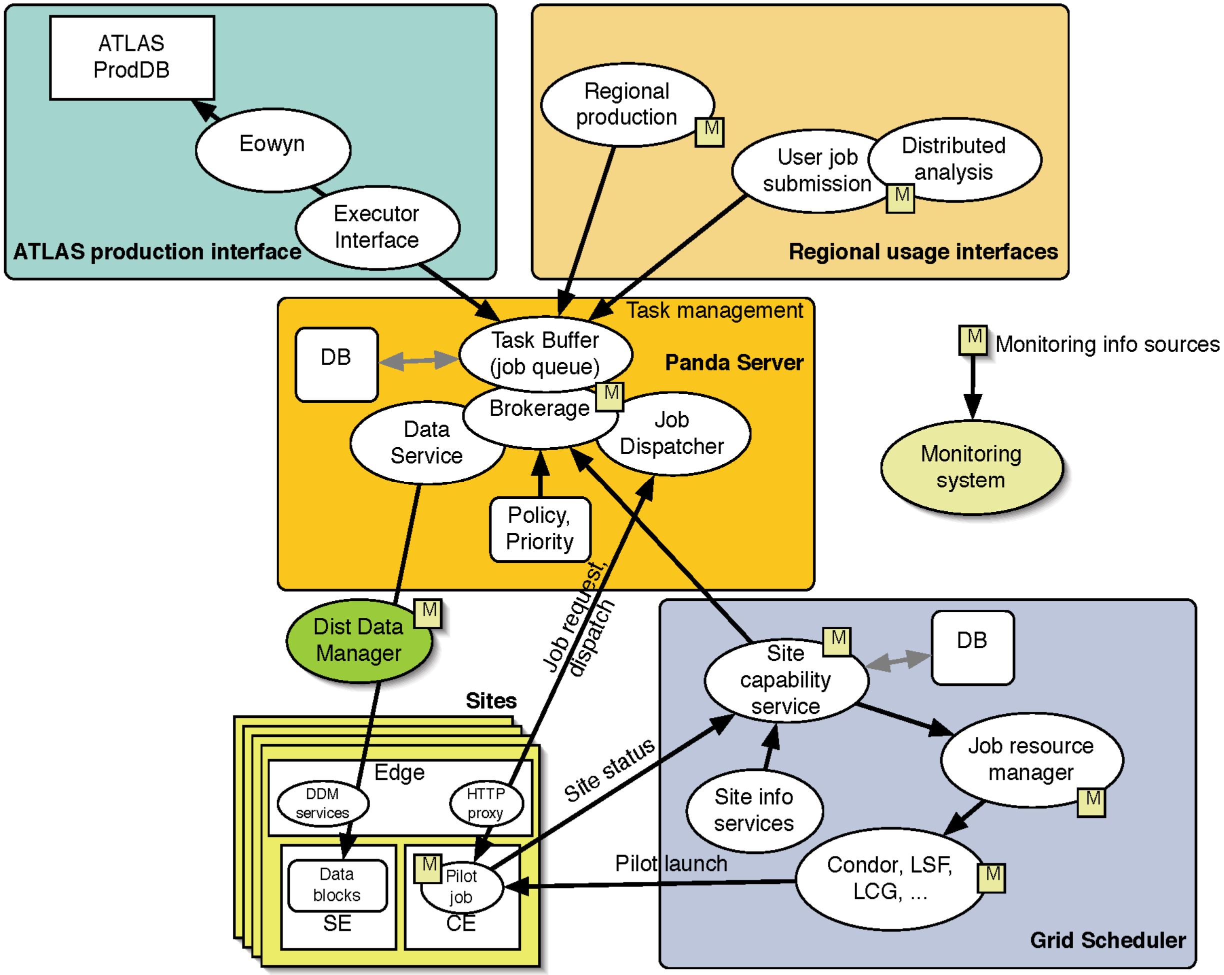
Panda:
Production and Distributed Analysis System

Tadashi Maeno (BNL)
on behalf of PANDA team

Overview

- PanDA – Production and Distributed Analysis
- Designed for analysis as well as production
- New system developed by US ATLAS team
- Project started Aug 2005, prototype Sep 2005, production Dec 2005
- Tightly integrated with ATLAS Distributed Data Management (DDM) system
  - Pre-staging of input files and automated aggregation of output files
- Highly automated, and requires low operation manpower
- Not exclusively ATLAS: has its first OSG user
  - Cf. protein molecular dynamics (CHARMM) talk tomorrow

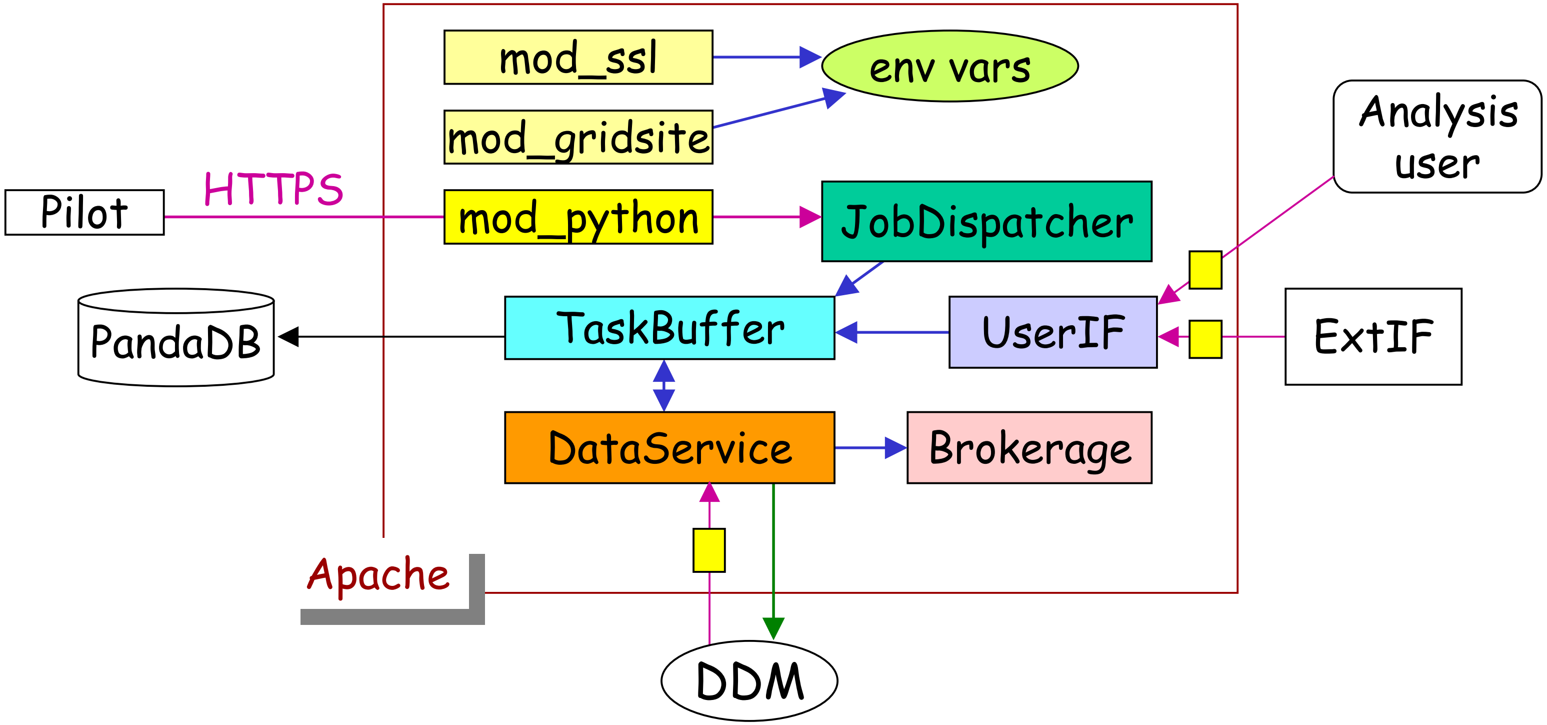
Panda System



- **Panda Server**
  task management
- **Pilot**
  run actual job
- **Scheduler**
  send pilot jobs
- **Panda Monitor**
  integrated monitor
  for production/analysis

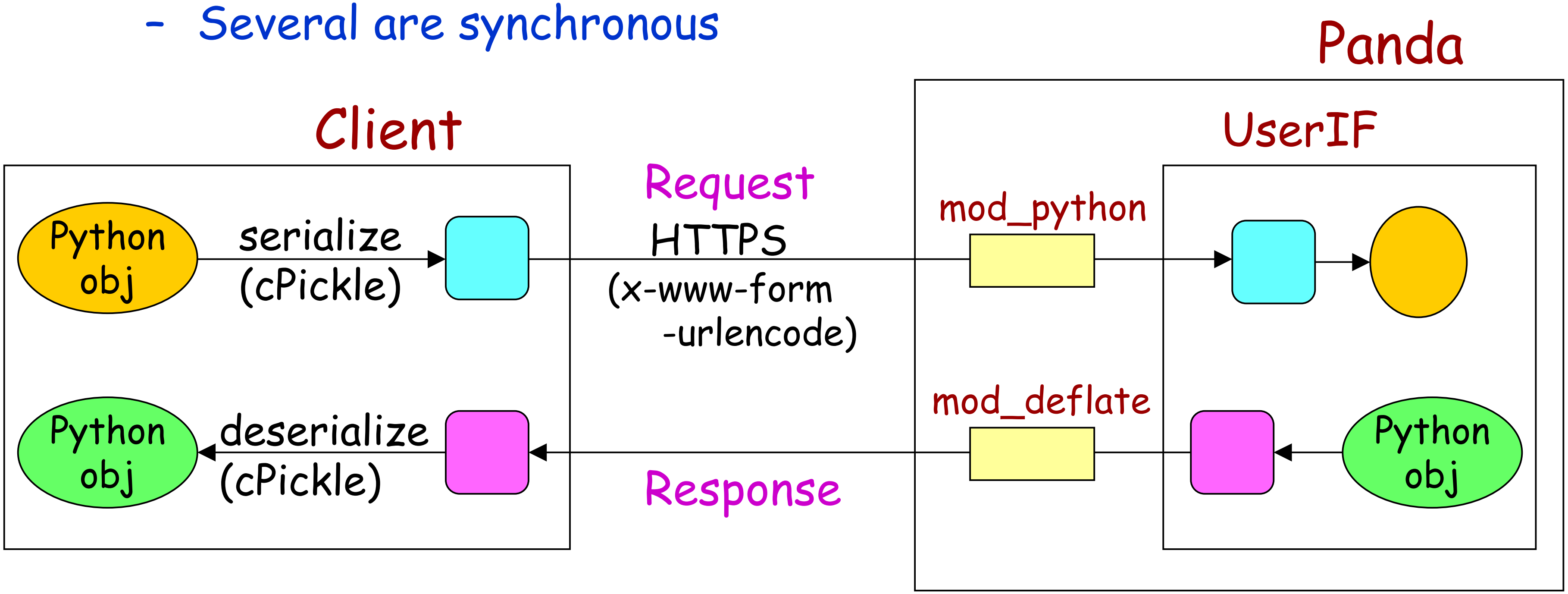
Panda Server

- LAMP stack
  - RHEL3 / SLC4
  - Apache 2.0.59
  - MySQL 5.0.27 - InnoDB
  - Python 2.4.4
- Multi-processing (Apache child-processes) and multi-threading (Python threading)



Panda Server (cntd)

- HTTP/S-based communication (curl+grid proxy+python)
- GSI authentication via mod_gridsite
- Most of communications are asynchronous
  - Panda server runs python threads as soon as it receives HTTP requests, and then sends back responses immediately. Threads do heavy procedures (e.g., DB access) in background → better throughput
  - Several are synchronous



Pilots

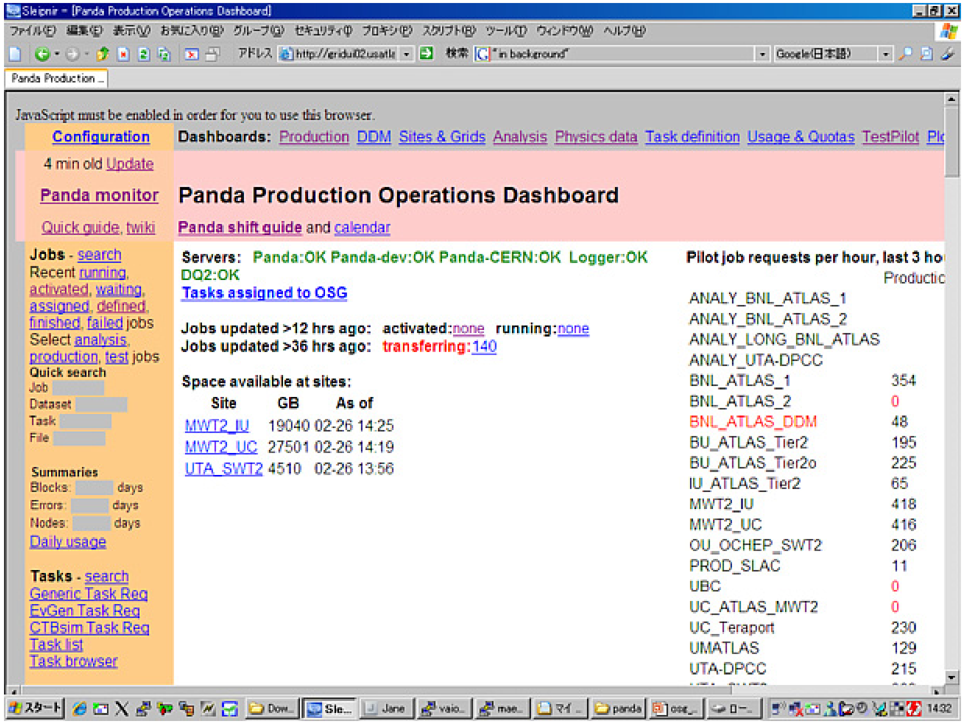
- Are prescheduled to batch system and grid sites
- Pilot runs actual job when CPU becomes available → low latency
- Access to storage element
- Multi-tasking
  - Job-execution
  - Zombie detection
  - Error recovery
  - Site cleanup

Scheduler

- Sends pilots to batch systems and grid sites
- Three kinds of scheduler
  - CondorG scheduler
    - For most US ATLAS OSG sites
  - Local scheduler
    - BNL(condor) and UTA(PBS)
    - Very efficient and robust
  - Generic scheduler
    - Supports also non-ATLAS OSG VOs and LCG
    - Being extended through OSG Extensions project to support Condor-based pilot factory
      - Move pilot submission from a global submission point to a site-local pilot factory, which itself is globally managed as a Condor glide-in

Panda Monitor

- Apache-based monitor
- Provides uniform I/F for all grid jobs (production and analysis)
- Extensible to other OSG VOs (CHARMM added)
- Three instances running in parallel
- Caching mechanism for better response



Sleipnir - [Panda Production Operations Dashboard]
ファイル(F) 編集(E) 表示(V) お気に入り(B) グループ(G) セキュリティ(I) プロキシ(P) スクリプト(R) ツール(T) ウィンドウ(W) ヘルプ(H)
アドレス http://gridui02.usatla
検索 "in background"
Google(日本語)
Panda Production ...
スタート
Dow...
Sle...
Jane
vaio...
mae...
マイ ...
panda
osg_...
ロ-...
14:32

JavaScript must be enabled in order for you to use this browser.

Configuration

Dashboards: Production DDM Sites & Grids Analysis Physics data Task definition Usage & Quotas TestPilot Pl

4 min old Update

Panda monitor

Panda Production Operations Dashboard

Quick guide, twiki

Panda shift guide and calendar

Jobs - search
Recent running, activated, waiting, assigned, defined, finished, failed jobs
Select analysis, production, test jobs

Quick search
Job
Dataset
Task
File

Summaries
Blocks: days
Errors: days
Nodes: days
Daily usage

Tasks - search
Generic Task Req
EvGen Task Req
CTBsim Task Req
Task list
Task browser

Servers: Panda:OK Panda-dev:OK Panda-CERN:OK Logger:OK DQ2:OK

Tasks assigned to OSG

Jobs updated >12 hrs ago: activated:none running:none
Jobs updated >36 hrs ago: transferring:140

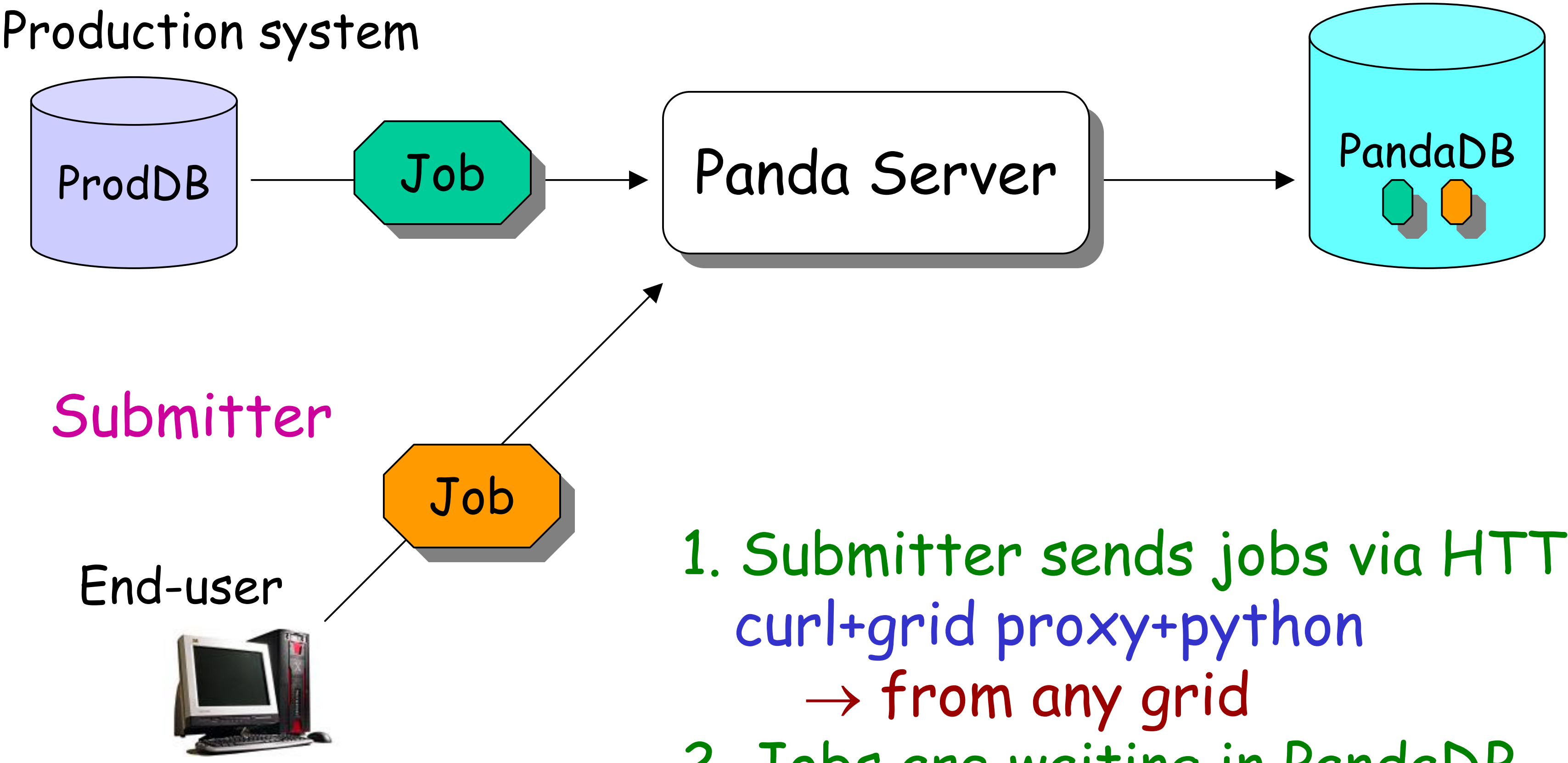
Space available at sites:

Site	GB	As of
MWT2_IU	19040	02-26 14:25
MWT2_UC	27501	02-26 14:19
UTA_SWT2	4510	02-26 13:56

Pilot job requests per hour, last 3 ho

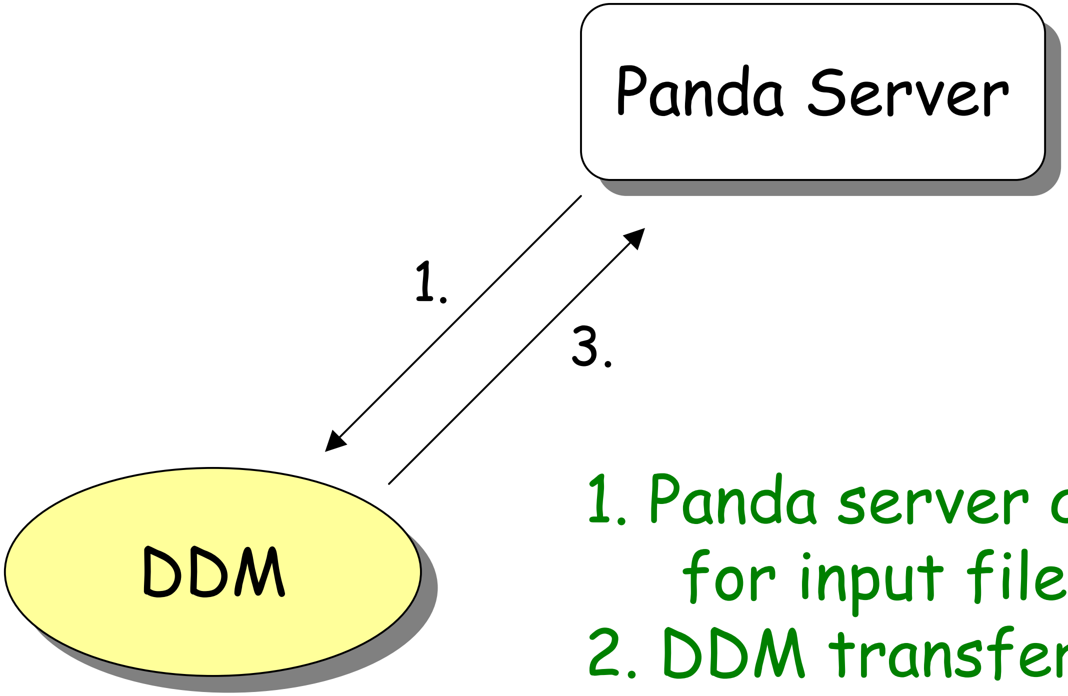
	Productic
ANALY_BNL_ATLAS_1	
ANALY_BNL_ATLAS_2	
ANALY_LONG_BNL_ATLAS	
ANALY_UTA-DPCC	
BNL_ATLAS_1	354
BNL_ATLAS_2	0
BNL_ATLAS_DDM	48
BU_ATLAS_Tier2	195
BU_ATLAS_Tier2o	225
IU_ATLAS_Tier2	65
MWT2_IU	418
MWT2_UC	416
OU_OCHEP_SWT2	206
PROD_SLAC	11
UBC	0
UC_ATLAS_MWT2	0
UC_Teraport	230
UMATLAS	129
UTA-DPCC	215

Typical Workflow (1/3)



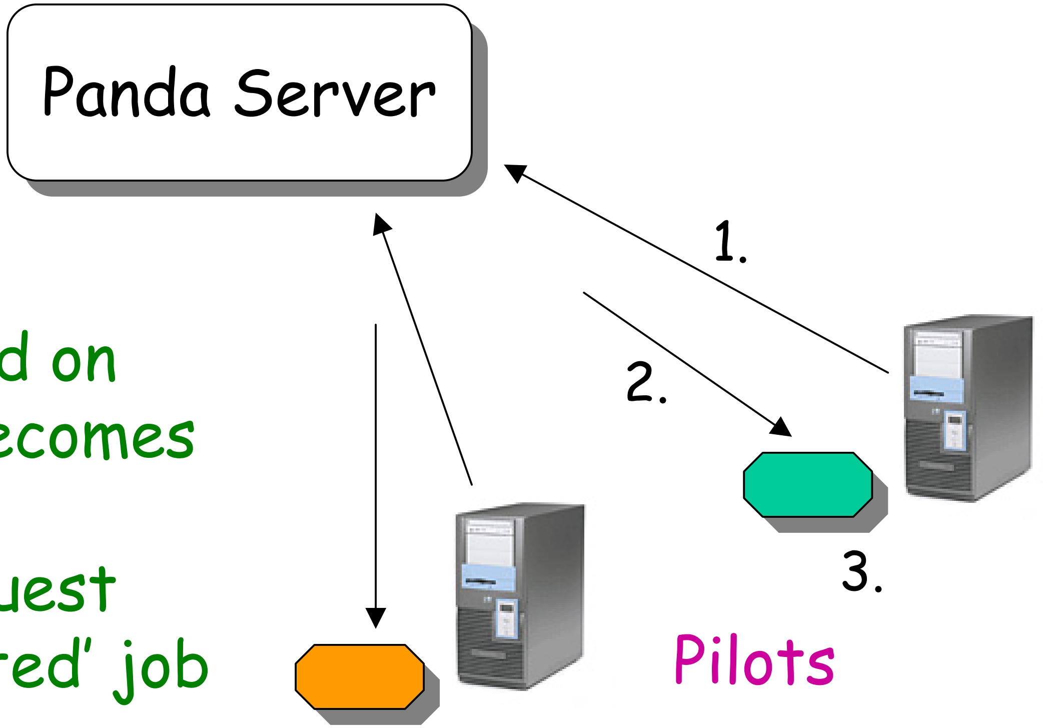
1. Submitter sends jobs via HTTPS
   curl+grid proxy+python
   → from any grid
2. Jobs are waiting in PandaDB

Typical Workflow (2/3)



1. Panda server queues a transfer for input files of jobs
2. DDM transfers files asynchronously
3. DDM sends a notification to panda server as soon as the transfer gets completed
4. Jobs get activated in PandaDB

Typical Workflow (3/3)

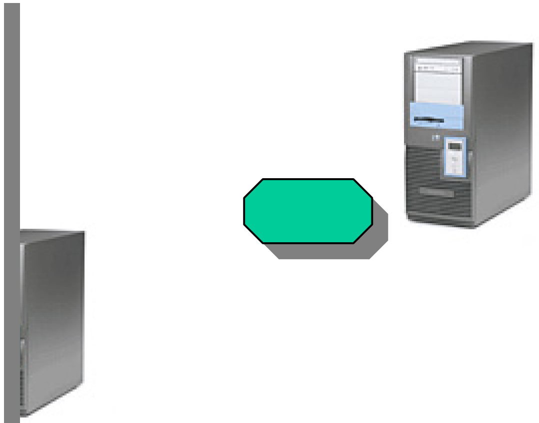


Pilots are pre-scheduled on WNs, and when CPU becomes available each pilot

1. sends an HTTP request
2. receives an 'activated' job as an HTTP response
3. runs the job

Typical Workflow (3/3)

Panda Server



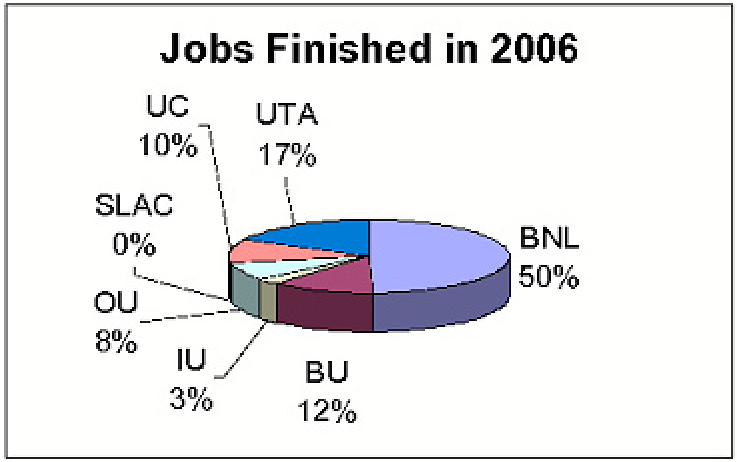
- ➢ Pipeline structure
  - – Data-transfer and job-execution run in parallel
- ➢ Pre-scheduled pilots
  - – pull jobs when CPU's become available

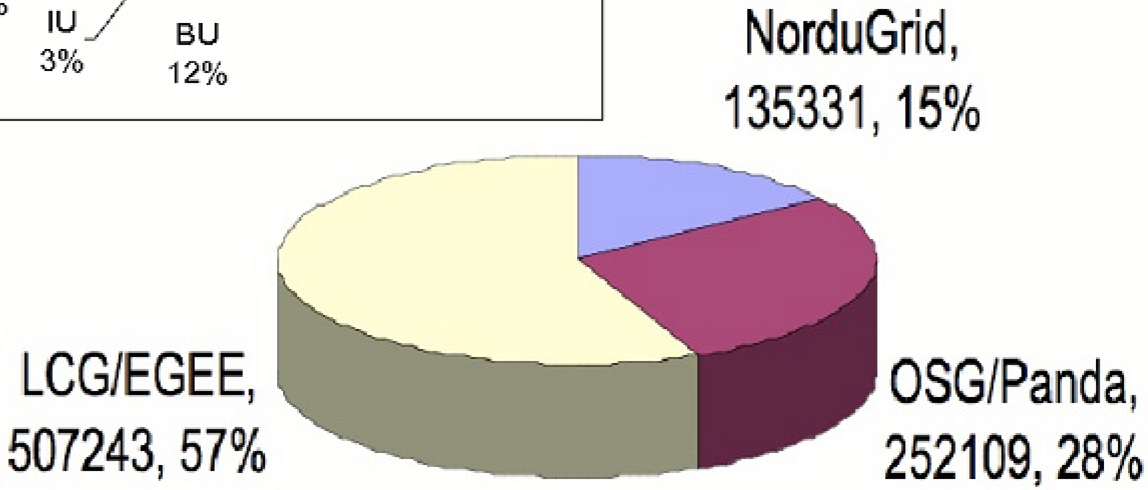
Jobs can run without waiting on WNs

Current Status (1/2)

- ATLAS MC production
  - Computer System Commissioning (CSC) is on going
  - Massive MC samples produced for software validation, physics studies, calibration and commissioning
  - Many hundreds of different physics processes fully simulated with Geant 4
  - More than 10k CPU's participated in this exercise
- CSC production with Panda performing very well
  - All managed US production : ~28% of total ATLAS production
  - Low operation load : single shifter, spends only small fraction of time on Panda issue

Completed ATLAS Production Jobs 2006

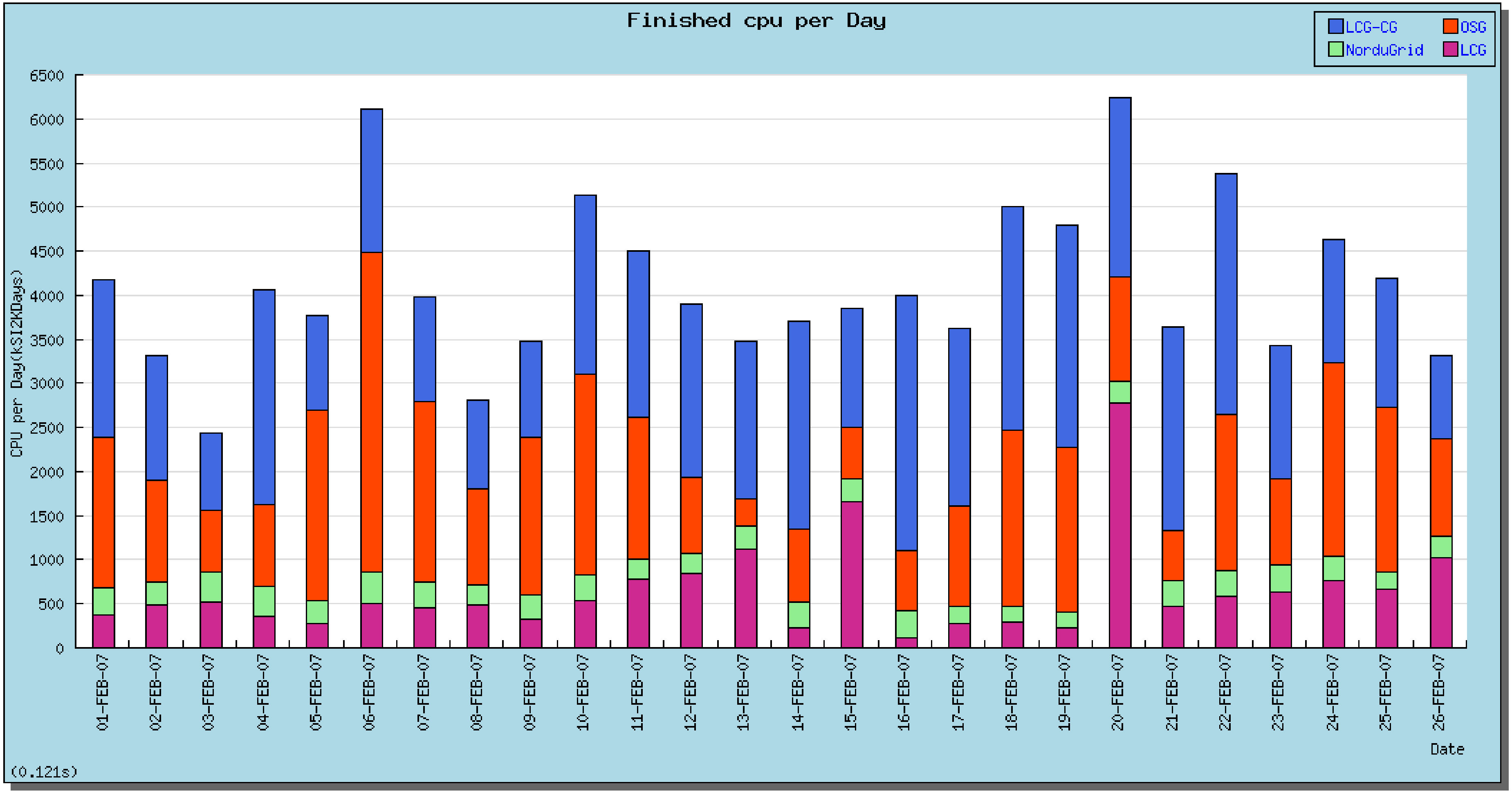




Panda production : 50% of the jobs done on Tier1 facility at BNL
50% done at US ATLAS Tier2 sites

CPU/day for Successful Jobs (Feb 2007)

Current operation scale is ~1/6 of that expected in datataking



Current Status (2/2)

- Distributed Analysis effort
  - Has been in general use since June 2006
  - Popular with users (~100) and has been interested in ATLAS outside US which we're working to satisfy
- Development is not complete and ended. But we don't expect 'big bang' migration because steady operation is important. ATLAS data-taking starts soon.

Near-Term Plans

- Use generic scheduler/pilot system deployed on OSG and LCG to support ATLAS production and analysis across these grids
- Deployment of experiment-neutral Panda as prototype OSG service
  - Drawing on CHARMM experience to improve support for non-ATLAS VOs
- Glide-ins, pilot factory and further Condor integration
  - Through OSG extensions project, collaborating with Condor and CMS
- Introduce partitioning in the Panda server's LAMP stack for scalability

Conclusions

- The Panda project initiated 18 months ago has been successful in US ATLAS
  - Used for US production and analysis, utilizing resources and personnel efficiently
- Panda provides stable and robust services for coming data-taking of ATLAS experiment
  - No big-bang migration
- Panda is now being extended further
  - OSG: non-ATLAS users, extensions project
  - ATLAS: deployment across LCG and OSG