Open Science Grid Use By DZero

OSG Workshop, São Paulo
December 10, 2010

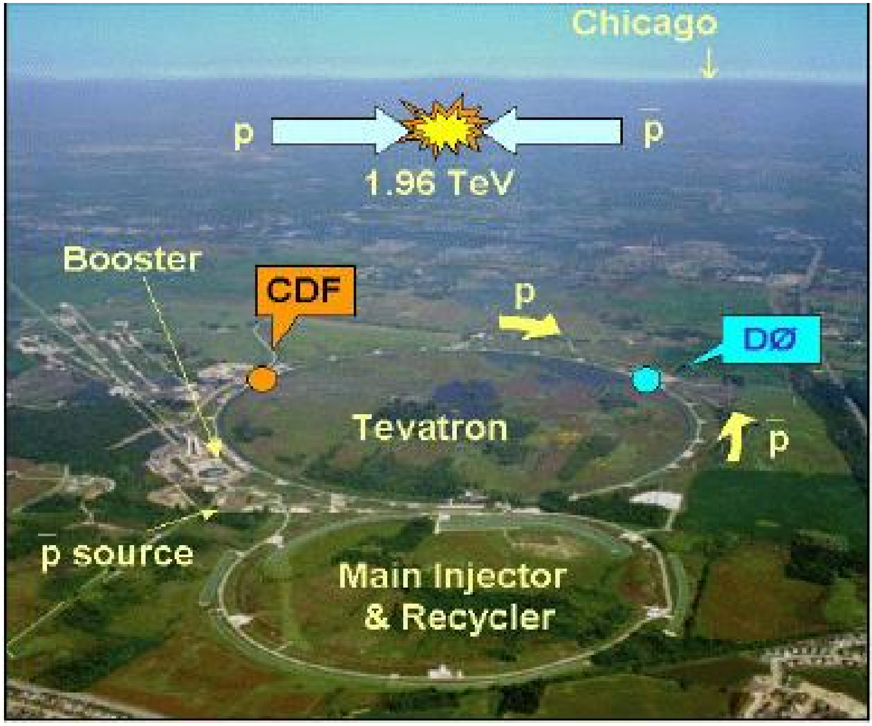
Joel Snow
Langston University

Outline

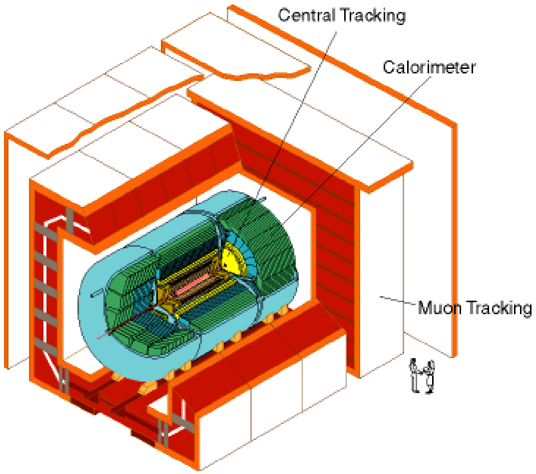
- What is DZero?
- Why does DZero use OSG?
- Interoperability with OSG and LCG
- How does DZero use OSG?
- The Monte Carlo use case
  - MC Production System
  - MC Production Results
- Summary

What is the DZero Experiment?

- Particle physics experiment
  - At the Tevatron Accelerator at Fermilab in Batavia, IL, USA
  - Collides 1 Tev protons with 1 Tev antiprotons
- Global enterprise
  - Collaborators are disbursed over 4 continents

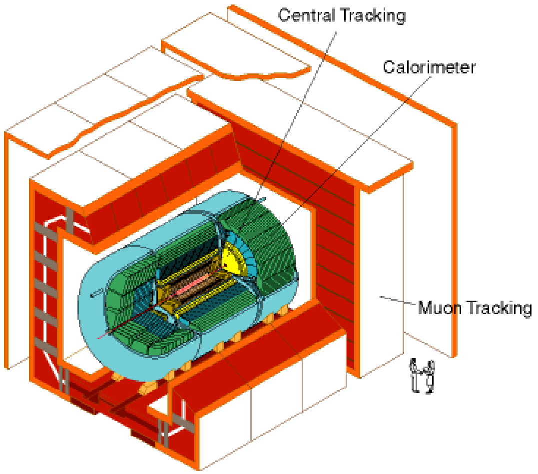


DZero Experiment

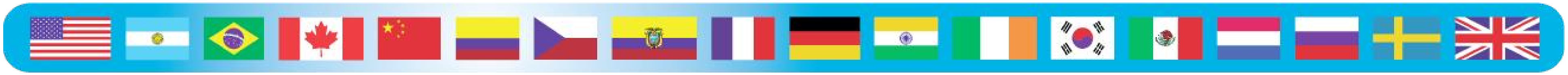


- Detector 30'x30'x50', 5000 tons
- ~1,000,000 data channels
- Inspects 1.7 million p anti-p collisions/sec
- Records ~100 events/sec
- Data flow 20MB/sec.
- 300,000 GB of data recorded/year
- 7.7 billion events collected in Run II to date
- Took data 1992-1996, upgrade 1996-2001, running nearly continuously since

DZero Experiment



- Global enterprise
  - 491 physicists
  - 19 countries on 4 continents
  - 86 institutions (37 in U.S.)
- 133 Run I and 210 Run II publications to date
- Will run into 2011, perhaps longer
- Expect dataset to double if extended to 2014
- Challenging as resources migrate to LHC experiments



Scenario

- Simulation data (MC) crucial to physics analysis
- Tevatron luminosity and hence raw data volume is at record levels
- Challenge for analysts and production
- Personnel & computing resources migrating to LHC experiments
- DZero coping strategy
  - Increase automation
  - Leverage resources and support

DZero Evolution

- Mature experiment, but nimble
  - history of adopting innovative technologies
    - distributed data handling – SAM
    - early adopter of the grid for production - SAMGrid
  - significant investment in these technologies
- Grid technology allows opportunistic usage
  - DZero can mix “traditional” dedicated and opportunistic resources
- Grid interoperability
  - Leverages resources and support, reduces personnel needs per CPU hour

Sequential data Access via Metadata

- Fermilab system first used by DZero
- SAM distributed data handling system predates the grid
- Set of servers working together to store and retrieve files and metadata
- Permanent storage and local disk caches
- Database tracks location, metadata of files, job processing history
- Delivers files to jobs (using GridFTP over WAN), provides job submission capabilities

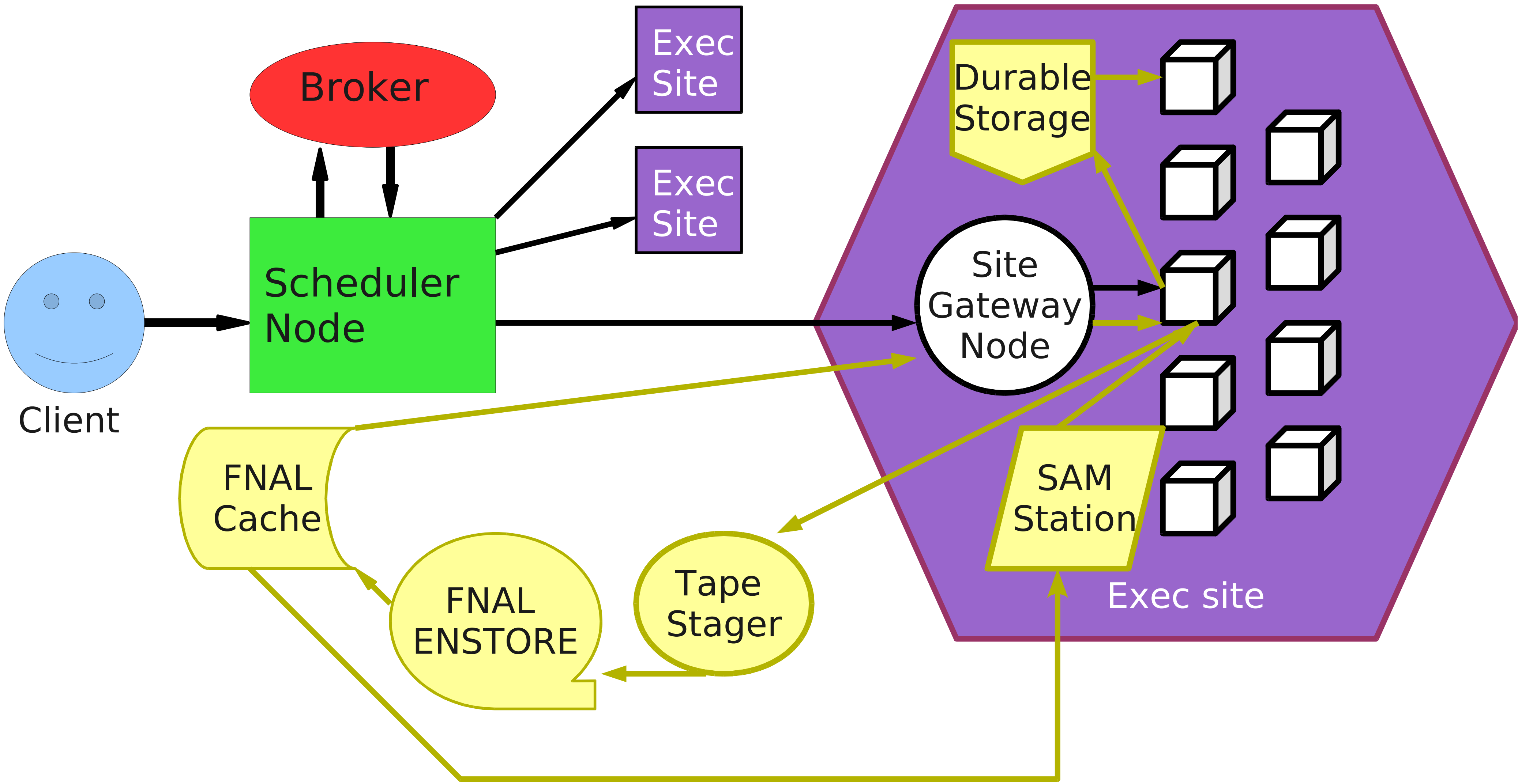
SAMGrid

- Fermilab developed grid first used by DZero for global MC production in 2004
- SAMGrid = SAM + Job and Information Management (JIM) components
- Provides the user with transparent remote job submission, data processing and status monitoring.
- VDT based (Globus + Condor)
- Logically consists of
  - Multiple execution sites
  - Resource selector
  - Multiple Job Submission (Scheduler) sites
  - Multiple Clients (User Interface) to Submission site.

SAMGrid Operation

- User submits job request to queuing node (based on Condor scheduler) via remote client (based on Condor client commands).
- Jobs are matched and submitted to execution sites (based on on Globus gatekeeper/jobmanager).
- At exec site job requests are split into multiple job instances (for MC 250 events/job)
- Job instances submitted to a local batch system *or to another grid*.
- Exec site triggers data delivery (binary, control, input data) and controls data traffic shaping.

SAMGrid Components



SAMGrid Reflections

- Enabled Dzero's use of opportunistic computing cycles
- Very productive for Monte Carlo
- Deployment proved limited in scope
  - Sites require operational manpower and expert support
  - People power and lab support migrating to LHC experiments
- Still in use, but more computing needed!

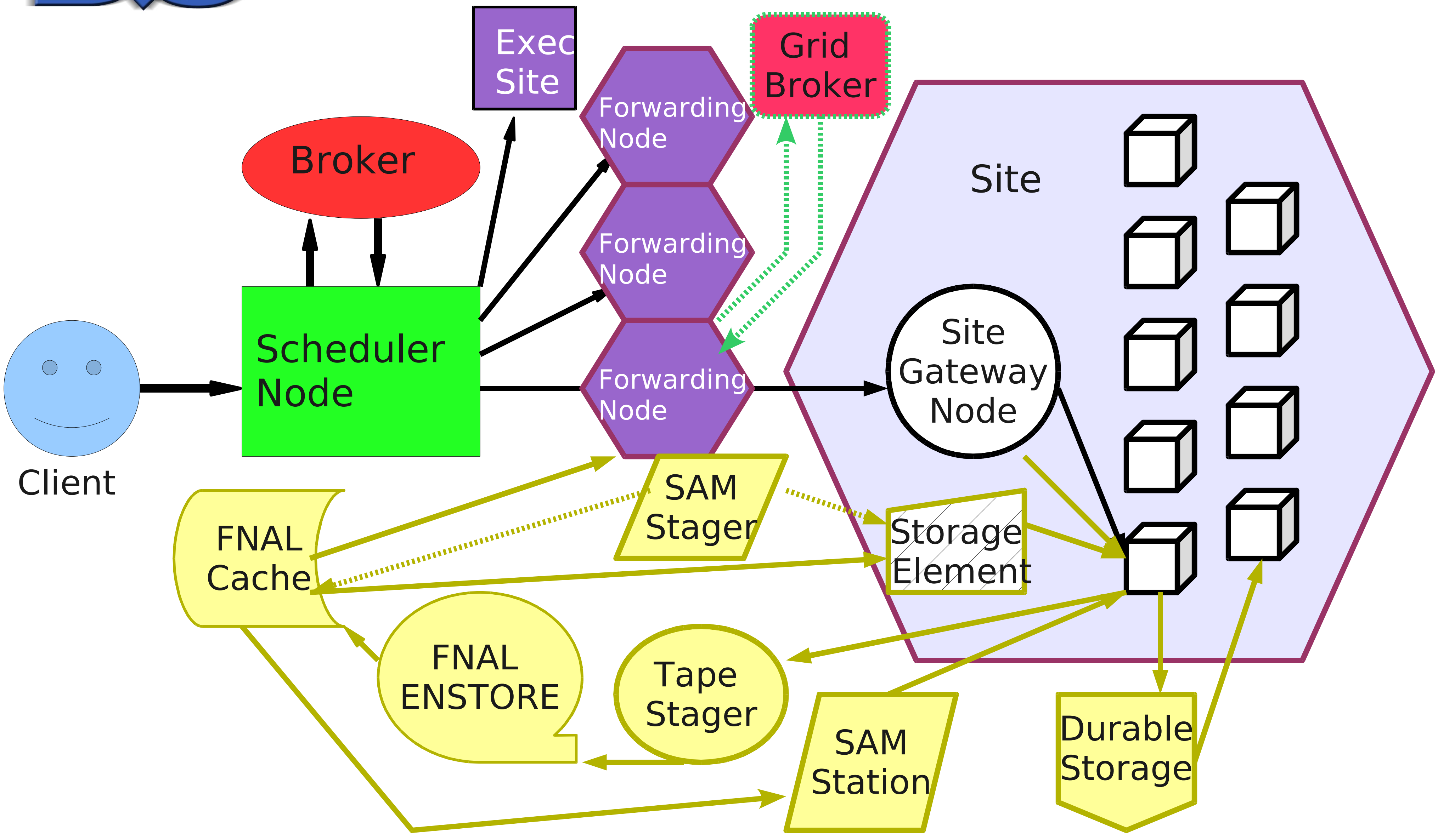
Why Does DZero Use The OSG?

- Dzero has a huge amount of data
- Dzero has limited computing and human resources
- SAMGrid was not enough
- Other grids like OSG and LCG have resources available
  - provides opportunistic job slots ✔
  - comes with support ✔

SAMGrid Interoperability

- As Open Science Grid (OSG) and LHC Computing Grid (LCG) became operational it was desirable to leverage these resources for DZero
- FNAL and DZero developed and deployed SAMGrid interoperability with both LCG and OSG resources
- Execution site acts as a Forwarding node
  - packages SAMGrid jobs for OSG/LCG job submission

SAMGrid OSG/LCG



How Does DZero Use The OSG?

- All raw data is first reconstructed using OSG facilities
  - Only done at OSG sites at Fermilab
  - Jobs submitted using SAMGrid framework
- A large fraction of simulation (Monte Carlo) is done using OSG facilities
  - Jobs submitted using SAMGrid framework
- A small number of specialized physics analyses use OSG facilities
  - Not submitted using the SAMGrid framework
- LCG jobs are submitted through an OSG glidein factory at Fermilab

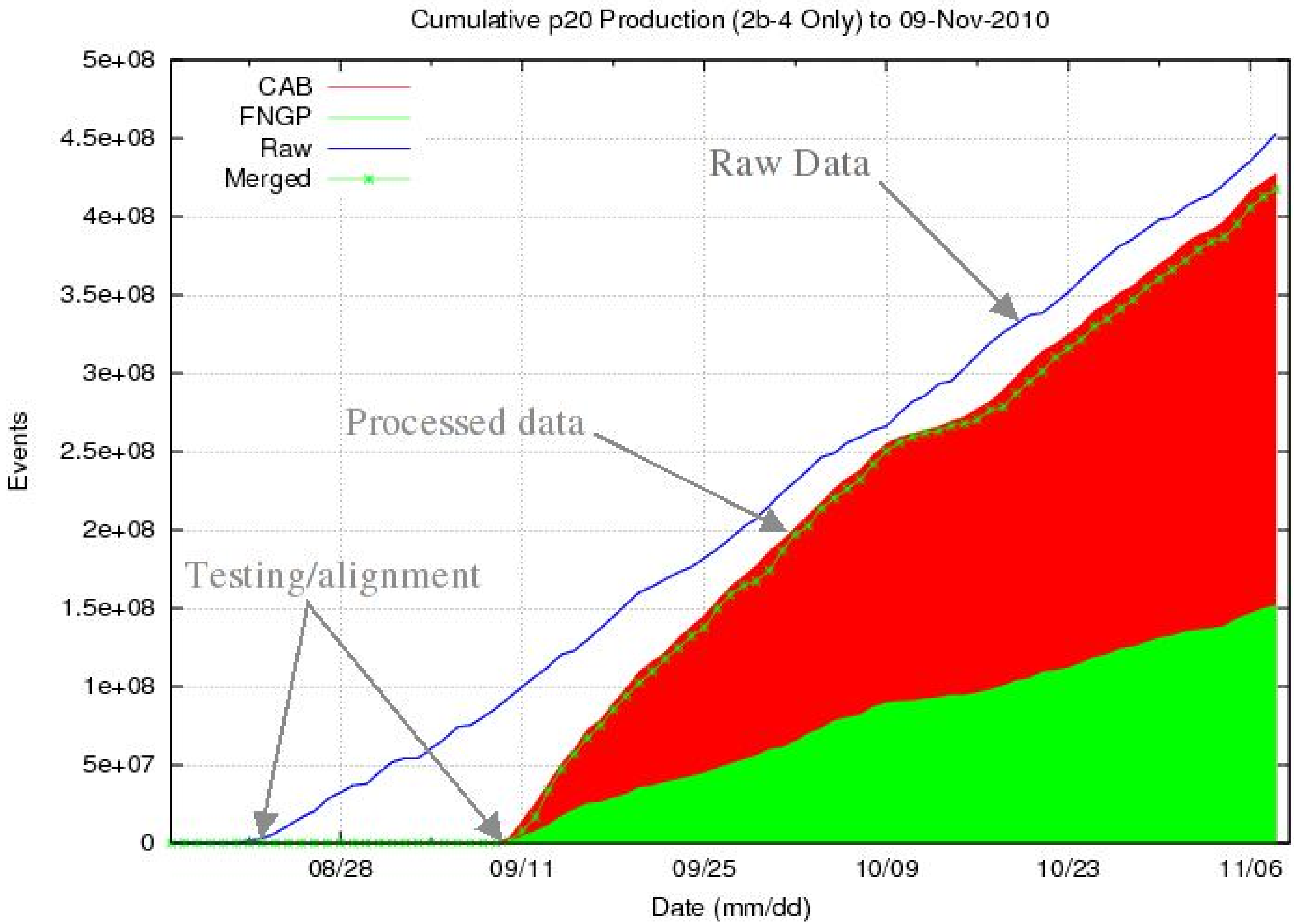
How Does DZero Use The OSG?

Dzero VO Gratia statistics for November 2010

	# Jobs	Wall Time (h)	Success %	Sites
Data Production	80,290	1,161,569.8	91.6	2
MC Production	242,009	1,472,122.7	74.7	25
Physics Analyses	18	144.5	100.0	1
Total	322,317	2,633,837.0	78.9	25

2,633,837 hours = 300 years!

Data Production on the OSG



Monte Carlo Production on the OSG

- Request system
- MC Applications
- Job flow
- Data flow
- Automated grid job submission
- Monitoring
- Production results

Production System

- MC production gets work from the SAM Request System
  - Physics groups' MC requests are parametrized and prioritized

MC Request Summary

Group	Weight	Request Total	Processed Events	Weighted Events	Next Job
algo	1	2000000	1000000	1000000	0
bphysics	1	25400000	17900000	17900000	0
dzero	4	737789248	554765470	138691367	0
higgs	1	218478000	190253000	190253000	94799
jes	4	35400001	27000000	6750000	0
np	1	58849999	41385001	41385001	0
qcd	1	6700000	6600000	6600000	0
test	1	2198001	1641000	1641000	0
top	1	221785000	162160000	162160000	94834
wz	1	11580000	10415000	10415000	0

The Next Request to be processed is Request ID = 94834

Updated Sat Nov 29 11:00:24 CST 2008

MC Requests 1

Updated Sat Nov 29 11:00:24 CST 2008

Request ID	Status	Group	User	Priority	# Events	Description	Cardfile Vers. Dir. Prod. Decay
95054	approved	top	mackin	5	100000	p20 pythia t+t->incl m_t=185 gev with non-standard br	pythia --- v01-01-37 --- top --- t+t --- incl_BR.n
95053	approved	top	mackin	5	200000	p20 pythia t+t->incl m_t=185 gev with non-standard br	pythia --- v01-01-37 --- top --- t+t --- incl_BR.n
95052	approved	top	mackin	5	200000	p20 pythia t+t->incl m_t=185 gev with non-standard br	pythia --- v01-01-37 --- top --- t+t --- incl_BR.n
95051	approved	top	mackin	5	100000	p20 pythia t+t->incl m_t=180 gev with non-standard br	pythia --- v01-01-37 --- top --- t+t --- incl_BR.n
95050	approved	top	mackin	5	200000	p20 pythia t+t->incl m_t=180 gev with non-standard br	pythia --- v01-01-37 --- top --- t+t --- incl_BR.n
95049	approved	top	mackin	5	200000	p20 pythia t+t->incl m_t=180 gev with non-standard br	pythia --- v01-01-37 --- top --- t+t --- incl_BR.n

Production System

- Request in the form of a Python dictionary


```
Request_94775({
        'requestId' : 94775L,
      'requestType' : 'simulation',
    'requestStatus' : 'new',
          'archive' : SamBoolean('FALSE'),
         'comments' : 'pythia hl+b->bb+b m_hl=100 gev',
            'group' : 'higgs',
   'numberOfEvents' : 150000L,
         'priority' : 5L,
    'statusHistory' :
RequestStatusHistory([RequestStatusHistoryEntry({
          'byWhom' : UserIdentifier(userName='mackin'),
            'date' : SamTime(1227540201.0),
   'requestStatus' : 'new',
   })]),
   'userIdentifier' : UserIdentifier(userName='mackin'),
           'params' : Params({
          'global' : CaseInsensitiveDictionary({
          'datatier' : 'reconstructed',
       'description' : 'Pythia hl+b->bb+b m_hl=100 GeV',
         'groupname' : 'higgs',
             'phase' : 'mcp20',
   'producedforname' : 'mackin',
         'requestid' : '94775',
           'runtype' : 'Monte Carlo',
            'stream' : 'notstreamed',
   }),
    'digitized' : CaseInsensitiveDictionary({
'calorimeternoise' : 'off',
       'd0release' : 'p20.09.03',
'frameworkrcpname' : 'runD0Sim_noCalNoise_run2b.rcp',
    'mergeminbias' : 'on',
    'minbidataset' :
'zerob_p20_09_03_RunIIbMC_online_0sup_only_sample_sept06_shutdown2007_warmcellfi
x',
        'minbiopt' : 'Fixed',
        'numminbi' : '1.0',
}),
    'generated' : CaseInsensitiveDictionary({
    'cardfiledir' : 'higgs',
'cardfileversion' : 'v01-01-00',
'collisionenergy' : '1960.0',
      'd0release' : 'p20.09.03',
          'decay' : '3b_sm.n',
          'etagt' : '-5.0',
          'etalt' : '5.0',
      'generator' : 'pythia',
     'higgslmass' : '100.0',
     'pdflibfunc' : 'LHPDFCTEQ6L1',
     'production' : 'hl+b',
           'ptgt' : '15.0',
           'ptlt' : '-1.0',
        'topmass' : '170.0',
      'useevtgen' : 'on',
}),
'reconstructed' : CaseInsensitiveDictionary({
       'appfamily' : 'reconstruction',
         'appname' : 'd0reco',
      'appversion' : 'p20.09.03',
       'd0release' : 'p20.09.03',
'frameworkrcpname' : 'runD0reco_mc.rcp',
}),
    'simulated' : CaseInsensitiveDictionary({
            'd0release' : 'p20.09.03',
             'geometry' : 'plate-run2b',
'keepparticlecalenergy' : 'off',
}),
}),
})
```

MC Applications

- Typical request has 4 phases – 1 appl. each
  - Generator – physics of interest is created
  - Simulator – propagation of particles of interest through the detector
  - Digitizer – Put simulated data in the form of raw data and overlay with generic background
  - Reconstruction – Reconstruct with first pass data analysis code
- Metadata of all phases saved in SAM
  - Typically only the output of the last phase is saved on tape at FNAL

MC Grid Job Flow

- SAMGrid jobs are broken into 250 event chunks at the Forwarding node for submission to the Condor_g system
  - Execution time trade-off to maximize usable sites
- Output file size too small for efficient tape storage (20-30MB)
  - Merged in separate grid job
- The 10k event merged files (1GB) are stored on tape via SAM and unmerged files are deleted from durable storage.

MC Production Grid Job Data Flow

- Bootstrap executable from Forwarding node - 3MB
- Initial environment/utility files from Forwrding node - 20MB
- Applications and execution environment from SAM cache - 800MB
- Optional input data file, overlay files from SAM cache - 200MB-1GB + 300-500MB
- For OSG or LCG jobs no VO specific pre-installed software required at the job site
- Output data file stored in “durable location” until merged

MC Merge Grid Job Data Flow

- Bootstrap executable from Forwarding node - 3MB
- Initial environment/utility files from Forwarding node - 20MB
- Applications and execution environment from SAM cache - 800MB
- Files to be merged from durable location - 1GB
- Output file stored on tape via SAM – 1GB

Data Transport Issues

- OSG jobs use WAN transport to workers from remote SAM caches
  - Pro: No VO specific software pre-installed at job site – great site selection flexibility
  - Con: WAN transport less reliable than LAN
    - Less than optimum job efficiencies
- Use of local OSG SE's as SAM caches mitigates problem
  - 9 OSG SE's in use; space managed by SAM
  - significant improvement in efficiency seen
  - Dzero first to use OSG opportunistic storage

Automatic Monte Carlo Request Processing

- From approved request to final data storage
- Easy to use – minimizes manpower needs
- Site independent
  - deploy for any grid site (SAMGrid, OSG, LCG)
  - capable of managing many sites
- Handle recovery of common failures
- Integrated with existing MC request priority protocol

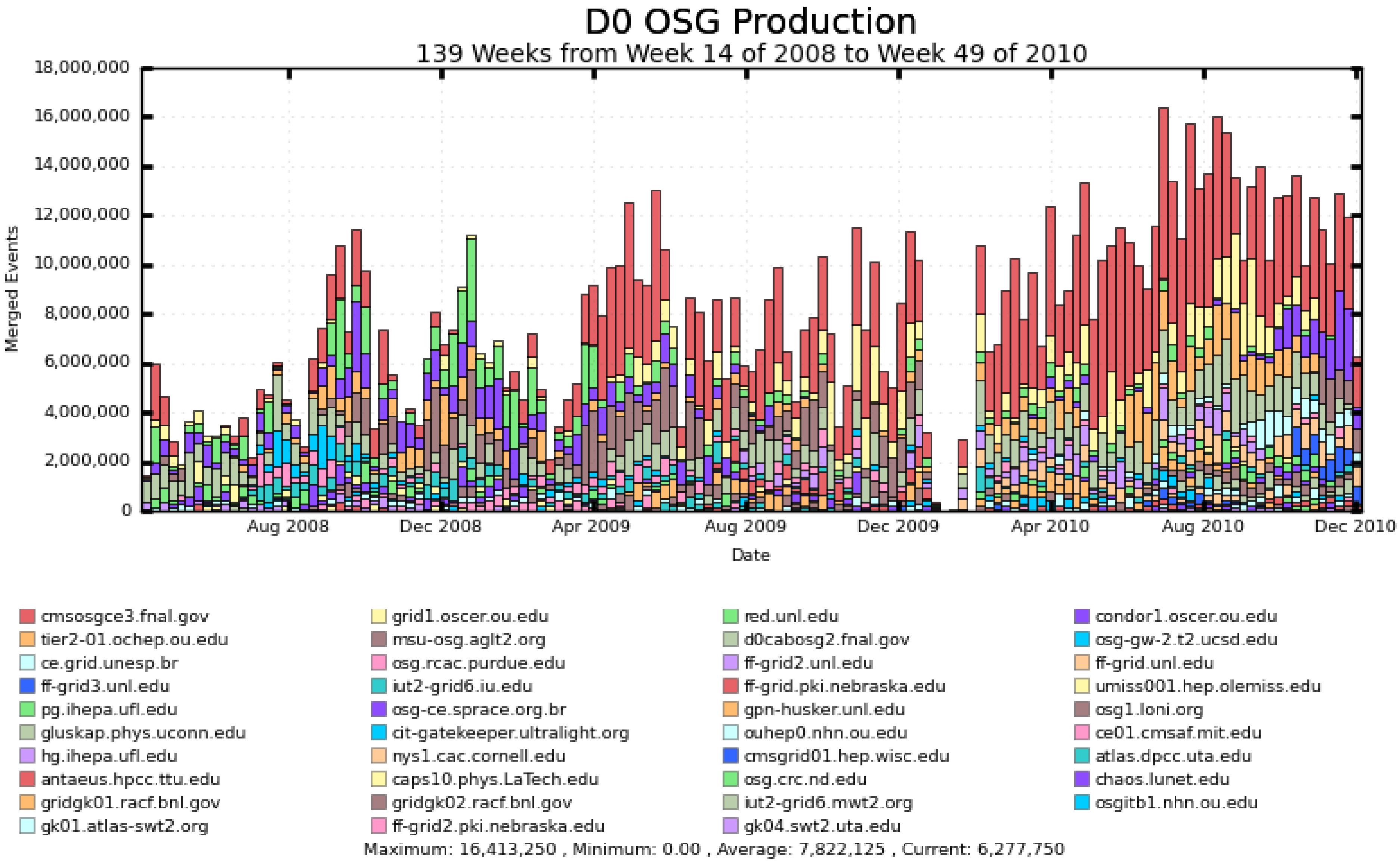
Auto MC System Components

- SAM Client for DB queries
- JIM for job submission and monitoring
- Daemon
  - periodically awakens to do work
- Local database
  - request processing data and history
- Grid credentials

Job Monitoring and Status

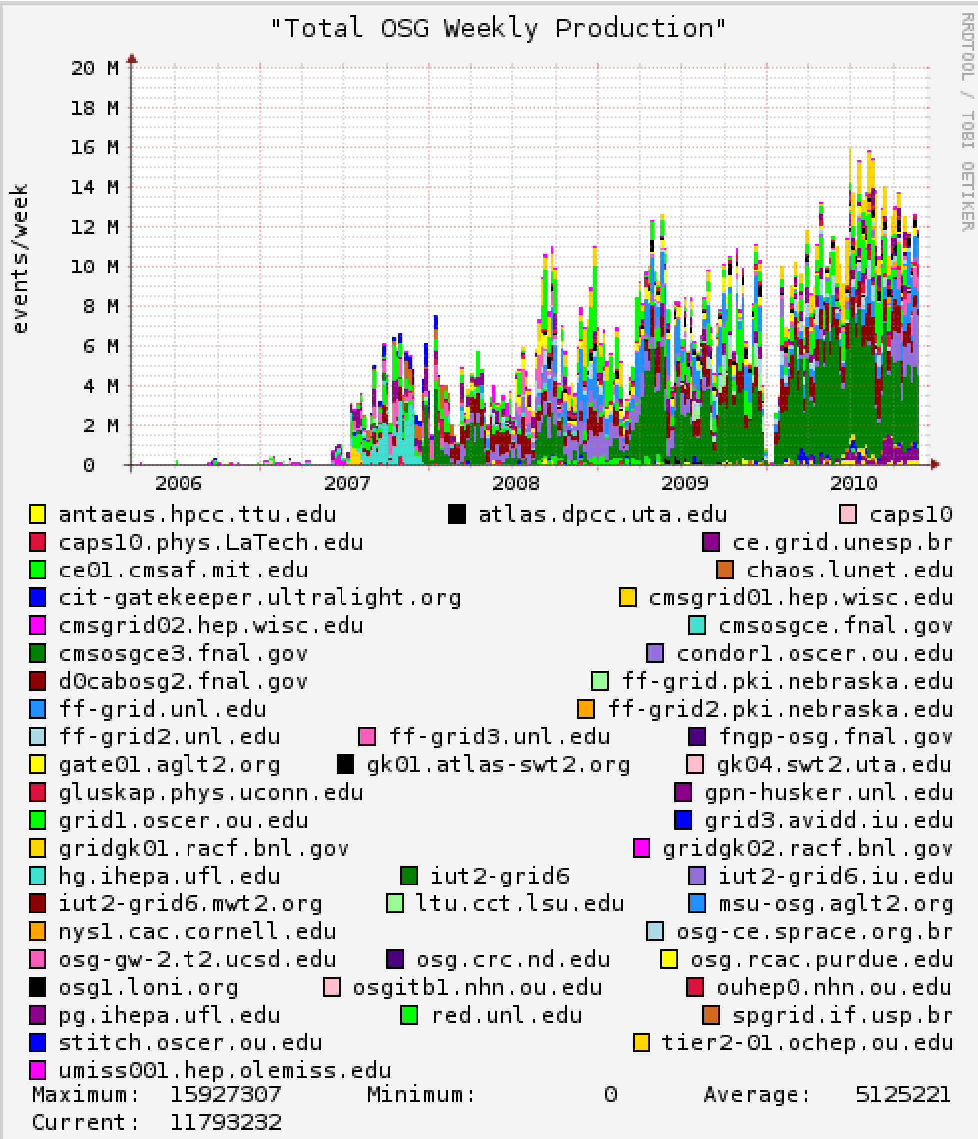
- OSG tools
  - MyOSG (http://myosg.grid.iu.edu)
    - RSV, GIP, Gratia, ...
- Condor tools
- SAM database
  - Request status, some job info, and file status
- SAMGrid databases
  - Jobs instrumented to report history
- All are needed

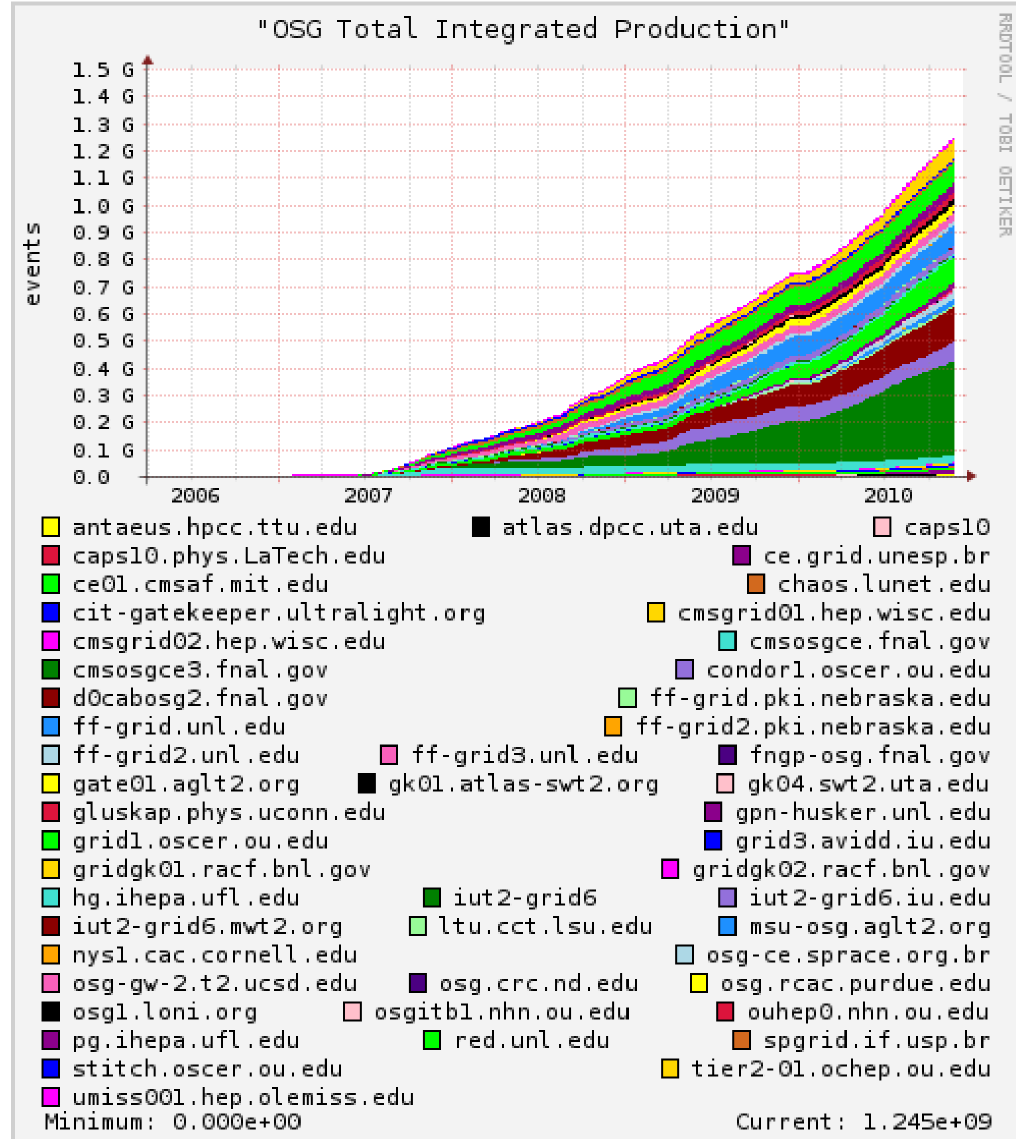
MC OSG Production Results



MC OSG Production Results

April 1, 2006 - December 1, 2010

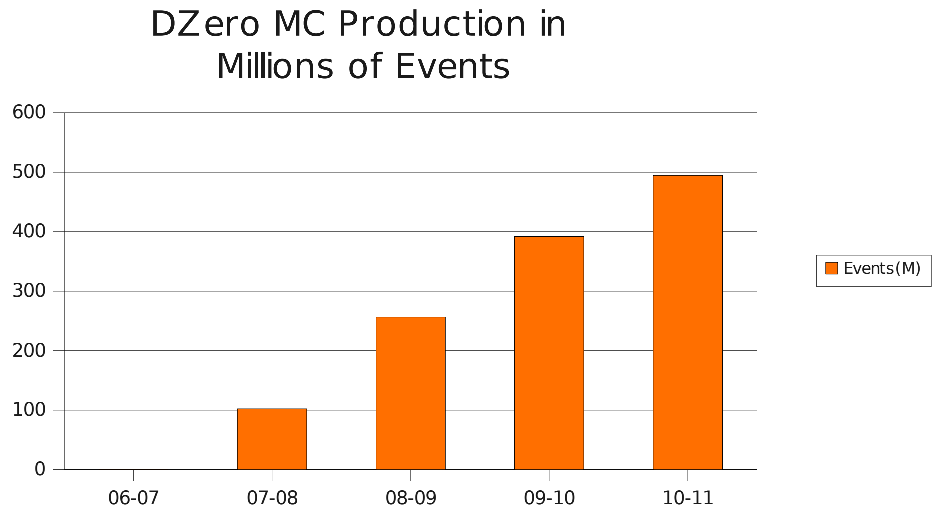




Cumulative since April 1, 2006

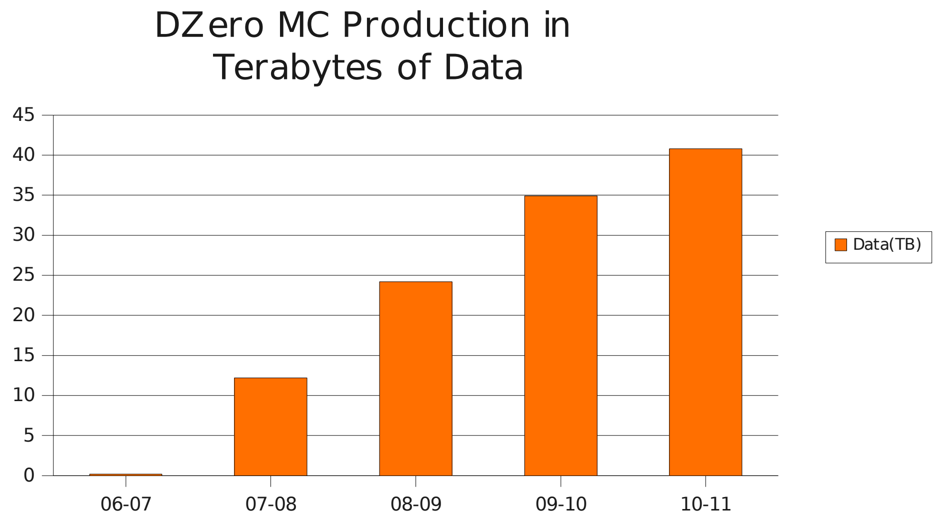
Production Results Last 5 Years

DZero OSG MC Production in Millions of Events



Period	Events (M)
2010/04/01-2010/12/01	494.5
2009/04/01-2010/04/01	391.9
2008/04/01-2009/04/01	256.4
2007/04/01-2008/04/01	102.3
2006/04/01-2007/04/01	1.1

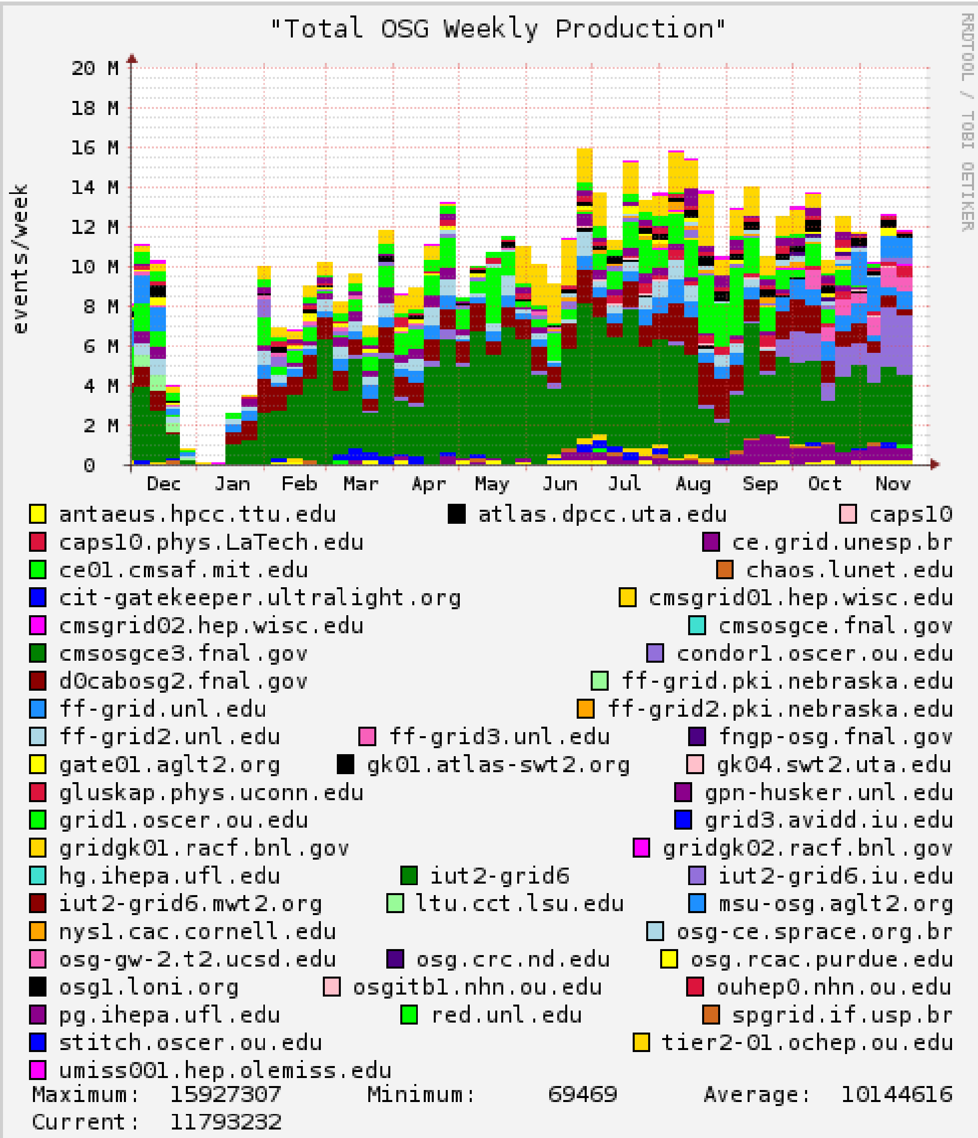
DZero MC Production in Terabytes of Data

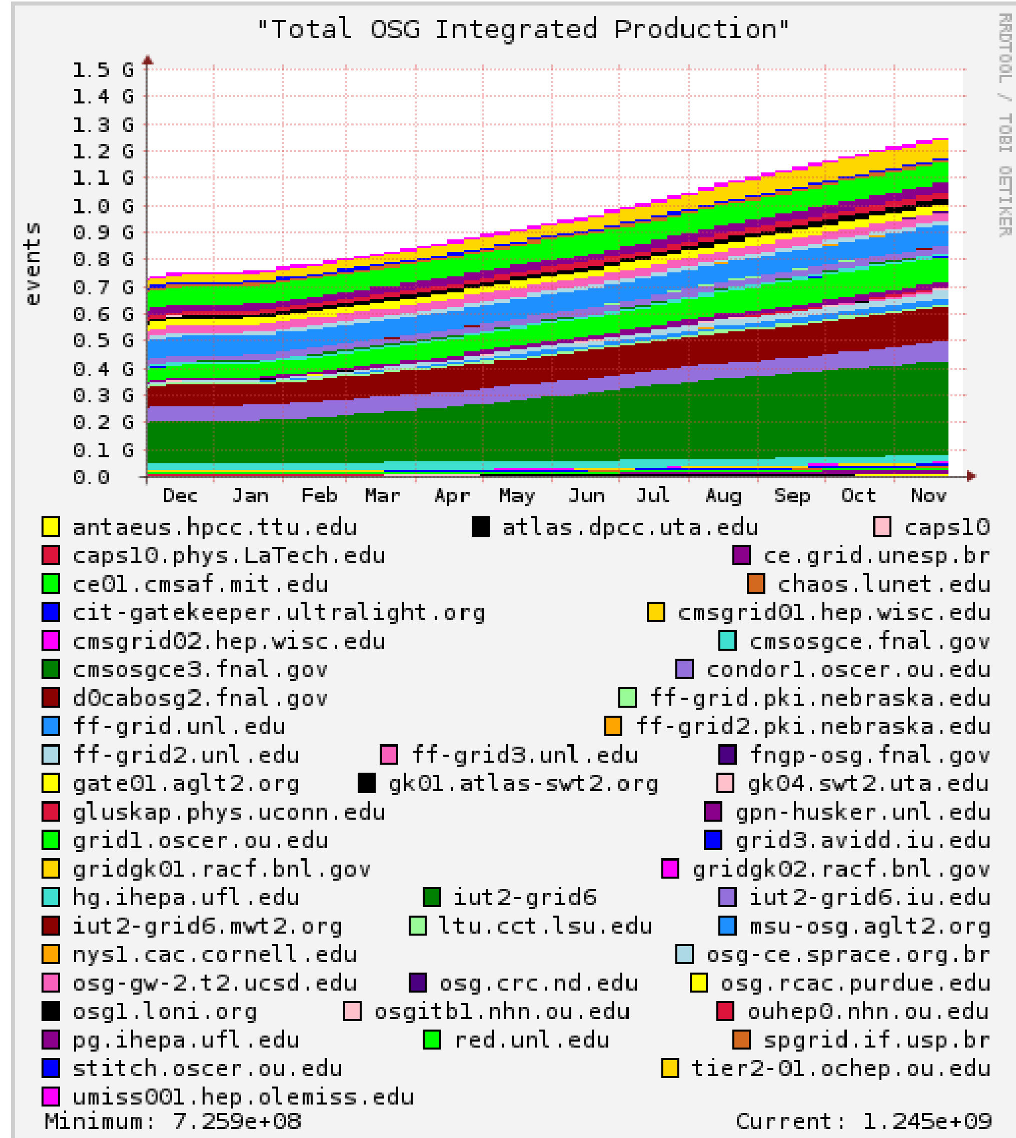


Period	Data (TB)
2010/04/01-2010/12/01	40.8
2009/04/01-2010/04/01	34.9
2008/04/01-2009/04/01	24.4
2007/04/01-2008/04/01	12.2
2006/04/01-2007/04/01	0.2

MC OSG Production Results

Dec. 1, 2009 – Dec. 1, 2010

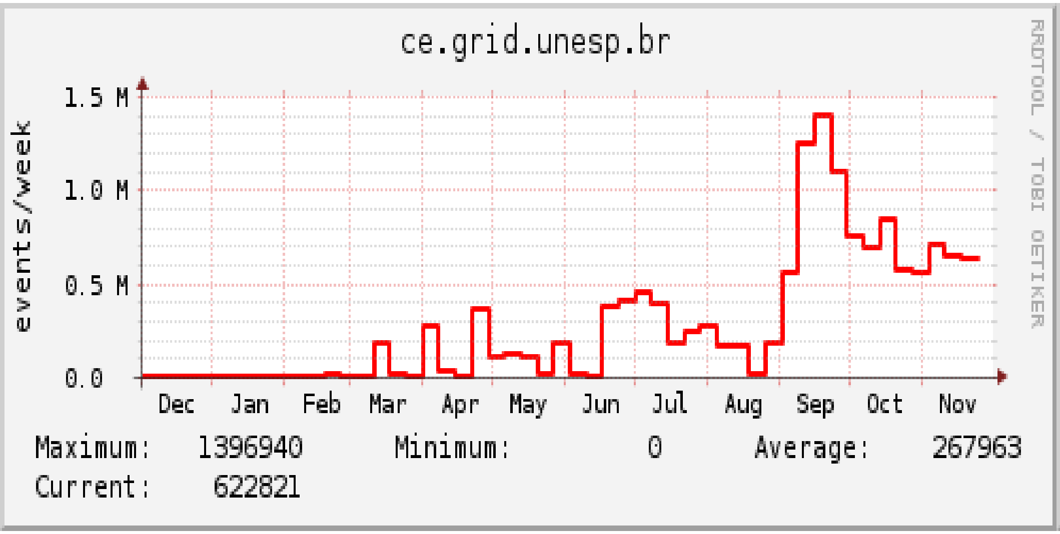


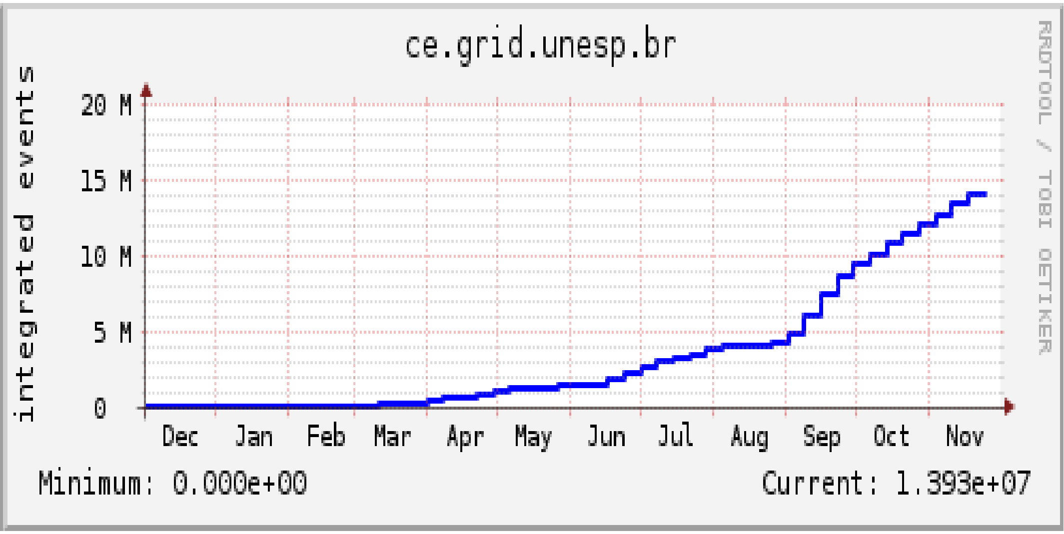


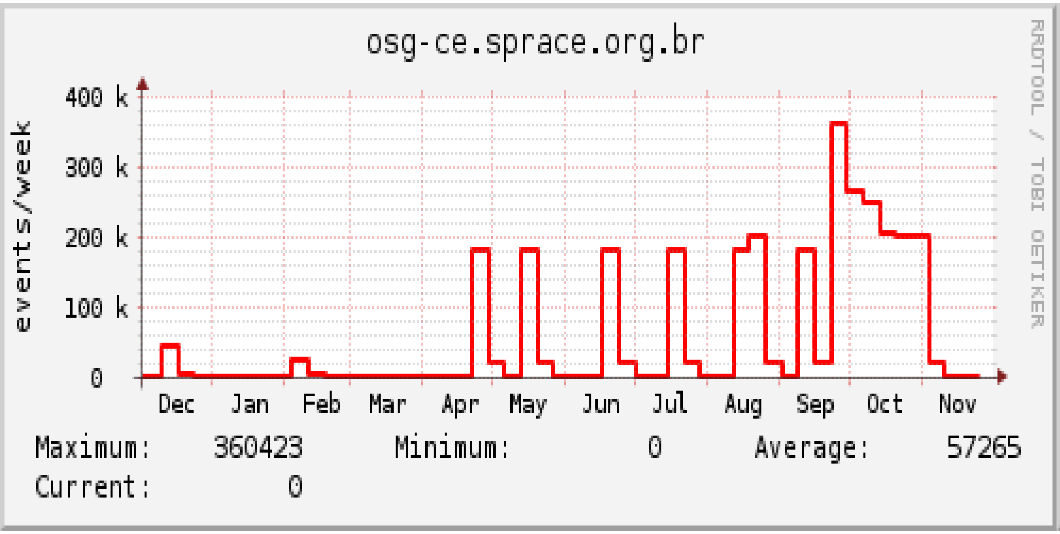
Cumulative since April 1, 2006

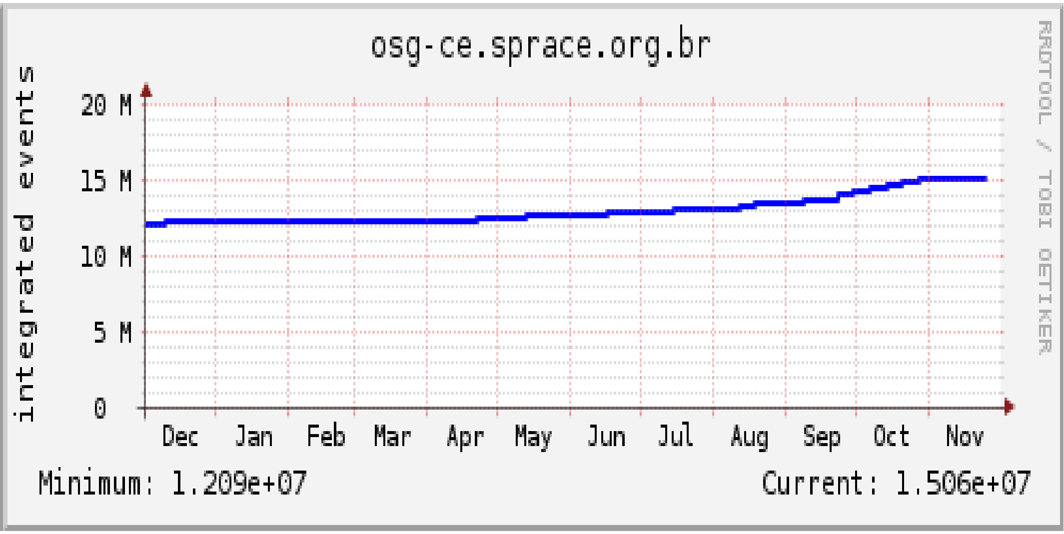
MC OSG Production Results

Dec. 1, 2009 – Dec. 1, 2010









Cumulative since April 1, 2006

Summary

- DØ is very dependent on OSG technology, infrastructure, and support
  - Data production, MC production, Analysis
  - MC production
    - Almost all opportunistic batch slots
    - Heavy user of opportunistic storage
- Leveraging OSG resources has proven a great success for Dzero
  - Data production able to keep up with record Tevatron luminosity and the resulting explosion of data
  - MC production able to provide all needed simulation data for physics analyses