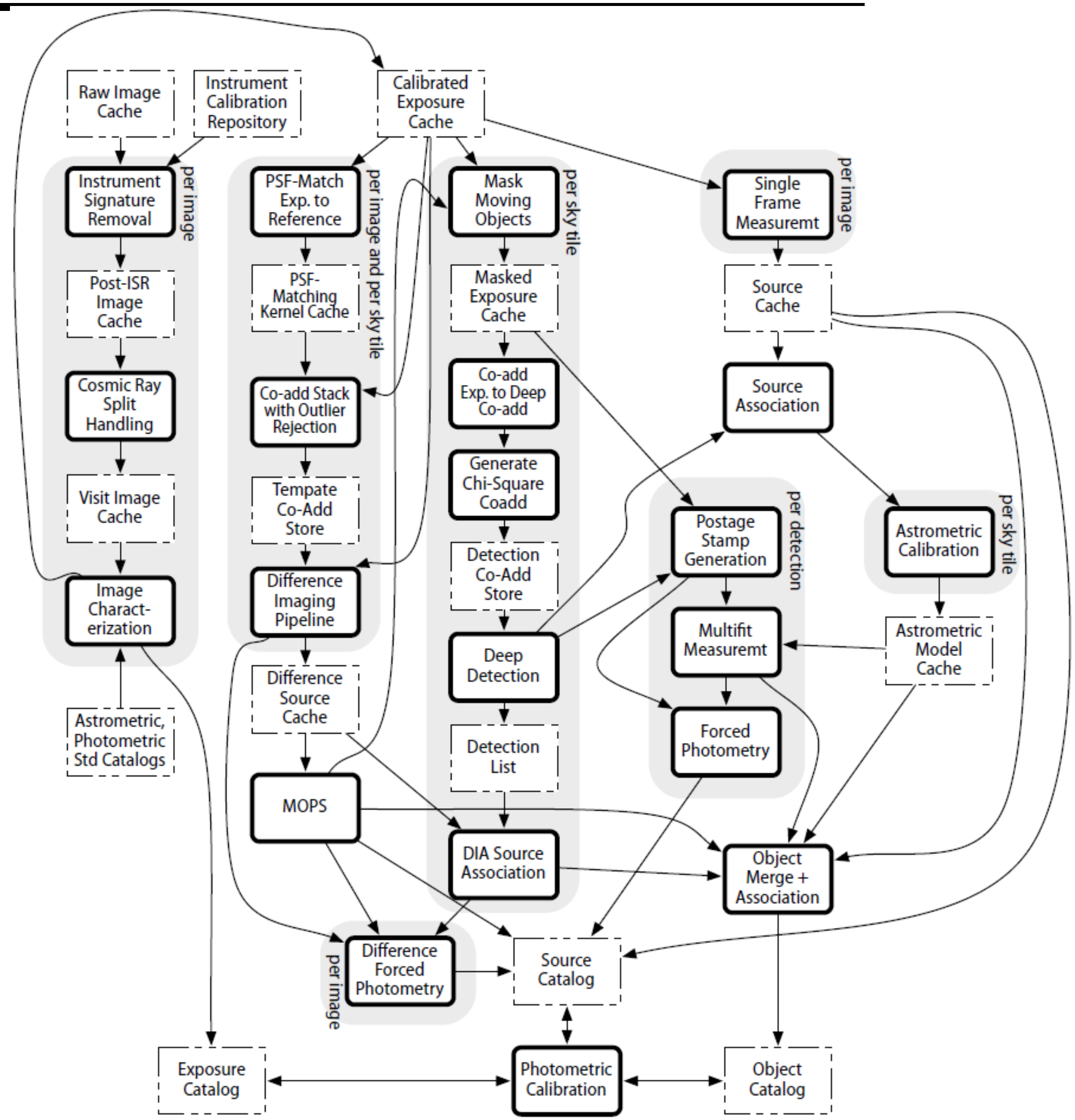


Alert Production (Real-time)

- **Three MPI pipelines**
  - **Image Processing and Source Detection (IPSD)**
    - Calibrate images, create difference images, detect variable sources
  - **Moving Objects Prediction (MOPS)**
    - To filter out known solar system objects
  - **Association Pipeline**
    - Filter out known objects, left with new sources
    - Send sources to alert distribution system
- **Long-running processes**
  - **Events tell IPSD, MOPS when new data is ready**
  - **Data is already staged (out-of-band)**
  - **IPSD, MOPS signal Association pipeline via Event**

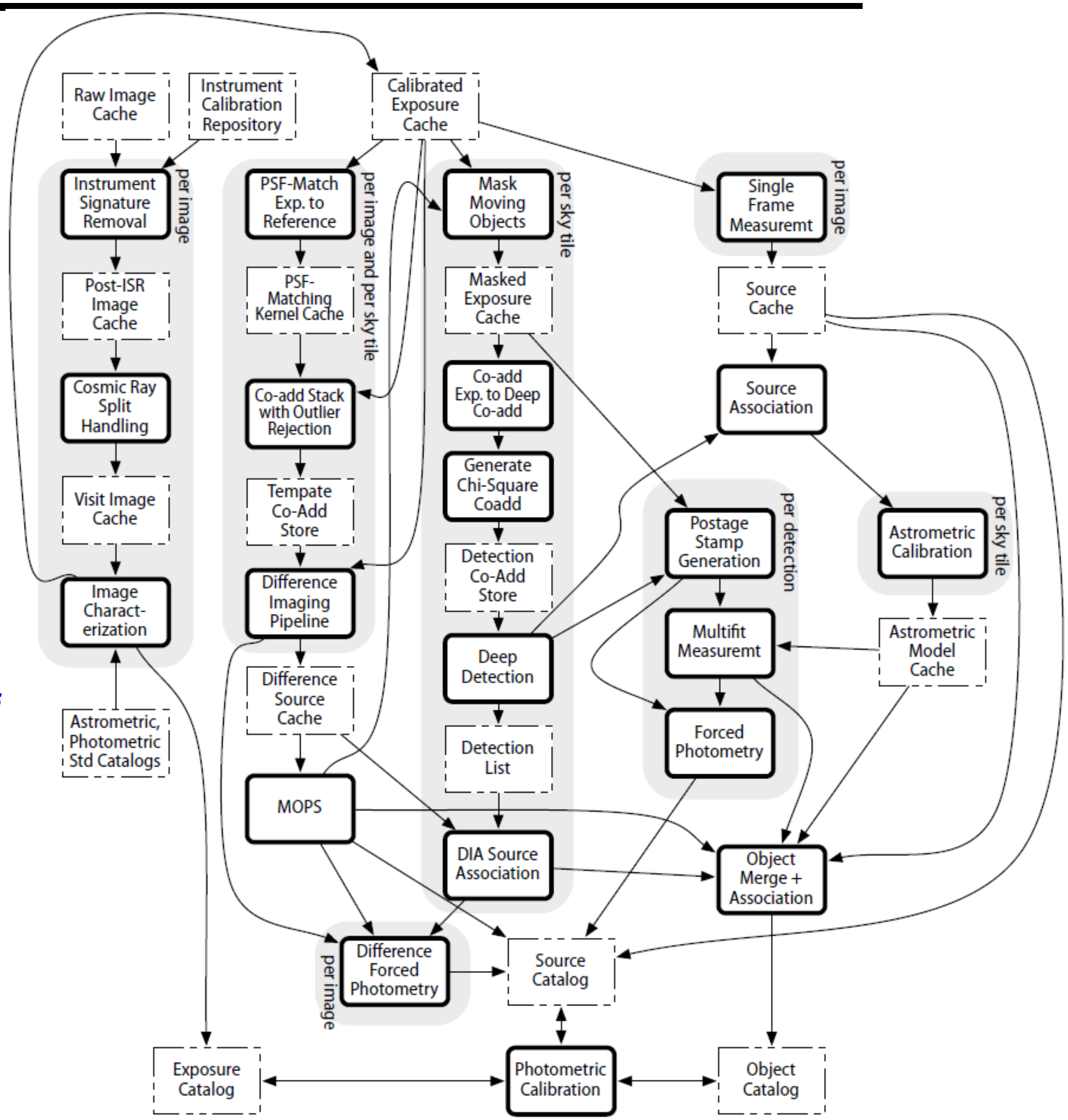
Data Release Pipelines (High-Volume)

- **Sequencing is more complicated**
  - **Many interdependencies**
  - **Shifting “axes of parallelism”**
    - By CCD segment
    - By sky-tile
    - By object
- **Real-time is not required**
  - **Trade performance for robustness**
  - **Do more caching of data to disk**
  - **Leverage existing technology: Condor/DAGMan**



Data Release Pipelines

- **Short and Narrow pipelines**
  - Single slice, operating on single data-parallel unit of data
  - Only one unit will pass through pipeline per instantiation
- **Homegrown data-driven, opportunistic schedule**
- **External process for staging input data**
  - Volume to process >> capacity of disk
  - Will listen for events indicating progress of pipelines
  - Will stage data from mass storage before needed



JobOffice: Opportunistic, Data-Driven scheduler

- DAGMan commonly used to manage interdependencies between Condor Jobs
  - To create DAG, must collect all knowledge about inputs/outputs, trace flow, *a priori*
  - Information management does not scale easily
- Homegrown "Job Office" inspired by opportunistic blackboard-based pipeline systems used in astronomy
  - Each pipeline type has a manager (i.e. JobOffice) configured to know what type of data it operates on
  - JobOffice converts events about available data into candidate jobs
  - When a Job is ready (all prerequisites available), sends job to available pipeline
  - When pipeline completes, it announces availability of its output products.

Issues and On-going Questions

- **How realistic are our technology predictions?**
  - **Can contingency cover its inaccuracies?**
- **Can we effectively use the calculated TF of the proposed systems?**
  - **Moore's Law is being met via multiple cores per chip**
  - **Effective use of multi-core systems is a software challenge**
    - Bottleneck: memory bandwidth
  - **GPU-enhanced clusters**
    - Which of our pipelines can take advantage? How will it affect our software development?
- **Storage Technology Choices**
  - **How will cost/performance curve for SSD vs. HDD evolve?**
    - Will SSD play a role in high performance computing component?
  - **How will cost/performance curve for disk vs. tape evolve?**
    - Will disk play a bigger in long-term storage?
    - Cost model currently described in terms of a hybrid system
    - Will technologies like MAID become important?

Issues and On-going Questions

- **How will down-selecting on grid technologies affect overall architecture**
  - **Be less generic in order to take advantage of unique product features**
  - **Fault tolerance is important**
- **How do we implement fault tolerance at all levels**
  - **We have a plan, strategies; not all have been prototyped, yet**
- **Role of Virtualization**
  - **Community processing: means for providing user-supplied algorithms in a secure environment**
  - **Static allocation versus dynamic allocation of resources**
  - **Snapshots as an alternative to checkpointing**

Roadmap to First Light

- **Two more years of R&D phase**
  - Further prototyping
  - Refine to final design
  - Significant software base in place
- **Construction begins 2012**
- **First Light: 2016**
- **Operations: 2017**
- **Data Releases:**
  - Two data release in first year
  - Yearly after that