Processing System Hardware

Heidi Brandenburg
IPAC/Caltech
WSDC Functional Block Diagram

- **Ingest**
  - Level-0,1b frames
  - Coadds
  - Frame index
  - Calibration data
  - Ancillary data
  - 2MASS ref catalog
  - QA Data
  - Meta-data

- **Ops Archive**
  - Coadd
  - Frame (epoch)
  - QA Meta-data
  - Image Meta-data
  - Final Products
  - Catalogs
  - Atlas

- **FTP Site**
- **Scan/Frame Pipelines**
- **Multi-Frame Pipeline**
- **MOS**
  - H/K
  - Ancillary data
  - QA Data
  - Meta-data

- **HRP**
  - Images

- **SOC**
  - Maneuvers

- **UCLA**
  - Mission Status
  - Expectant Public

- **Science Team**

- **Web**
  - Interdisciplinary Research Archive Server (IRSA) Archive
    - Working Source DB
      - Coadd
      - Frame (epoch)
      - QA Meta-data
      - Image Meta-data
      - Final Products
      - Catalogs
      - Atlas

- **WSDC**
  - Caltech

WISE Science Data Center CDR – January 29-30, 2008
Driving Requirements

The WSDC operational hardware must

- Deliver processed data in time to meet the operations scenario
- Be scalable, so we can start with a little now and add more with need
- Support heterogeneous operating systems, programming languages, file systems, and database engines
Processing System Hardware

- 25 Cluster Compute Nodes
  - 2xQuad Core Xeon
  - ≥ 8 GB RAM
  - ≥ 500 GB Local Disk

Ingest

Development

Spare
  - 2xDual Core Xeon
  - ≥ 16 GB RAM
  - RAID 5
  - Redundant Power

Network Switch
  - gigabit ethernet
  - gigabit ethernet (potentially trunked)

Operations Archive
  - Sun File Server
  - Nexsan Storage

WISE Science Data Center CDR – January 29-30, 2008
Current Status

WISE Science Data Center CDR – January 29-30, 2008
Cluster

- ~25 2xQuad-core Commodity Dell servers. 1U, 8 GB RAM, 500GB SATA storage
- Some machines will have better resources: more RAM/faster CPUs
- We run RHEL4.
- Current frame pipeline scales with increasing cores and increasing number of cluster nodes*

* Scales as long as we can push data in and out fast enough, of course.
Operations Archive

- The operations archive hosts the WSDC software tree and functions as persistent data store for L0, L1, and L3 products
- Accessed by pipelines to retrieve their inputs and push their (minimal) products
- Critically important that the operations archive can support the data rate required to meet the mission operations scenario. Currently we calculate this rate as 2Gbs.
Backup

- Current backup provided by IPAC Systems Group
- Future backup of operations archive occurs on WISE hardware
- Utilize hot swappable disk for local, cycling backup
- Utilize LTO for long term offsite backup (Telemetry, expanded L0 products, Archive)
Disaster Recovery

- In case of disaster WSDC provides minimal functionality: ingest of telemetry and housekeeping, generation of L0 images, and quicklook pipeline processing to assess spacecraft health.
- These functions can be provided by single offsite machine, configured as a normal cluster node, with attached storage and a copy of the WSDC software tree.
Job Scheduling

- Use Condor, a cluster scheduling package from the University of Wisconsin.
  - WISE has a simple use case: first pipeline in is the first pipeline executed
  - We don’t use MPI, checkpointing, backfilling or other fancy cluster technologies available with Condor
  - Condor can match jobs with big resource needs to the machines with those resources (example: Condor matches a big coadd to a machine with > 8 GB RAM)
Cluster Resource Monitoring

- Use Ganglia, an open source project for accumulating, collecting, and displaying near-realtime resource use measurements for clusters
- Can be expanded with custom monitors
Frame Pipeline Execution

- Depending on the simulated scene current frame pipeline executes in 150-180 seconds CPU time.
- On an unloaded node, elapsed clock time is less than time on CPU, typically resulting in CPU utilization of 110-120%
- CPU bound
Concurrent execution via Condor

- Condor partitions a node into 8 “virtual machines” - 1 per core
- Condor dispatches 1 job to each virtual machine
- Ran experiments queuing different numbers of frames through Condor. For runs with a fill ratio > 1, execution clock times are flat, with ~98% of time spent on CPU
- With nodes fully loaded, one pipeline per core, we are close to using our full 8GB RAM in portions of the frame pipeline.
- Still not waiting on IO.
Concurrent execution via Condor
Execution of a Simulated Scan

- 250 frame simulated scan
- Run on 4 machine development cluster
- Ops Archive disk mounted on dev server, exported to cluster via NFS
Execution of a Simulated Scan

- Single node statistics
Tests were done by loading more than 8 pipelines onto cluster nodes

- At 12 pipelines/node seeing IO waits to node local storage
- At greater numbers of pipelines, started paging
- Largest 5 minute average traffic on the cluster network ~40Mbps

Unable to properly test network loading scenarios via frame pipeline with current cluster hardware
Deployment Schedule

- **Phase 1 / WSDS v2**
  - Supports mission scenario testing
  - 10 cluster nodes + 1 master
  - 1 fileserver, 1 disk array (v2 - 4 months)
  - Ingest machine

- **Phase 2 / WSDS v3**
  - Supports Ops readiness tests, launch, IOC
  - 25 cluster nodes + 1 master
  - 2 fileservers, 2 disk arrays (v3 - 2 months)
  - Backup System (v3 - 4 months)
  - System complete as specified

- **Phase 3 / Q2 2010 (or 3 months post launch)**
  - Purchase additional disk and CPU based on measured needs for final processing
Issues/Concerns

- Testing network & operations archive scalability to 20 cluster nodes
- Condor job management tools are command-line and based upon its own job abstractions & terminology. TBD need for job management tools targeted at a human operator working on scans, frames, and coadds.