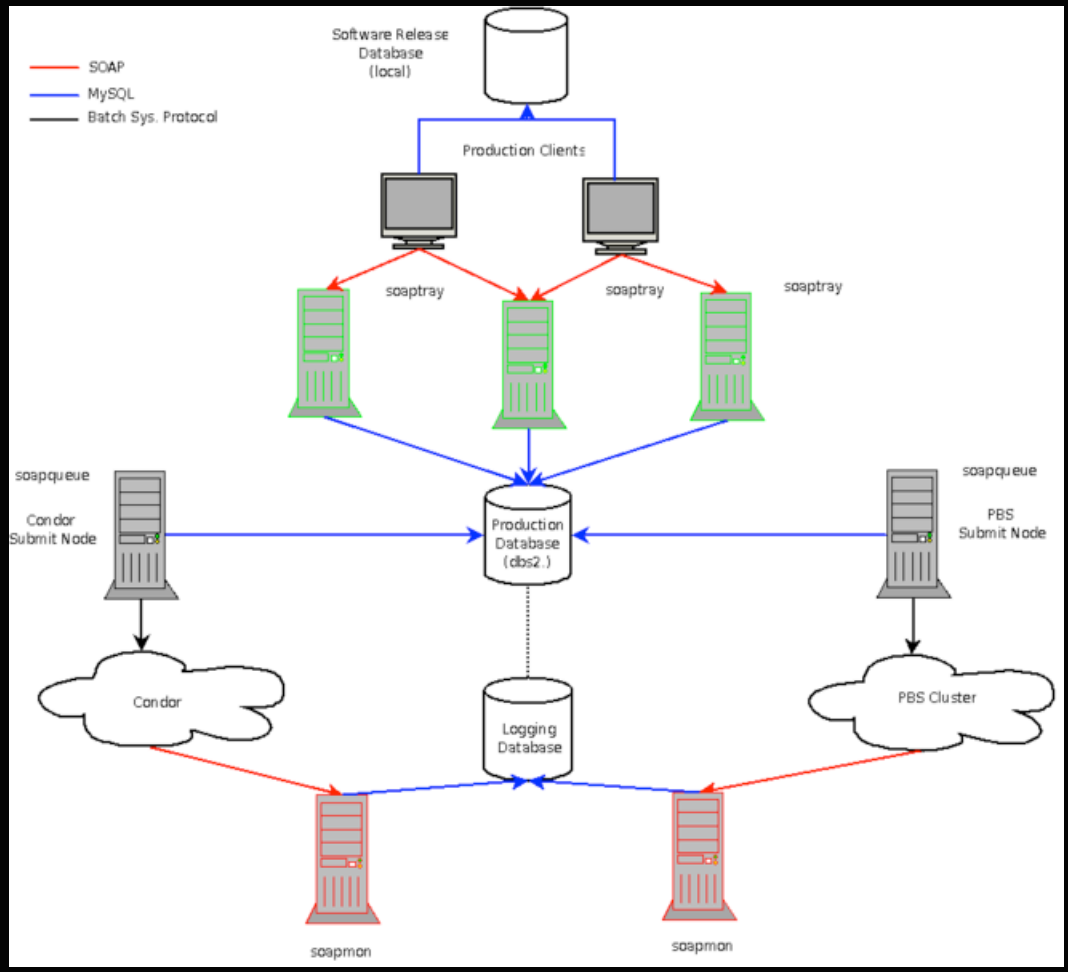
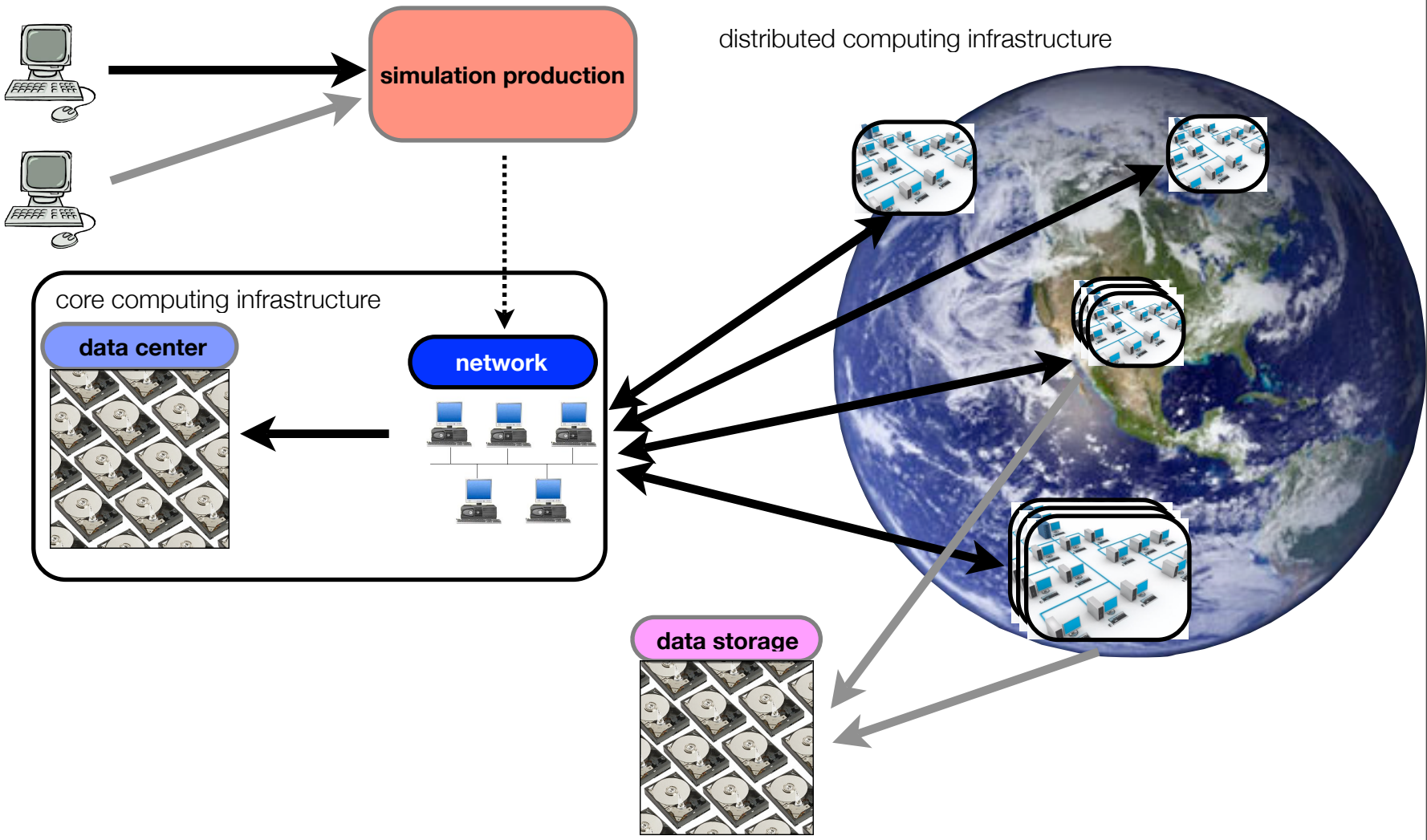
iceprod

Production Database
- global database
- store production history
- configured module parameters

Server daemons
- accept dataset requests from client
- provide job management
- error handling
- separate daemons handle
  - dataset submission,
  - queue/job management & monitoring
  - garbage collection

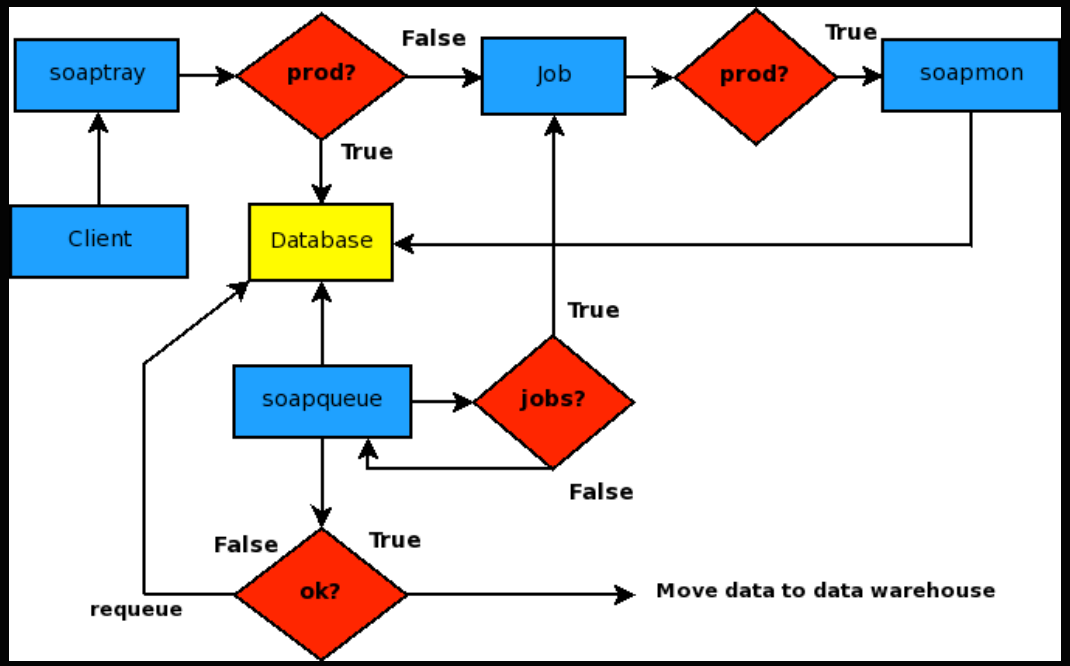


quick overview @ high level : how it runs



iceprod daemons

soaptray listens for incoming requests from client
soapqueue checks queue for jobs to process
soapmon receives monitoring updates from jobs
soapdh handles data movement when jobs complete
soaphisto merges and displays verification histograms

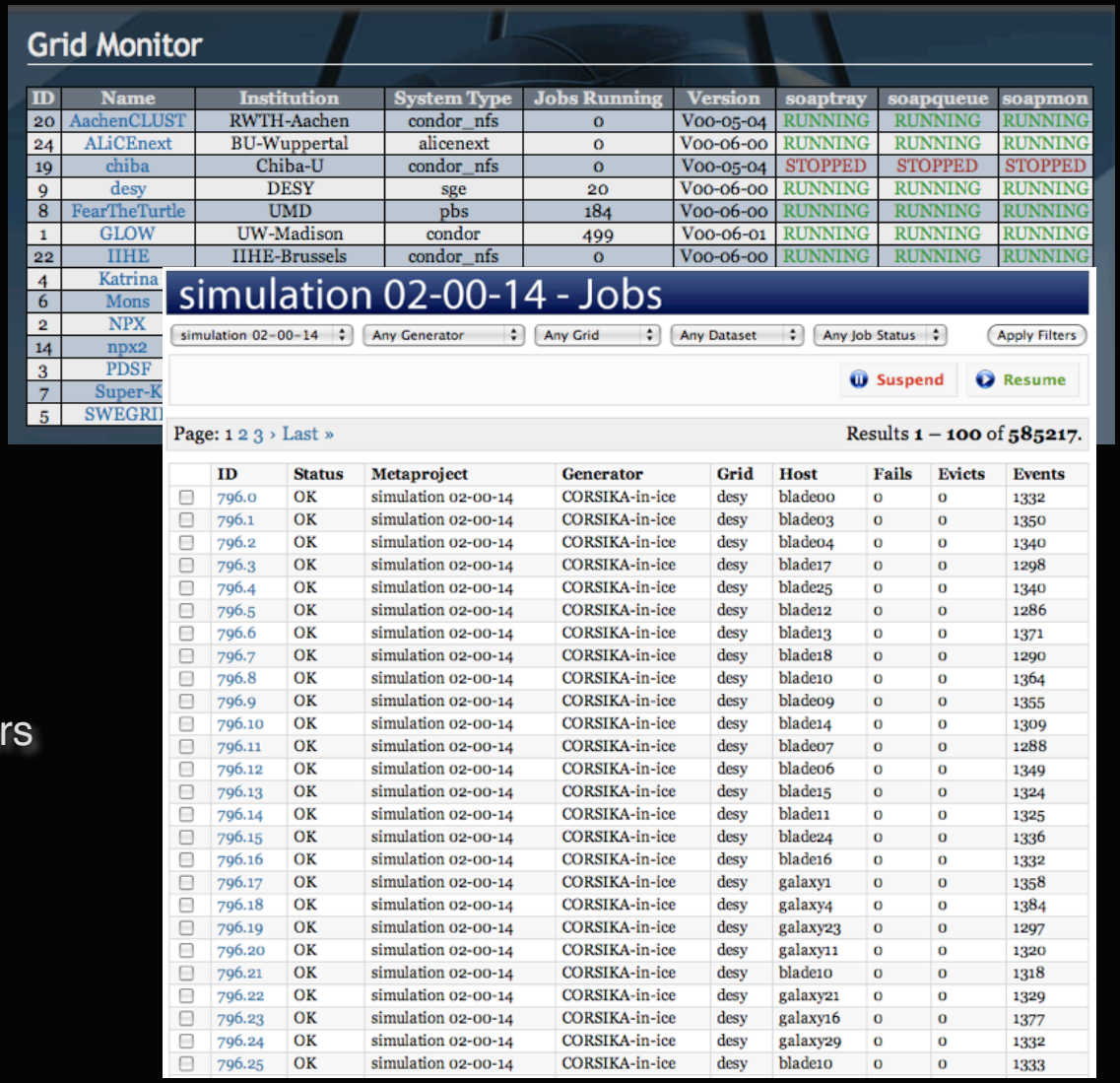


iceprod plugins

- adaptable to different sites and batch systems
- derived from base class I3Queue
- overloads submission commands for specific batch system

monitoring

- production status & troubleshooting
- remote job management
- unified monitoring for multiple clusters



Grid Monitor

ID	Name	Institution	System Type	Jobs Running	Version	soaptray	soapqueue	soapmon
20	AachenCLUST	RWTH-Aachen	condor_nfs	0	V00-05-04	RUNNING	RUNNING	RUNNING
24	ALiCEnext	BU-Wuppertal	alicenext	0	V00-06-00	RUNNING	RUNNING	RUNNING
19	chiba	Chiba-U	condor_nfs	0	V00-05-04	STOPPED	STOPPED	STOPPED
9	desy	DESY	sge	20	V00-06-00	RUNNING	RUNNING	RUNNING
8	FearTheTurtle	UMD	pbs	184	V00-06-00	RUNNING	RUNNING	RUNNING
1	GLOW	UW-Madison	condor	499	V00-06-01	RUNNING	RUNNING	RUNNING
22	IIHE	IIHE-Brussels	condor_nfs	0	V00-06-00	RUNNING	RUNNING	RUNNING
4	Katrina							
6	Mons							
2	NPX							
14	npx2							
3	PDSF							
7	Super-K							
5	SWEGRI							

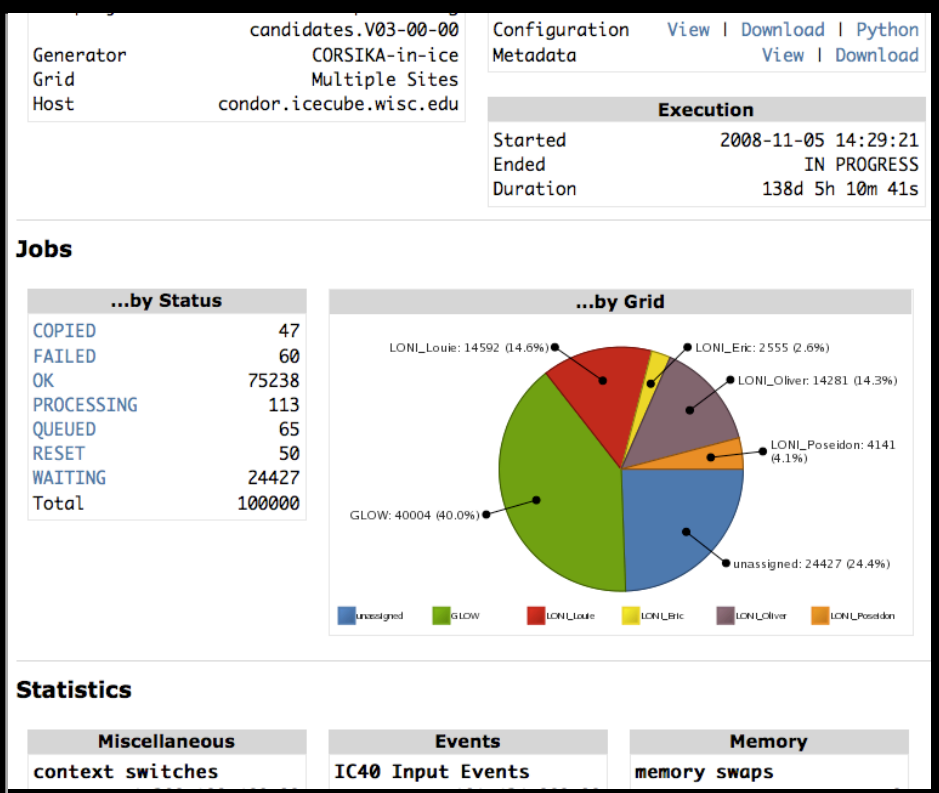
simulation 02-00-14 - Jobs

simulation 02-00-14 | Any Generator | Any Grid | Any Dataset | Any Job Status | Apply Filters

Suspend Resume

Page: 1 2 3 › Last » Results 1 – 100 of 585217.

	ID	Status	Metaproject	Generator	Grid	Host	Fails	Evicts	Events
☐	796.0	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade00	0	0	1332
☐	796.1	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade03	0	0	1350
☐	796.2	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade04	0	0	1340
☐	796.3	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade17	0	0	1298
☐	796.4	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade25	0	0	1340
☐	796.5	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade12	0	0	1286
☐	796.6	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade13	0	0	1371
☐	796.7	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade18	0	0	1290
☐	796.8	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade10	0	0	1364
☐	796.9	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade09	0	0	1355
☐	796.10	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade14	0	0	1309
☐	796.11	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade07	0	0	1288
☐	796.12	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade06	0	0	1349
☐	796.13	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade15	0	0	1324
☐	796.14	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade11	0	0	1325
☐	796.15	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade24	0	0	1336
☐	796.16	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade16	0	0	1332
☐	796.17	OK	simulation 02-00-14	CORSIKA-in-ice	desy	galaxy1	0	0	1358
☐	796.18	OK	simulation 02-00-14	CORSIKA-in-ice	desy	galaxy4	0	0	1384
☐	796.19	OK	simulation 02-00-14	CORSIKA-in-ice	desy	galaxy23	0	0	1297
☐	796.20	OK	simulation 02-00-14	CORSIKA-in-ice	desy	galaxy11	0	0	1320
☐	796.21	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade10	0	0	1318
☐	796.22	OK	simulation 02-00-14	CORSIKA-in-ice	desy	galaxy21	0	0	1329
☐	796.23	OK	simulation 02-00-14	CORSIKA-in-ice	desy	galaxy16	0	0	1377
☐	796.24	OK	simulation 02-00-14	CORSIKA-in-ice	desy	galaxy29	0	0	1332
☐	796.25	OK	simulation 02-00-14	CORSIKA-in-ice	desy	blade10	0	0	1333



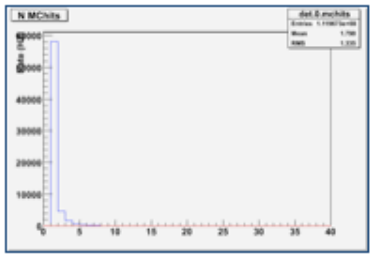
IceProd : web interface

production monitoring and statistics collection

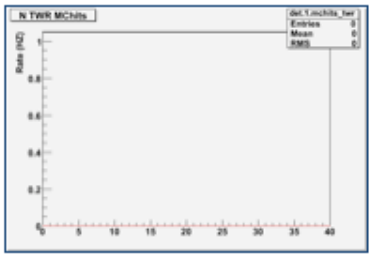
production history and configurations search engine

IceProd : production verification

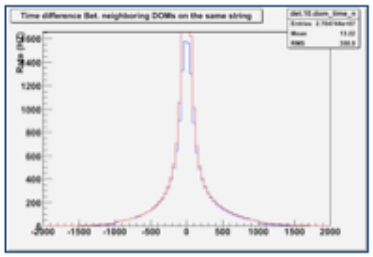
Detector Histograms



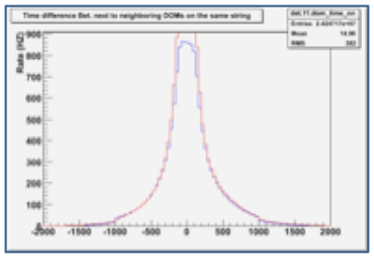
det.0.mchits



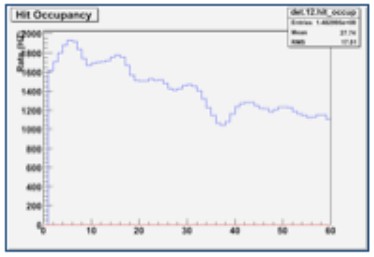
det.1.mchits_twr



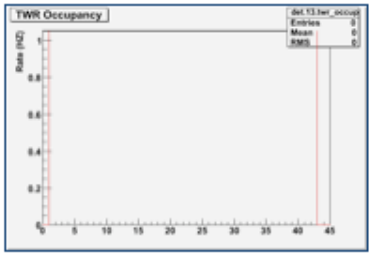
det.10.dom_time_n



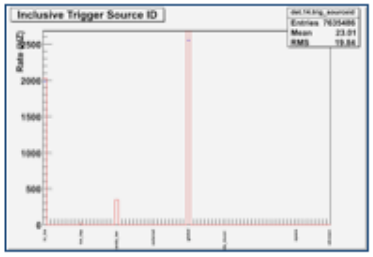
det.11.dom_time_nn



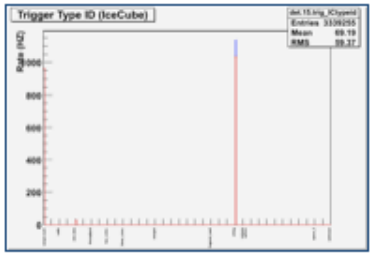
det.12.hit_occup



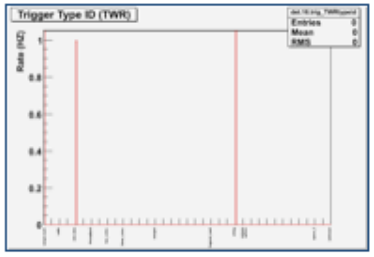
det.13.twr_occup



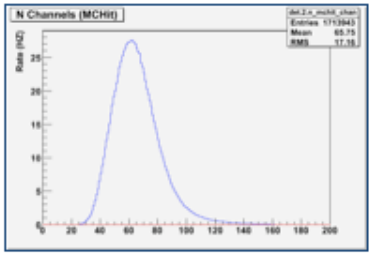
det.14.trig_sourceid



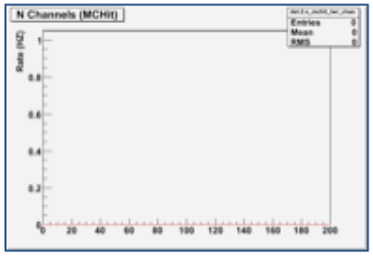
det.15.trig_ICtypeid



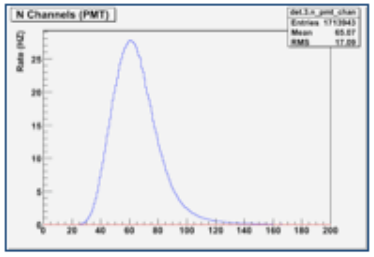
det.16.trig_TWRtypeid



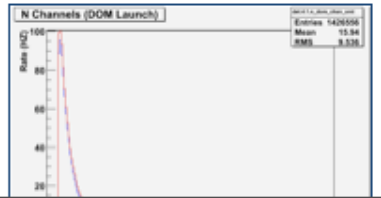
det.2.n_mchit_chan

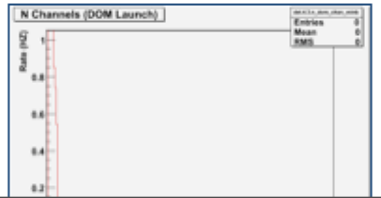


det.2.n_mchit_twr_chan



det.3.n_pmt_chan





syschck

- System checks added in order to check site readiness for production (IC59).
- External modules.i3 module (like sim and filter scripts)
- Makes use of existing job statistics to report status of system dependency checks.
- Additional checks are needed.

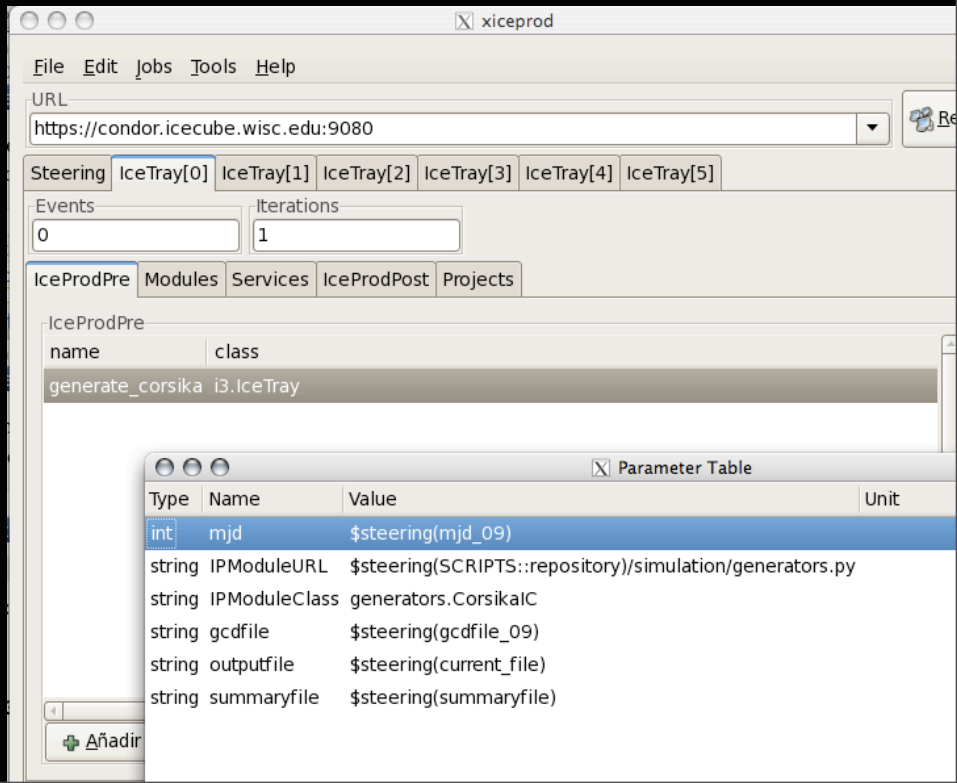
SYSCHK

Most Recent System Chec

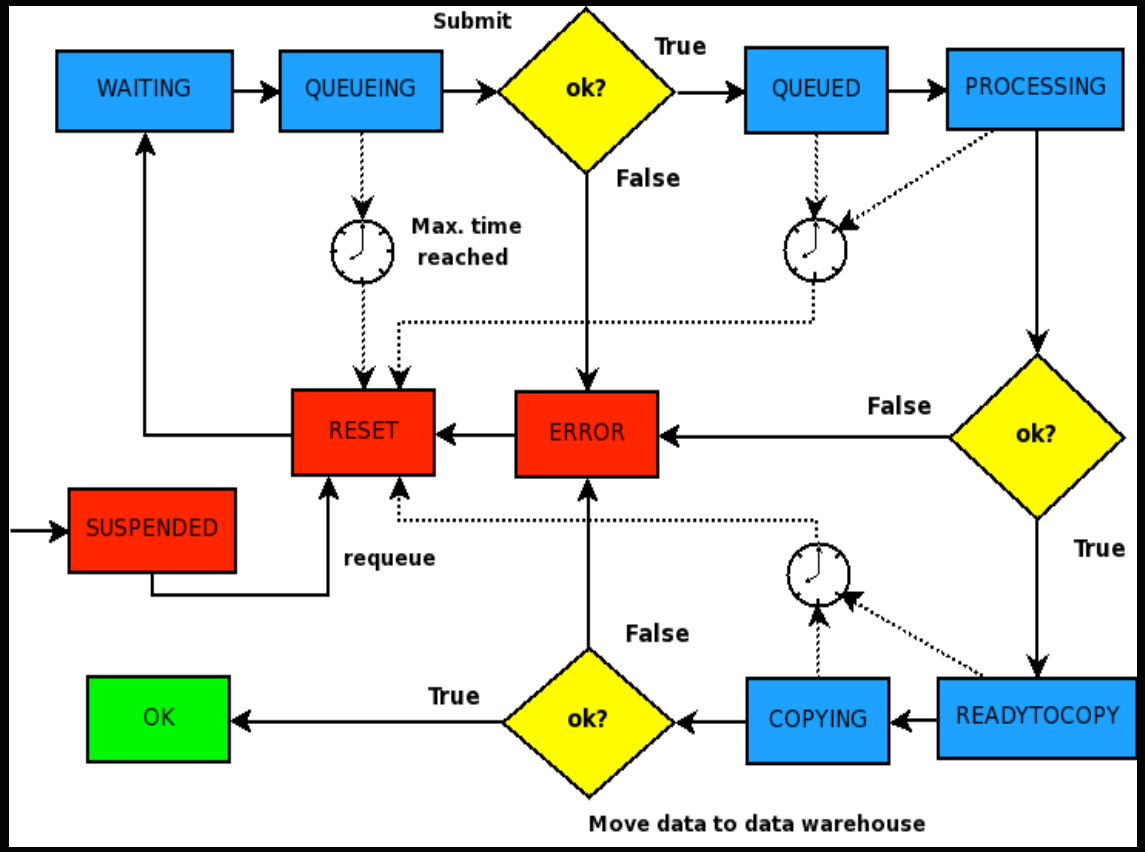
Miscellaneous	
CPUSpeed(GHz)	
sum	23.520
average	2.352
CPUSpeed>=2GHz	
sum	10.000
average	1.000
gccVerion>=3.4.6	
sum	10.000
average	1.000
HasJAVA_HOME	
sum	10.000
average	1.000
HasPhotonTables	
sum	10.000
average	1.000
HasPhotoRec	
sum	10.000
average	1.000
HasPhotoRecCscd	
sum	10.000
average	1.000
HasPhotoRecMu	
sum	10.000
average	1.000
i3exec runtime	
sum	470.033
average	47.003
JAVA_Version>=1.4.2	
sum	10.000
average	1.000

IceProd Client

- Graphical User Interface
- CMD line interface
- shell interface
- editor of XML IceProd/IceTray steering files
- XMLRPC interface to daemons
- job submission/control



job management



Usage of Grid Resources

1. In the last year, we have incorporated EGEE sites from Germany and Belgium into the IceCube grid.

2.We have begun preliminary work on OSG and are currently working to optimize use of grid resources.

3. New IceCube VO for use in both EGEE and OSG.

4. We have setup GridFTP infrastructure for data transfers.

Usage of Grid Resources

Grid	CPU years	nodes
EGEE (DE)	24.28	1389
OSG	0.003	213
GLOW	102.8	803
Other clusters	148.3	3275
Total	275.4	4886

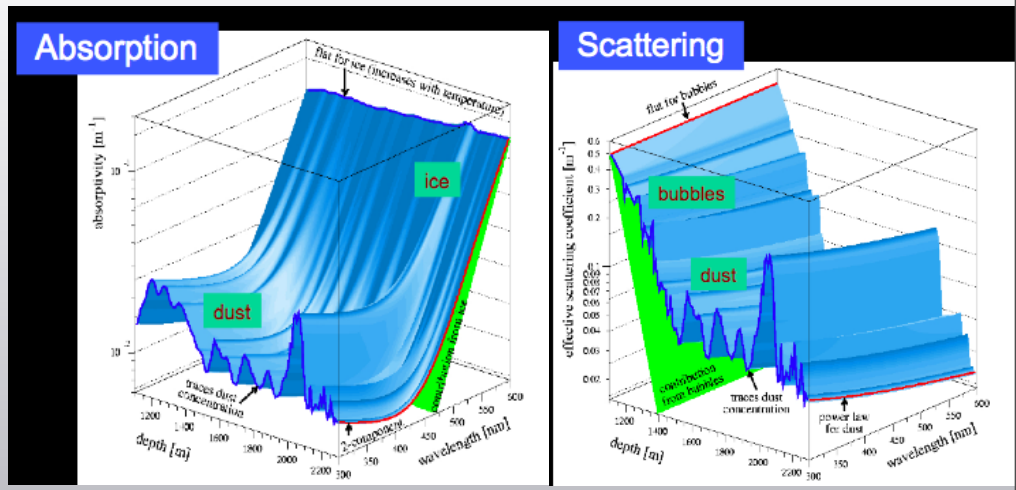
Only testing at this point

EGEE vs. OSG

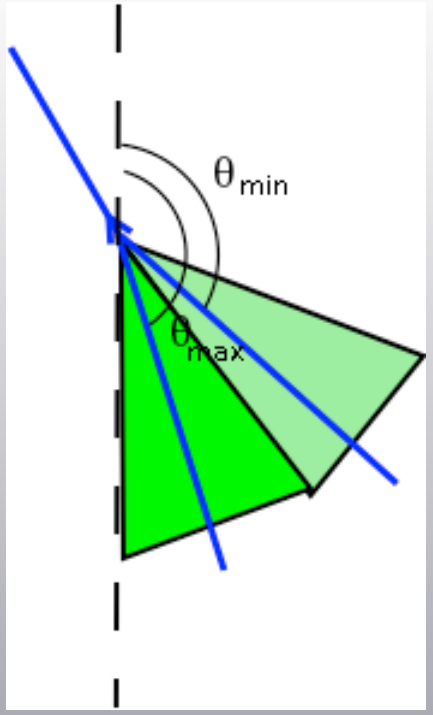
- Top-down design.
- Fairly homogeneous
- Common set of tools
- dCache VO storage

- Cooperative consortium
- Heterogeneous
- Varying policies
- No dedicated storage

Grid challenges: Photonics



- photon interaction probability tables are produces with detailed module of ice properties
- full set of tables is >14 GB (too large to load in memory on most nodes)
- we sort events in zenith bins and process process each bin separately.
- most of our current production clusters have tables pre-installed on nodes
- This limits our ability to add new clusters or large grids for simulation production.

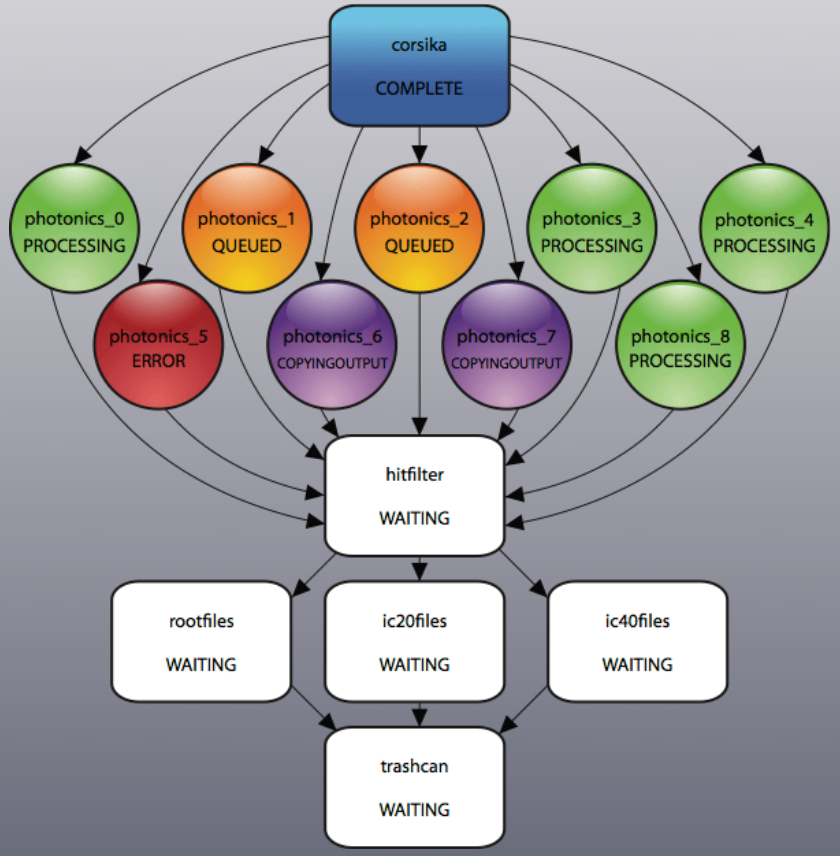


Event Sorting

- Sort events into separate files according to arrival direction zenith angle θ.
- Photonics are binned according θ
- Only need to load relevant tables for given angular bin.

The DAG

- Use Condor DAGMan to divide a simulation job into multiple Condor jobs
- Each Condor job is called a "task" that runs part of a simulation job
- Status updates are tracked for each task in the DB and displayed on the web



Benefits

- Parallel parts can execute simultaneously
- Photonics tables are only needed for part of the execution
- Tasks save their work in progress, so jobs can resume work in the middle if needed (coarse checkpointing)

Disadvantages

- Network overhead
  - Tasks fetch input from previous tasks and store their output over the network
  - Approximately 3GB of network traffic for one simulation job
  - DC will add a dedicated “scratch” GridFTP server
- Queueing overhead
  - Each task is queued separately

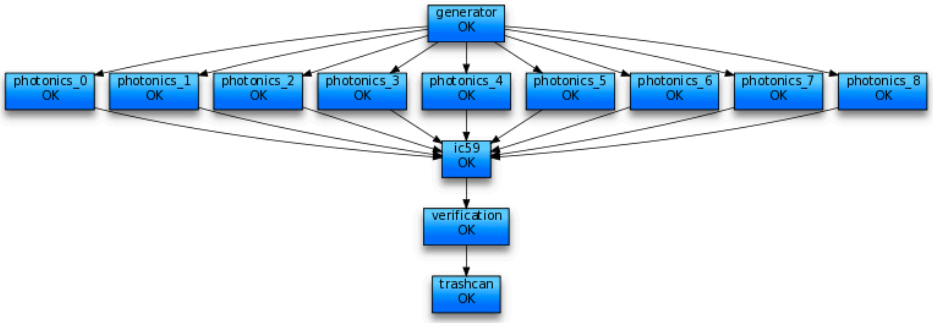
Condor DAG

- separate simulation segments into tasks
- assign task to a node in DAG
- let Condor manage the DAG dependencies
- copy only partial set of tables with each job
-

Tasks

Name	Tray	Iter	Host	Status	Start	Finish
generator	0	0	cabinet-7-7-31.t2.ucsd.edu	OK	2010-01-08 11:39:03	2010-01-08 14:23:33
photonics	1	0	g10n05.hep.wisc.edu	OK	2010-01-08 17:51:53	2010-01-08 18:14:25
photonics	1	1	g16n33.hep.wisc.edu	OK	2010-01-08 17:51:54	2010-01-08 18:13:44
photonics	1	2	g16n25.hep.wisc.edu	OK	2010-01-08 17:51:27	2010-01-08 18:18:56
photonics	1	3	g16n05.hep.wisc.edu	OK	2010-01-08 17:51:34	2010-01-08 18:18:46
photonics	1	4	g16n36.hep.wisc.edu	OK	2010-01-08 17:52:00	2010-01-08 18:21:07
photonics	1	5	cabinet-7-7-20.t2.ucsd.edu	OK	2010-01-08 17:51:59	2010-01-08 18:34:37
photonics	1	6	g12n22.hep.wisc.edu	OK	2010-01-08 17:52:06	2010-01-08 18:08:11
photonics	1	7	g12n31.hep.wisc.edu	OK	2010-01-08 17:52:06	2010-01-08 18:03:37
photonics	1	8	g12n08.hep.wisc.edu	OK	2010-01-08 17:56:22	2010-01-08 18:09:10
ic59	2	0	g14n23.hep.wisc.edu	OK	2010-01-08 19:01:43	2010-01-08 19:17:36
ic59	3	0	g14n23.hep.wisc.edu	OK	2010-01-08 19:17:36	2010-01-08 19:36:15
verification	4	0	cabinet-4-4-25.t2.ucsd.edu	OK	2010-01-08 21:22:12	2010-01-08 21:34:24
trashcan	0	0	cabinet-6-6-28.t2.ucsd.edu	OK	2010-01-08 21:40:42	2010-01-08 21:41:35

Task Graph



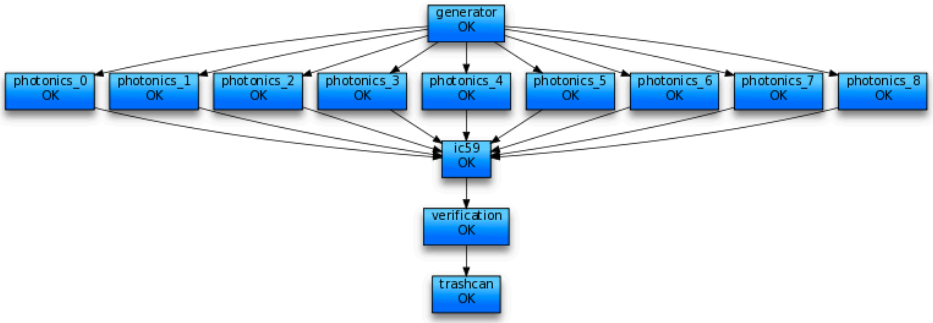
Condor DAG

- separate simulation segments into tasks
- assign task to a node in DAG
- let Condor manage the DAG dependencies
- copy only partial set of tables with each job
- **use SQUID proxy to cache files**

Tasks

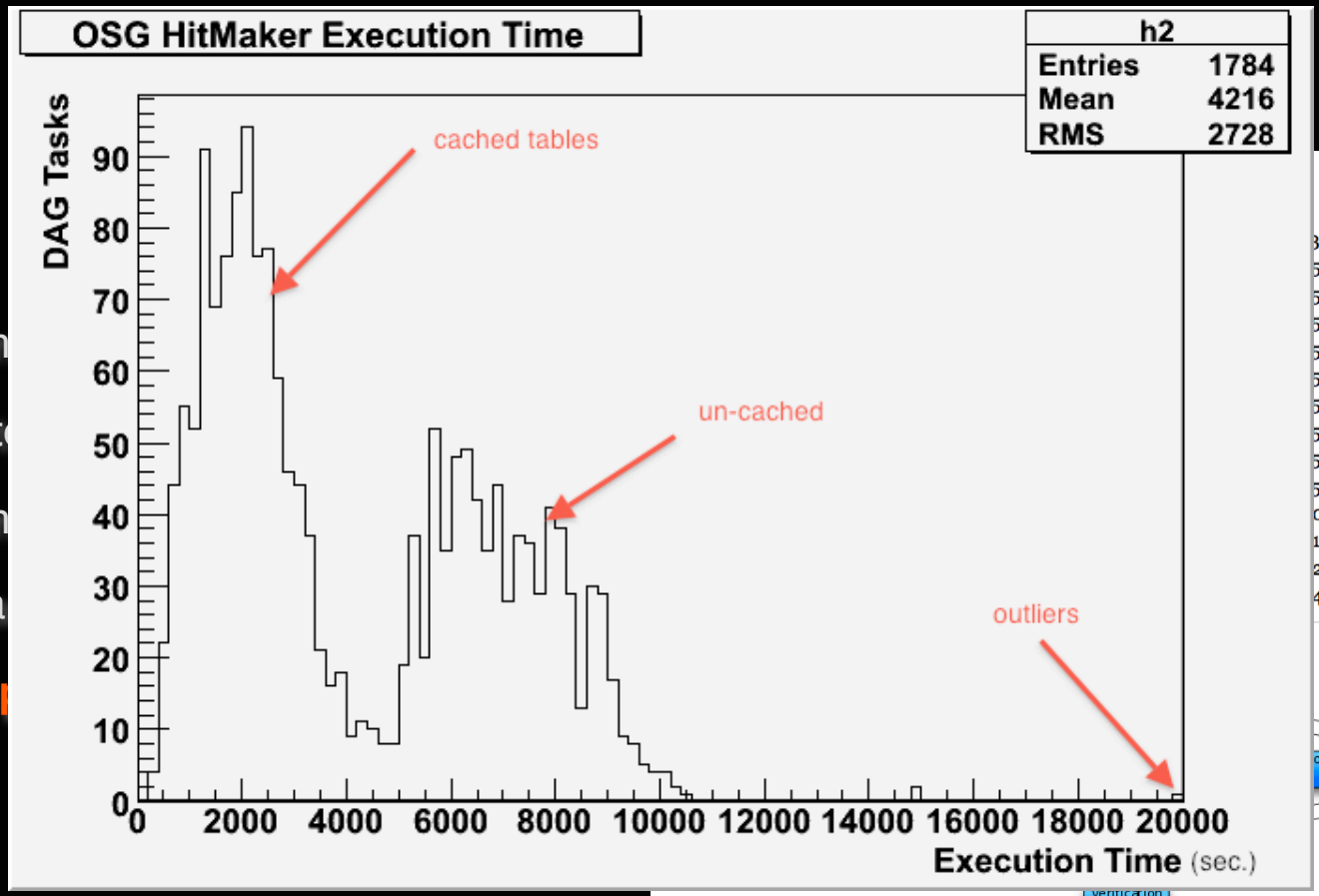
Name	Tray	Iter	Host	Status	Start	Finish
generator	0	0	cabinet-7-7-31.t2.ucsd.edu	OK	2010-01-08 11:39:03	2010-01-08 14:23:33
photonics	1	0	g10n05.hep.wisc.edu	OK	2010-01-08 17:51:53	2010-01-08 18:14:25
photonics	1	1	g16n33.hep.wisc.edu	OK	2010-01-08 17:51:54	2010-01-08 18:13:44
photonics	1	2	g16n25.hep.wisc.edu	OK	2010-01-08 17:51:27	2010-01-08 18:18:56
photonics	1	3	g16n05.hep.wisc.edu	OK	2010-01-08 17:51:34	2010-01-08 18:18:46
photonics	1	4	g16n36.hep.wisc.edu	OK	2010-01-08 17:52:00	2010-01-08 18:21:07
photonics	1	5	cabinet-7-7-20.t2.ucsd.edu	OK	2010-01-08 17:51:59	2010-01-08 18:34:37
photonics	1	6	g12n22.hep.wisc.edu	OK	2010-01-08 17:52:06	2010-01-08 18:08:11
photonics	1	7	g12n31.hep.wisc.edu	OK	2010-01-08 17:52:06	2010-01-08 18:03:37
photonics	1	8	g12n08.hep.wisc.edu	OK	2010-01-08 17:56:22	2010-01-08 18:09:10
ic59	2	0	g14n23.hep.wisc.edu	OK	2010-01-08 19:01:43	2010-01-08 19:17:36
ic59	3	0	g14n23.hep.wisc.edu	OK	2010-01-08 19:17:36	2010-01-08 19:36:15
verification	4	0	cabinet-4-4-25.t2.ucsd.edu	OK	2010-01-08 21:22:12	2010-01-08 21:34:24
trashcan	0	0	cabinet-6-6-28.t2.ucsd.edu	OK	2010-01-08 21:40:42	2010-01-08 21:41:35

Task Graph



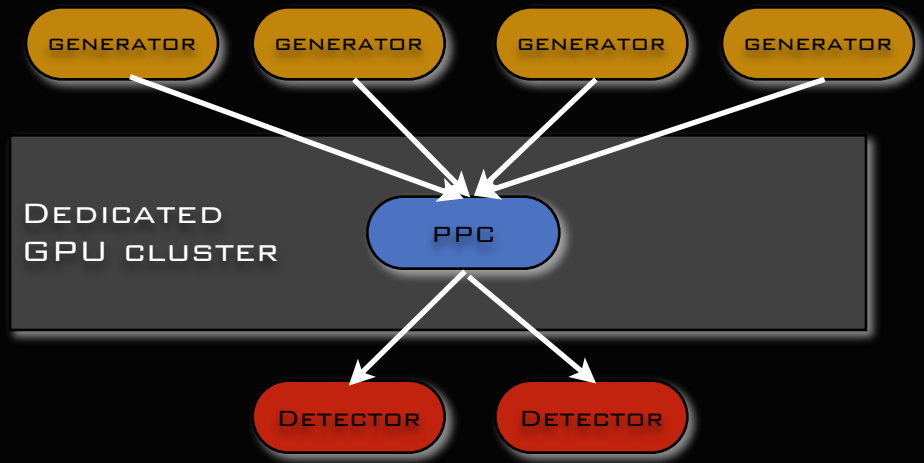
Condor DAG

- separate sim
- assign task t
- let Condor m
- copy only pa
- **use SQUID**



GPU-based simulation

- We have recently began experimenting with GPU-based implementation of portions of IceCube simulation.
- GPU-based photon propagation eliminates need for photon tables but requires a different approach to simulation production.
- New IceProd based DAG assigns separate tasks to different sites
- Most importantly: enable execution of photon propagation simulation on dedicated GPU clusters.



Condor Glide-ins

- We will be adding support for Condor Glide-in submission
- This will allow us to join multiple smaller clusters into a larger virtual cluster
- Glide-ins can use classAdds to advertise resources such as:
  - available segments of photonics tables
  - GPUs

Summary

- IceCube construction will complete in the beginning of 2011 with 86 strings.
- Monte Carlo production and data processing requires use of significant computing resources from the various members of IceCube.
- IceProd software is used to manage production across sites.
- One of the major obsticles for porting simulation to the Grid is the requirement for large photon propagation tables
  - DAGs allow us to split the table requirements but have some network overhead but some optimizations are possible.
  - GPU simulation eliminates need for PT tables but also requires DAGs
  - Future plans include incorporation of Condor Glide-ins to IceProd software.