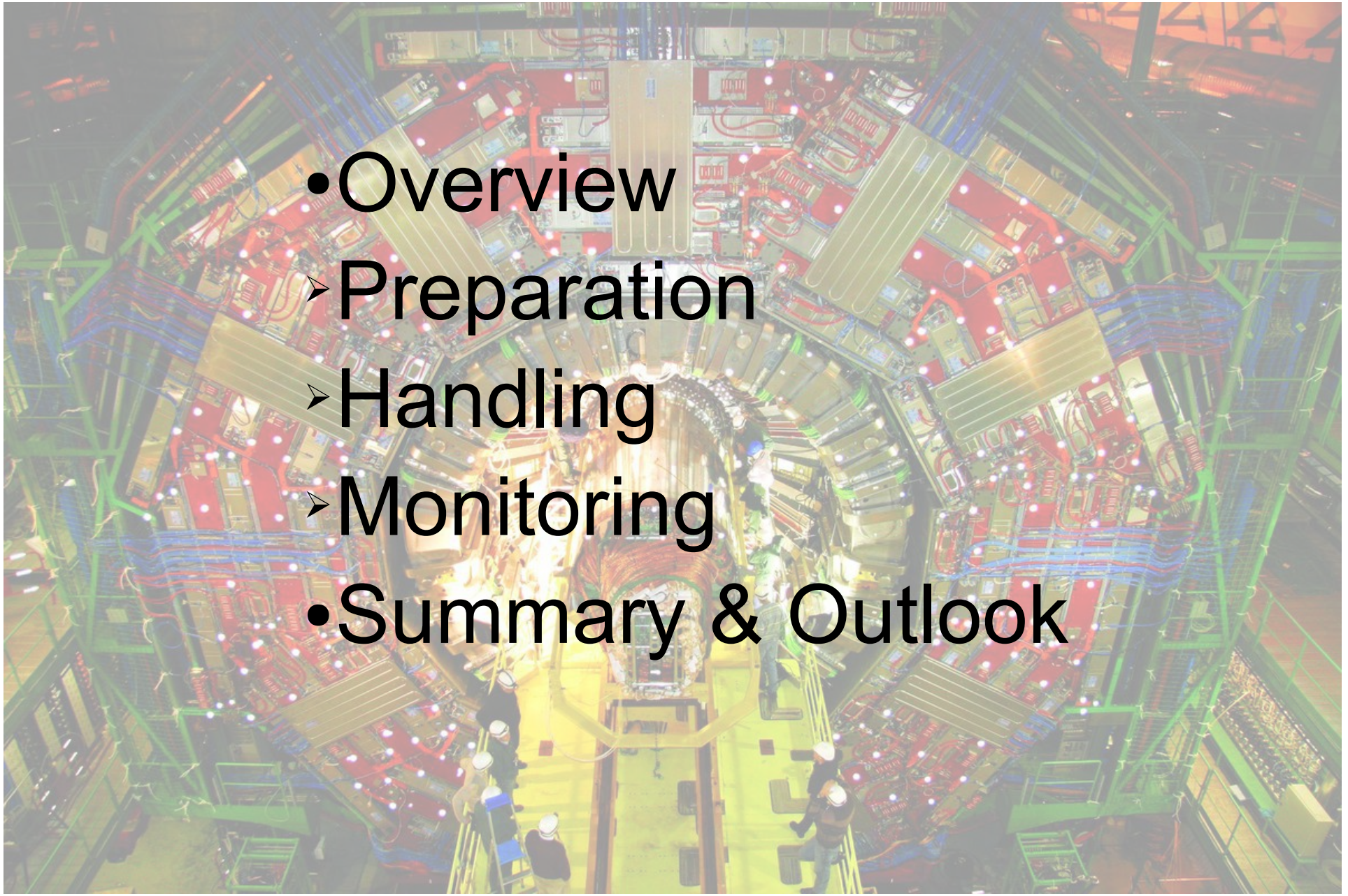
Handling, monitoring, and recent improvements in data processing and Monte Carlo production in the CMS Experiment

ICHEP, August 5, 2016

Jean-Roch Vlimant for the CMS Collaboration

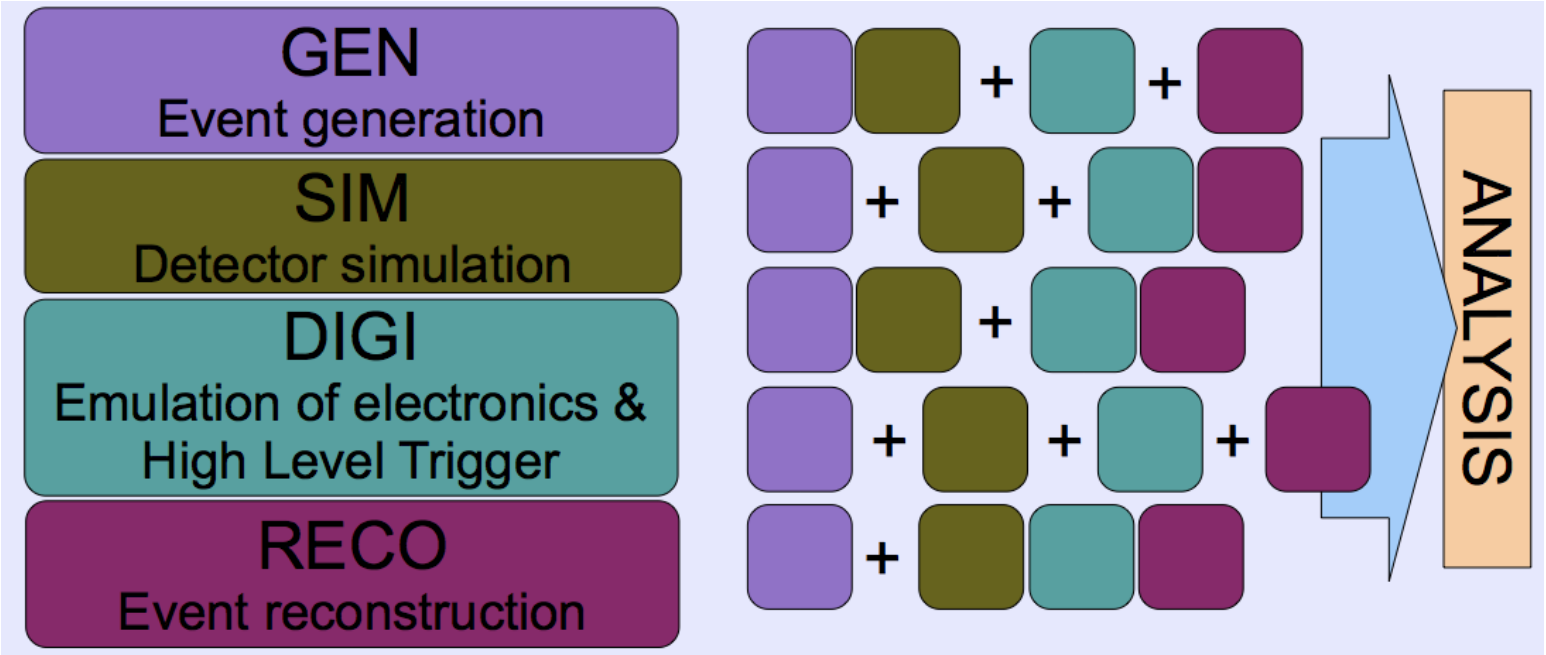
Outline



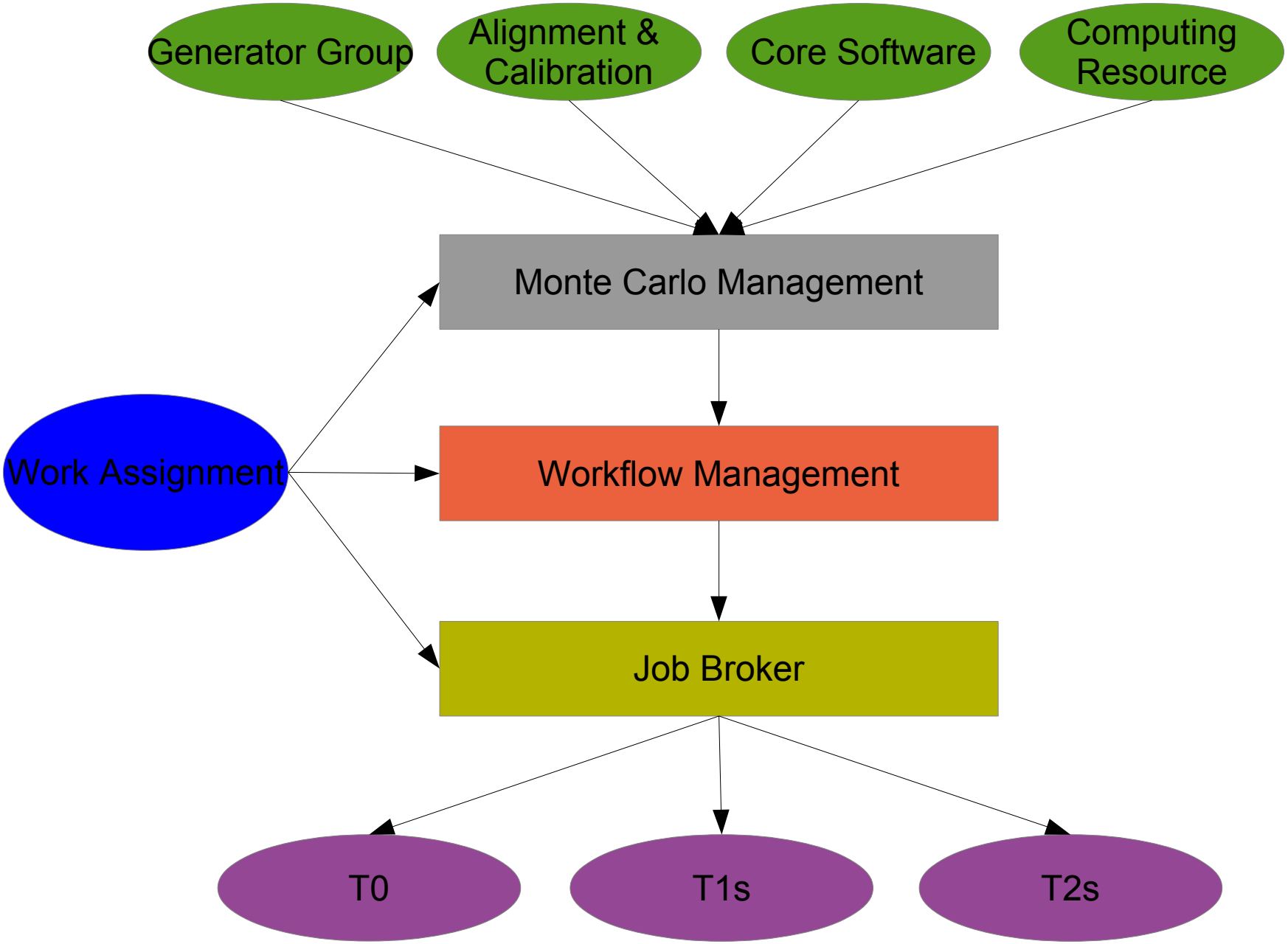
- Overview
- Preparation
- Handling
- Monitoring
- Summary & Outlook

Simulation Overview

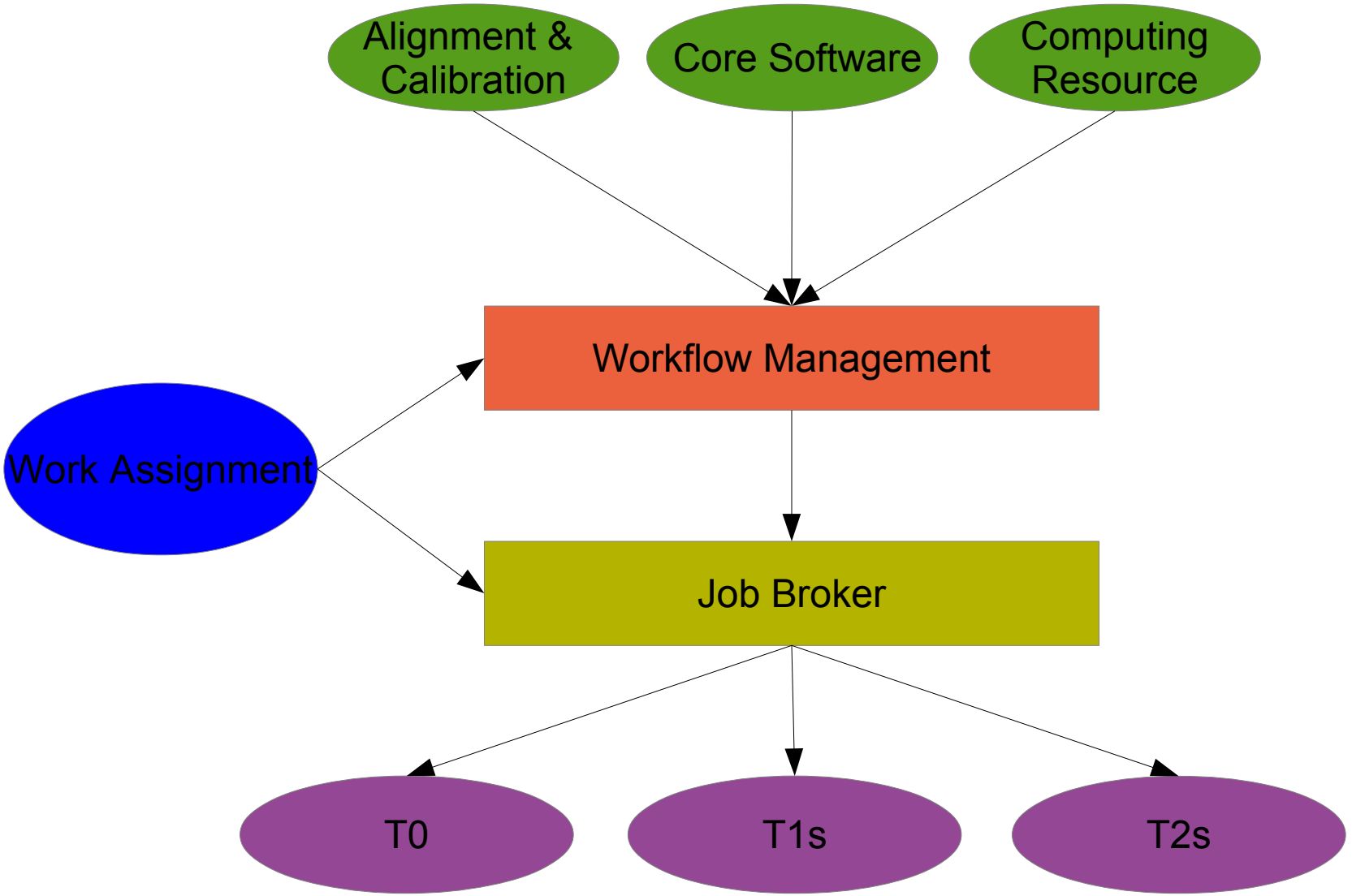
- Analyzing CMS data requires a **large volume of Monte-Carlo**
  - ↙ Billions of events in 10s of thousands of datasets
- Production is done in **successive steps**
  - ↙ Arrangement dictated by software requirements, flexibility, resource utilization, ...
  - → Working towards all-in-one
- Several ways to put a workflow together toward producing an analysis sample
  - → Requires a **flexible and automated production system**



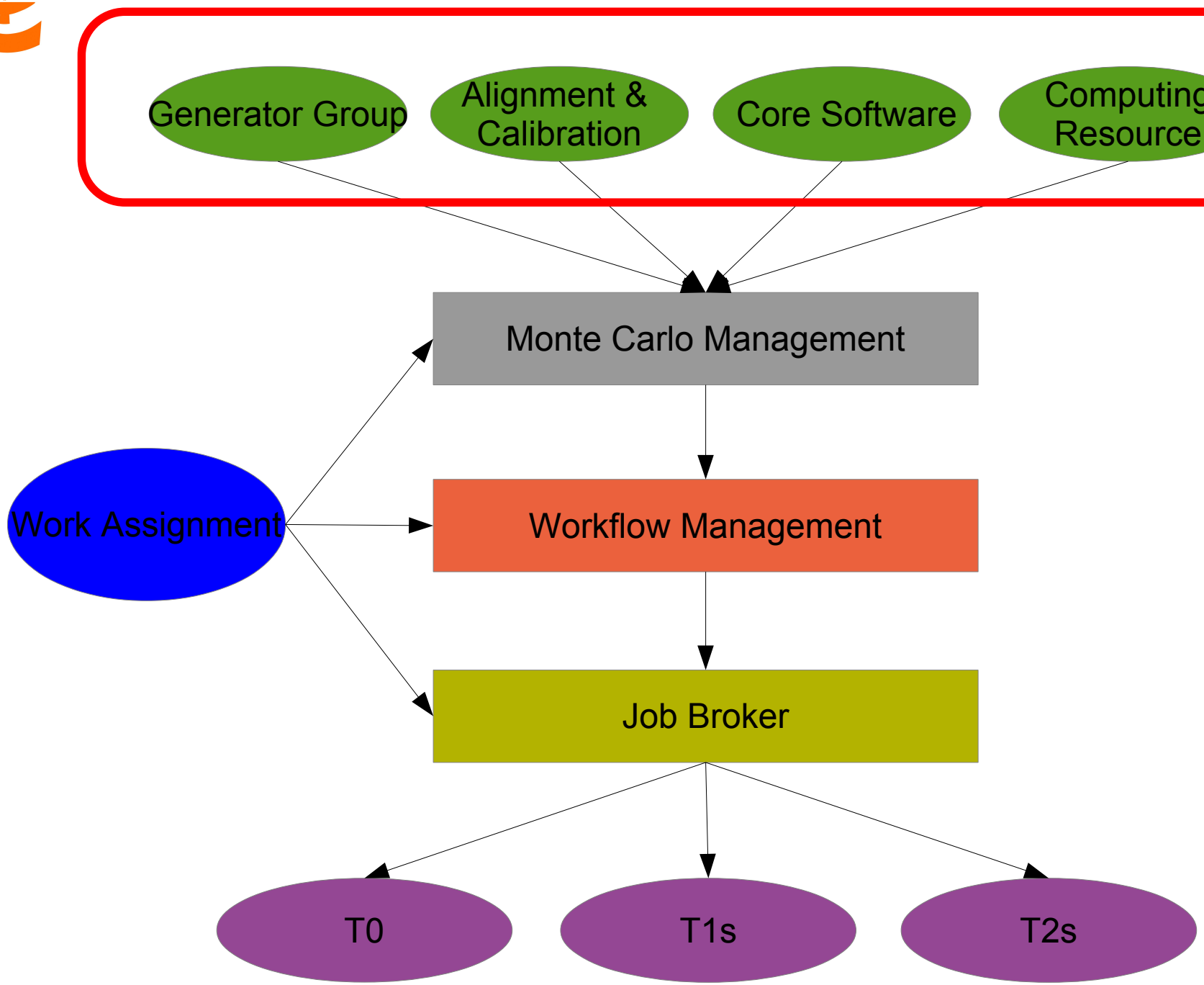
Simulation Flow-Chart



Data Reprocessing Flow-Chart

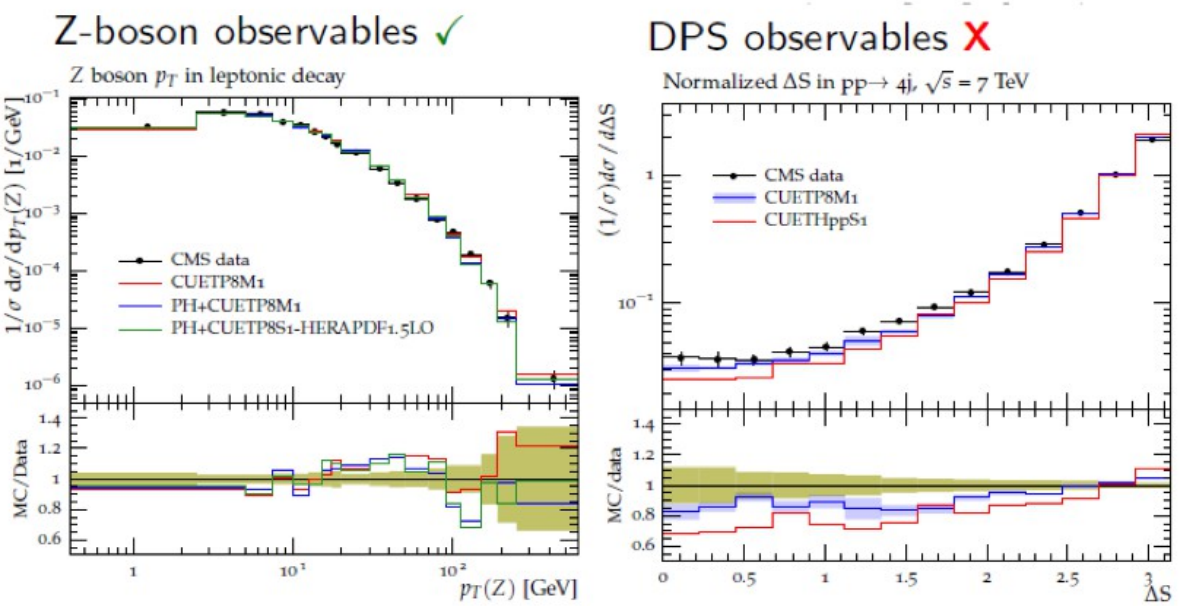



Production Preparation


Generator Group
Alignment & Calibration
Core Software
Computing Resource
Monte Carlo Management
Work Assignment
Workflow Management
Job Broker
T0
T1s
T2s

Generator Work

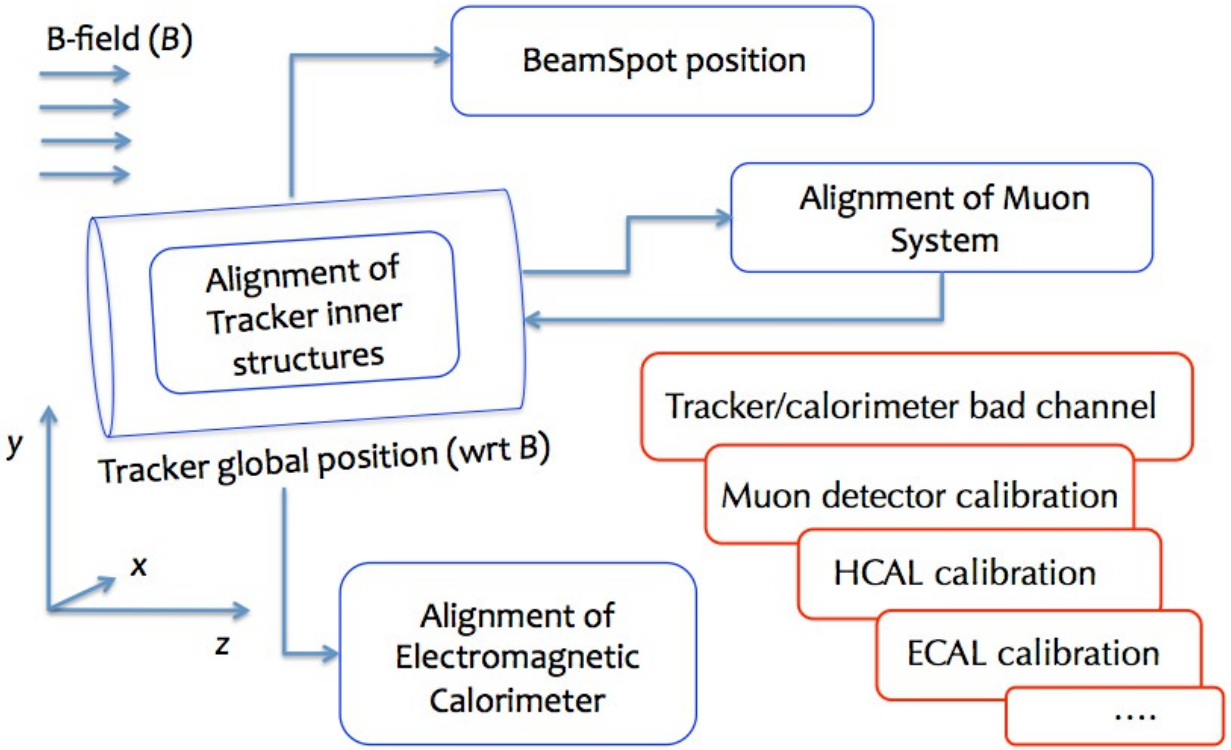
- Interaction/Integration of **external packages to CMS software**
  - ✔ Balance between local patching and main-stream integration
- Normalizing the needs from experiments
  - ✔ http://indico.cern.ch/event/454993/
- Tuning of generator parameters and models
  - ✗ Not always possible to integrate these in production planning
- Plan ahead how to **fulfill Physics deliverable** at conference, considering
  - ➔ Production timescale
  - ➔ Software maturity
  - ➔ Samples overlap
- **Advise** best analyzers on
  - ➔ X-section per samples
  - ➔ Re-weighting procedure
  - ➔ Interlocutors to theory groups



REFERENCE

Alignment & Calibration

- CMS has a **complex mechanical structure** with moveable parts
  - ✔ Needs for careful alignment
- Sub-detector has **performance evolving with time**, beam exposure, ...
  - ✔ Needs for careful calibration
- Data simulation must be **as representative as possible**
  - ✔ Needs for **applying calibration and alignment** scenari
  - ✗ Involves **educated gambling** when preparing for future data taking



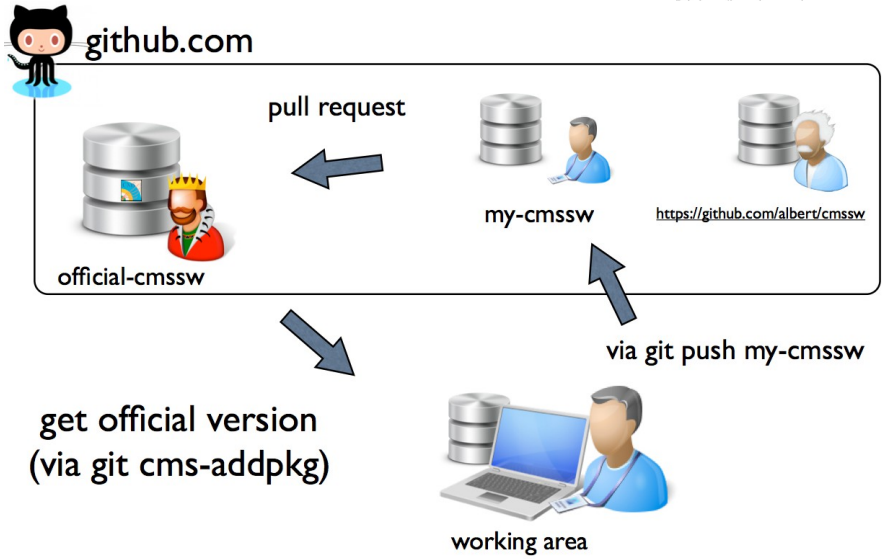
REFERENCE

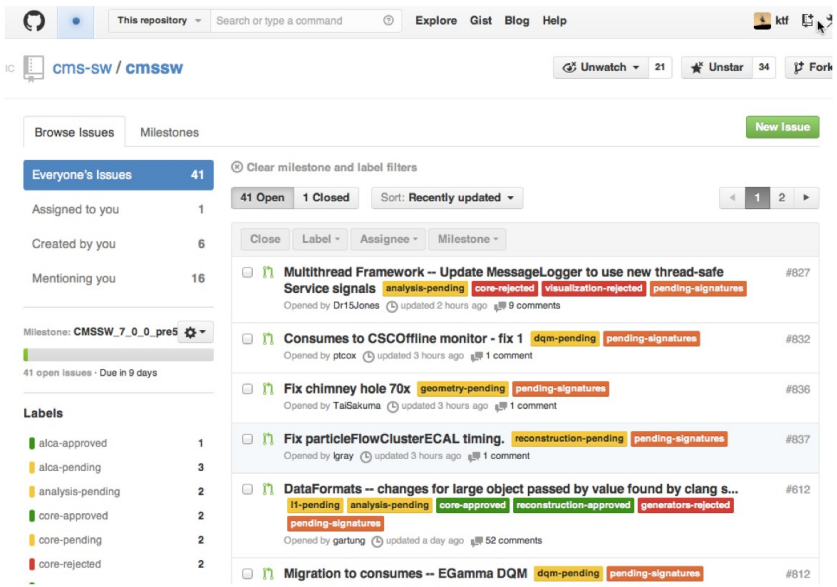
CMS Software

- CMS Software (CMSSW) consists of **millions of lines of code**
  - ✔ Pool of numerous developer
  - ✔ Coordinated in development areas
  - ✔ Centralized in main git repository
  - ✔ Continuous build system using Jenkins

- **Schedule of new features** matching physics delivery plans
- Integration couple with staged **validation procedure**

➔ Needs for planning a couple of months ahead of production start





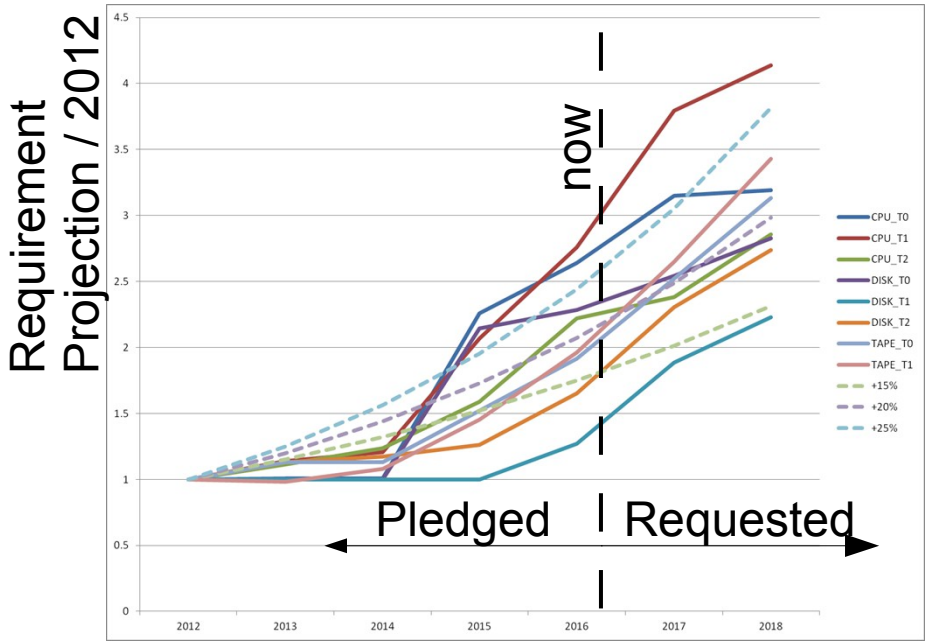
REFERENCE

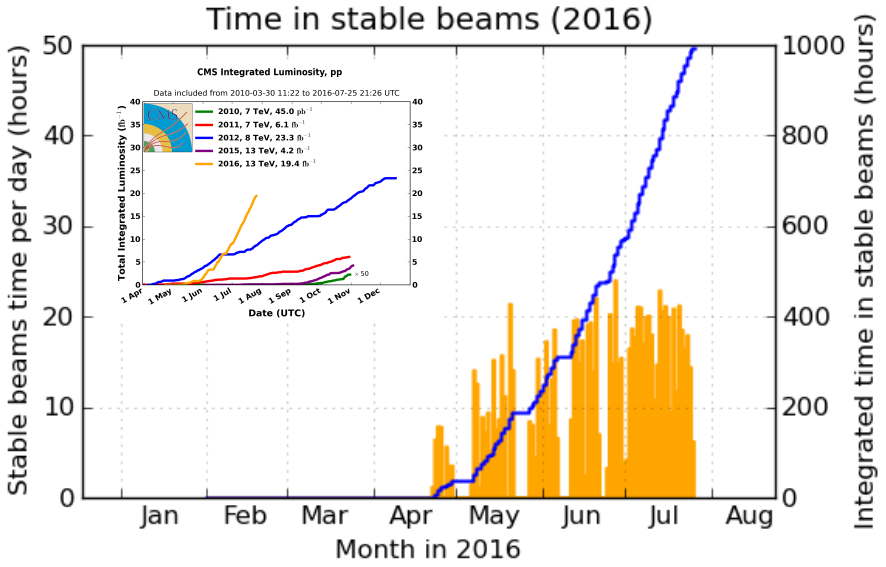
Resource Provisioning

- Grid computing involves large number of compute nodes
  - ✓ Needs refurbishing with **best hardware**
  - ✓ Large budget, and **inertia in availability**
  - → Need to carefully plan usage and place orders ahead of time

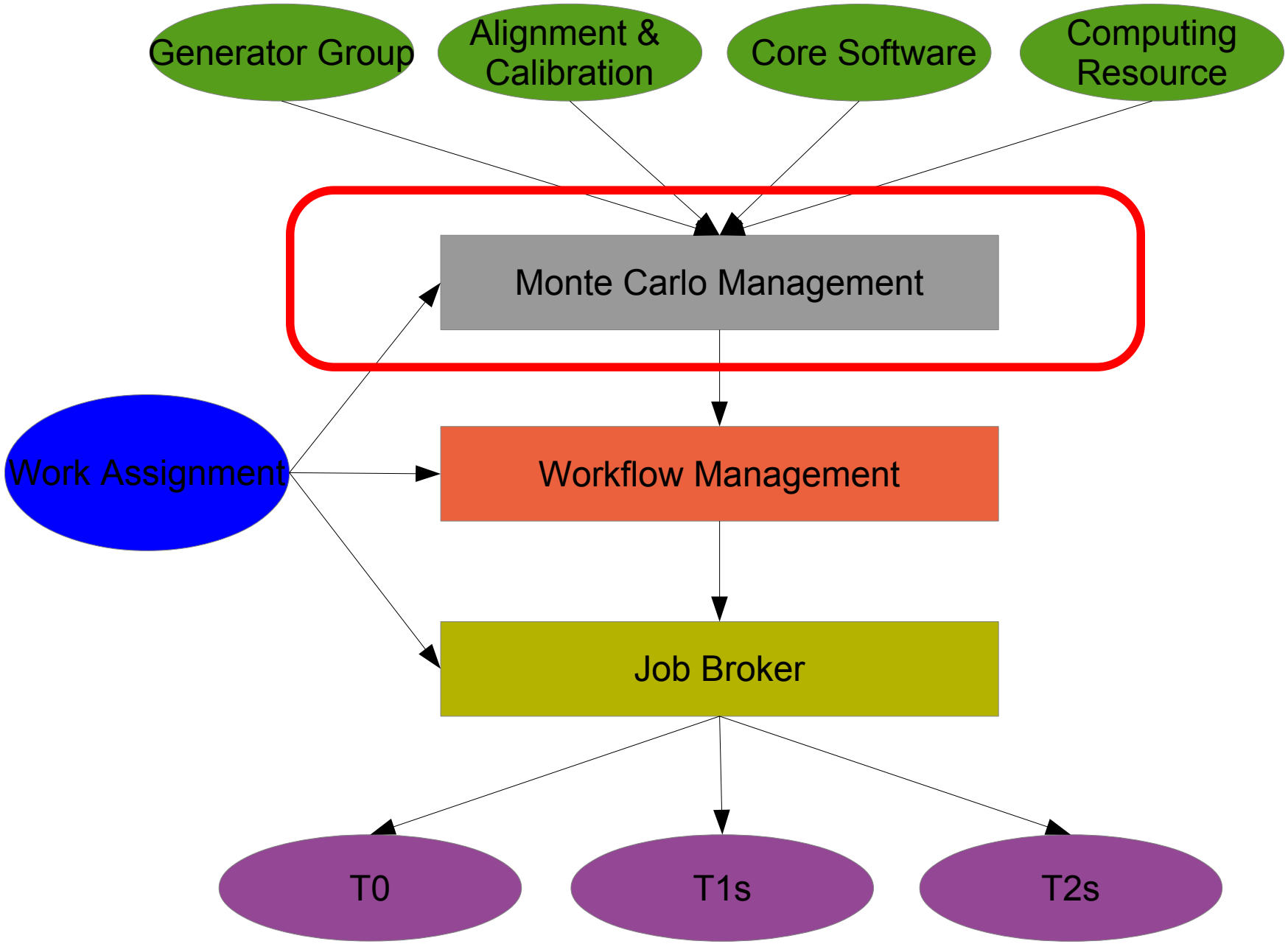
- **Define an operational model** based on software and Physics menu
- Balance between **budget cost and data/MC** volume we can have (including reprocessing)
- Scale the model to **expected accelerator performance** (was done for 1.3k beam hours)
- Fold in a **x2-3 software performance improvement**

- → Organize production and reprocessing mainstream and dedicated within that allocatior
- → Integrating opportunistic resource (AWS, HLT, HPC, ...)





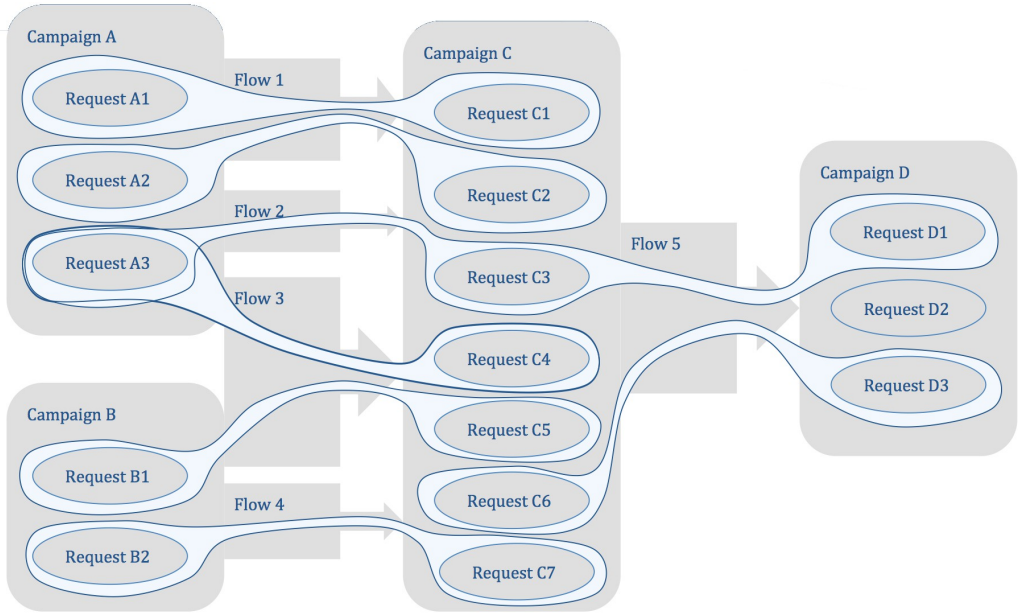
Handling Production


Generator Group
Alignment & Calibration
Core Software
Computing Resource
Monte Carlo Management
Work Assignment
Workflow Management
Job Broker
T0
T1s
T2s

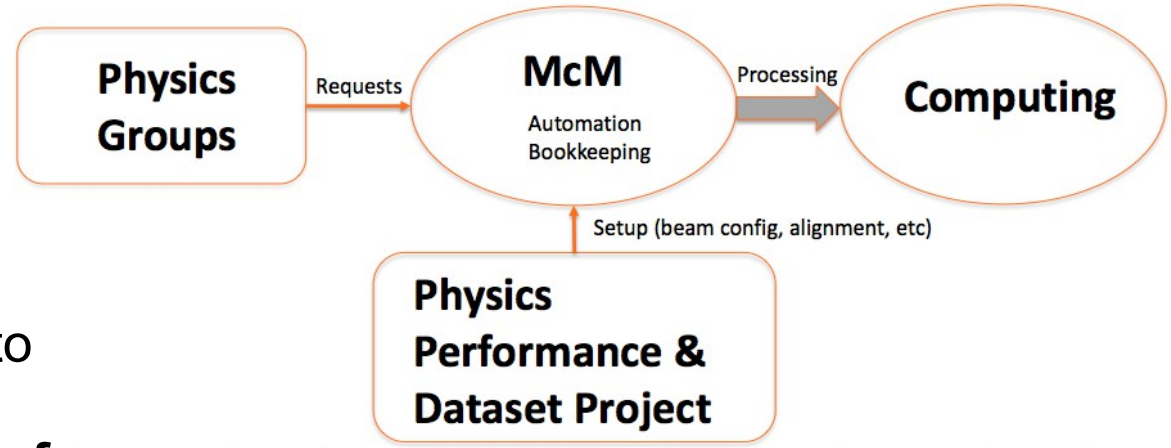
Configuration Assembling

Monte-**C**arlo **M**anagement (McM)

- CMS Software configuration and ingredients for production steps aggregated in **campaigns**
- Subsequent steps of production materialize in **chains of campaigns**
- Flow implement campaign modifiers
- Allow for complex chaining
- **Flexibility** for defining any specific request

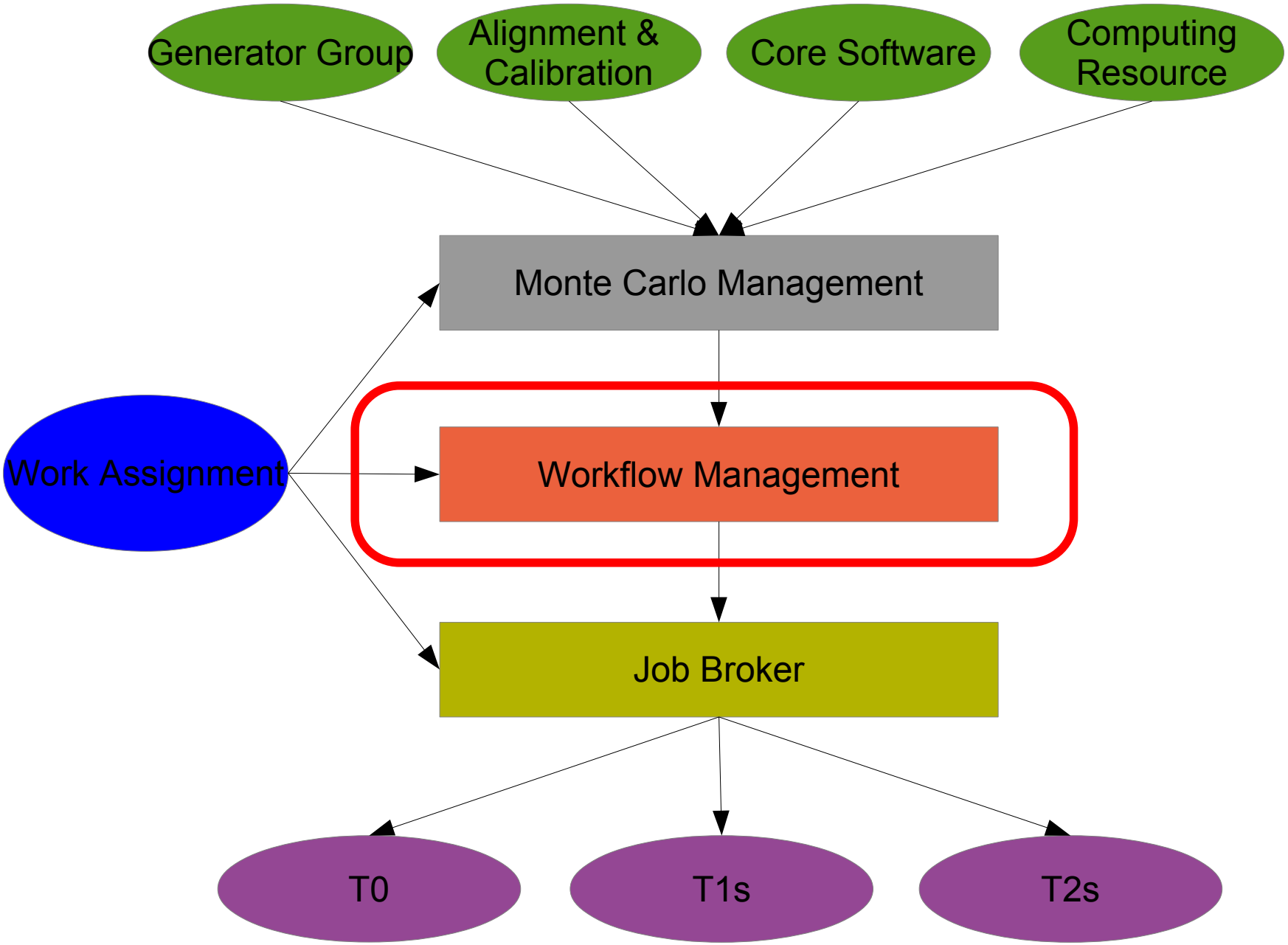


- ✓ Samples requests added by generator contact person
- ✓ Chaining operated by production managers
- ✓ **Automation** where relevant
- ✓ **Validation** histogram provided
- ✓ **Performance run-test** executed



- → Injection of **consolidated workflow** to production system
- → Ability to inject a workflow with **trees of processing steps**

REFERENCE


Generator Group
Alignment & Calibration
Core Software
Computing Resource
Monte Carlo Management
Work Assignment
Workflow Management
Job Broker
T0
T1s
T2s

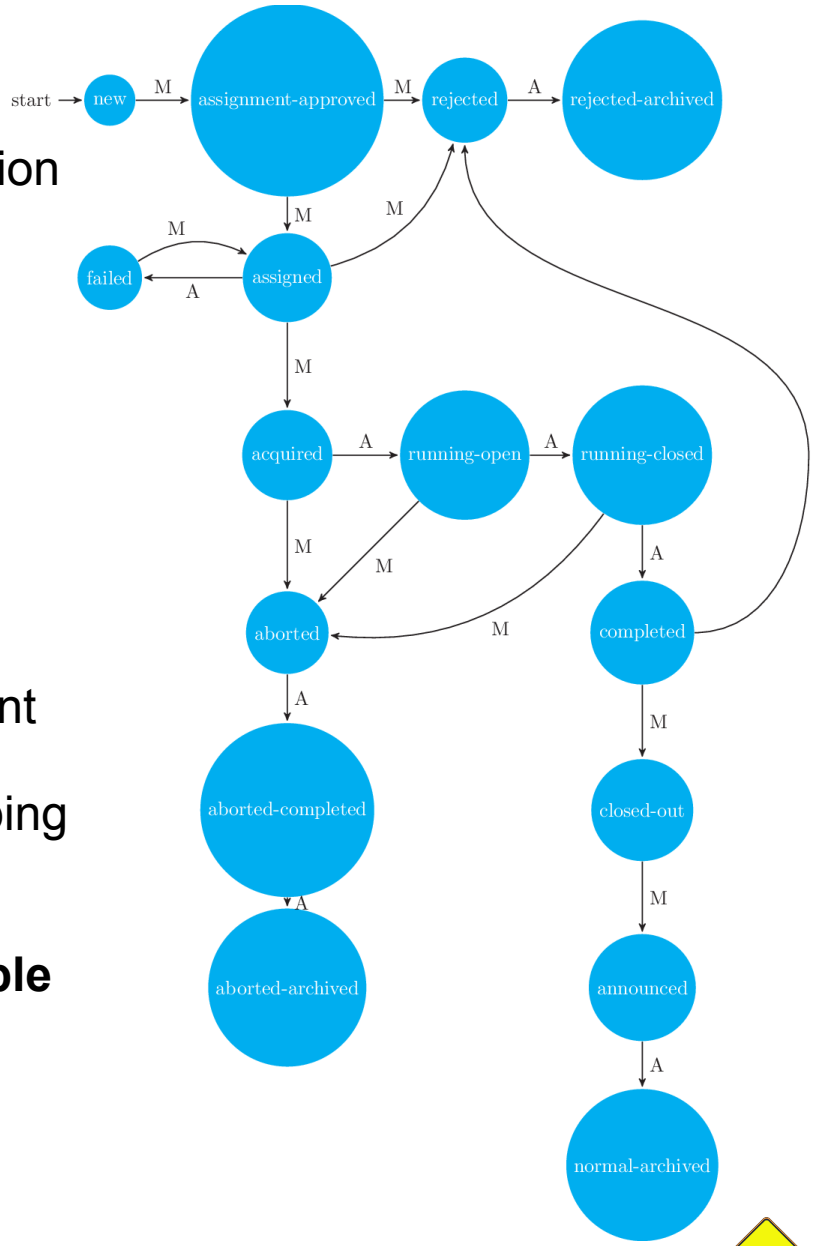
Workflow Management

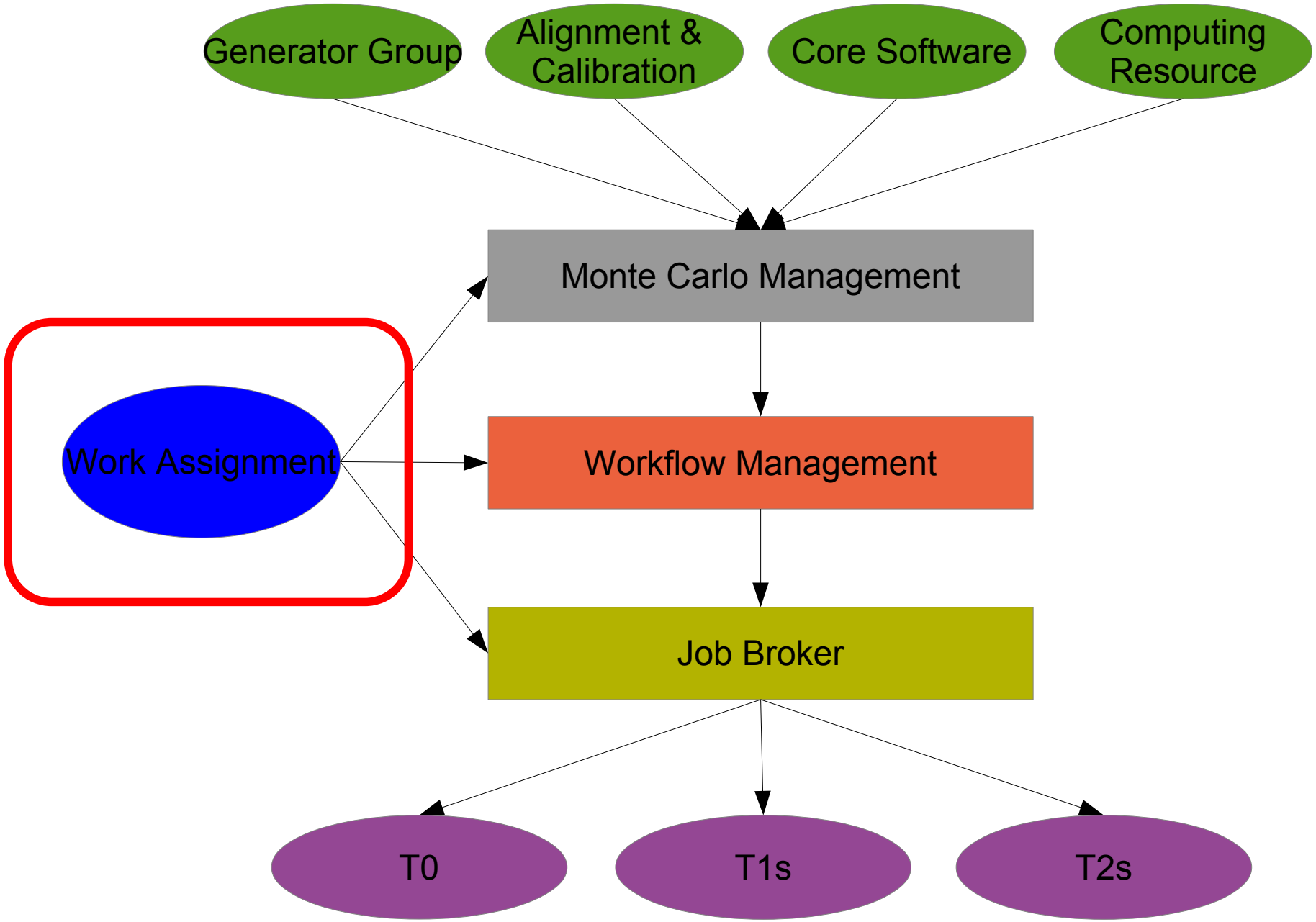
- **Receive assembled configuration**
- Driven by work assignment agent

- Prepare the **full tree of processing** towards the production of the final output
  - ✔ Actual data processing and production
  - ✔ Additional steps: merging small output files for transfer efficiency, cleaning of outputs, collecting of running log files, ...
- **Split jobs** according to workload specifications and data content
- Submit jobs to broker (HTCondor)
- Resubmit certain types of failures
- Keep the books of production data location for subsequent processing
- **Inject the produced data** with parentage into book keeping system

→ System composed **central request manager** and **multiple agents** supporting high load
  - ✔ 5k workflows
  - ✔ 200k jobs pending
  - ✔ 150k jobs running

→ Constant improvement for scalability




Generator Group
Alignment & Calibration
Core Software
Computing Resource
Monte Carlo Management
Work Assignment
Workflow Management
Job Broker
T0
T1s
T2s

Work Assignment

- Central production runs on T0, T1, T2
  - ✔ 50 sites, 200k cores
  - ➢ Extension to opportunistic ressources
- Mostly **homogeneous resource**, but not all sites are equivalent (performance, policy, availability, size, ...)

- Thousands of workload arranged in campaigns have **heterogeneous requirements** (CPU bound, I/O bound, high memory ,...)

➔ **Automation of transfer** in parametrized number of copies of the input data to site
  - ✔ Destinations picked according to CPU pledge
  - ✔ Monitoring of transfers

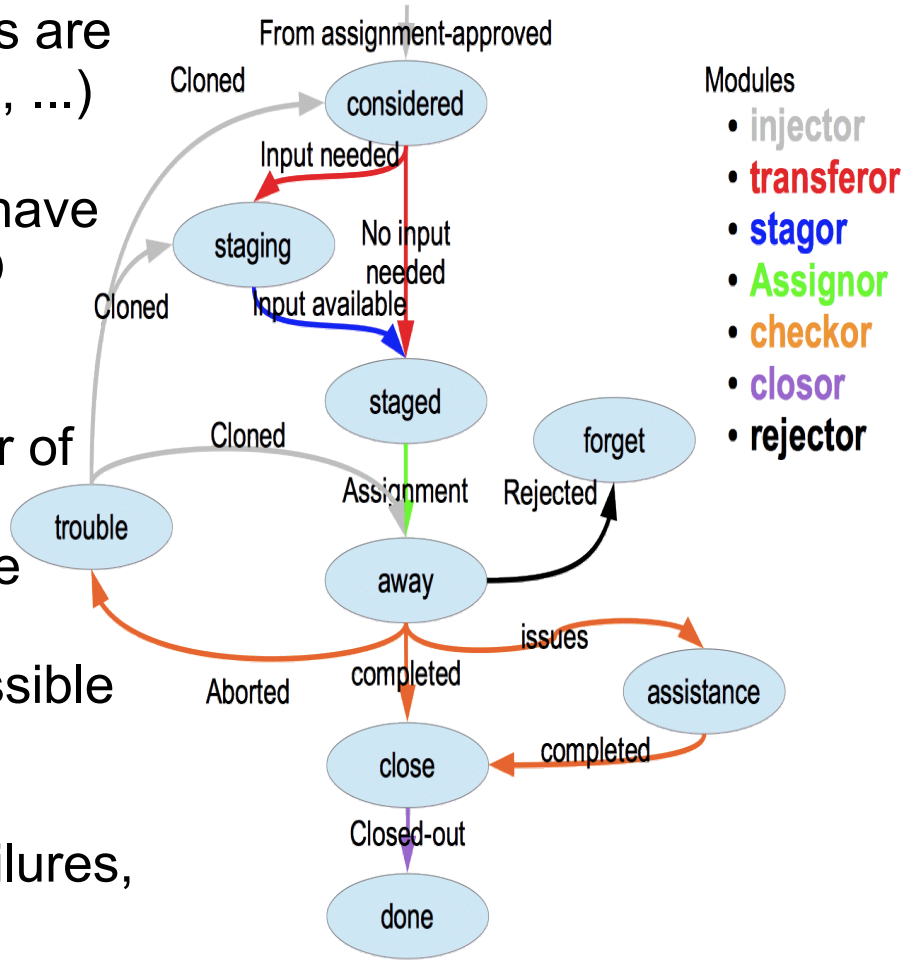
➔ **Automatic assignment** to as many sites as possible
  - ✔ Balance job priority with site bottlenecking

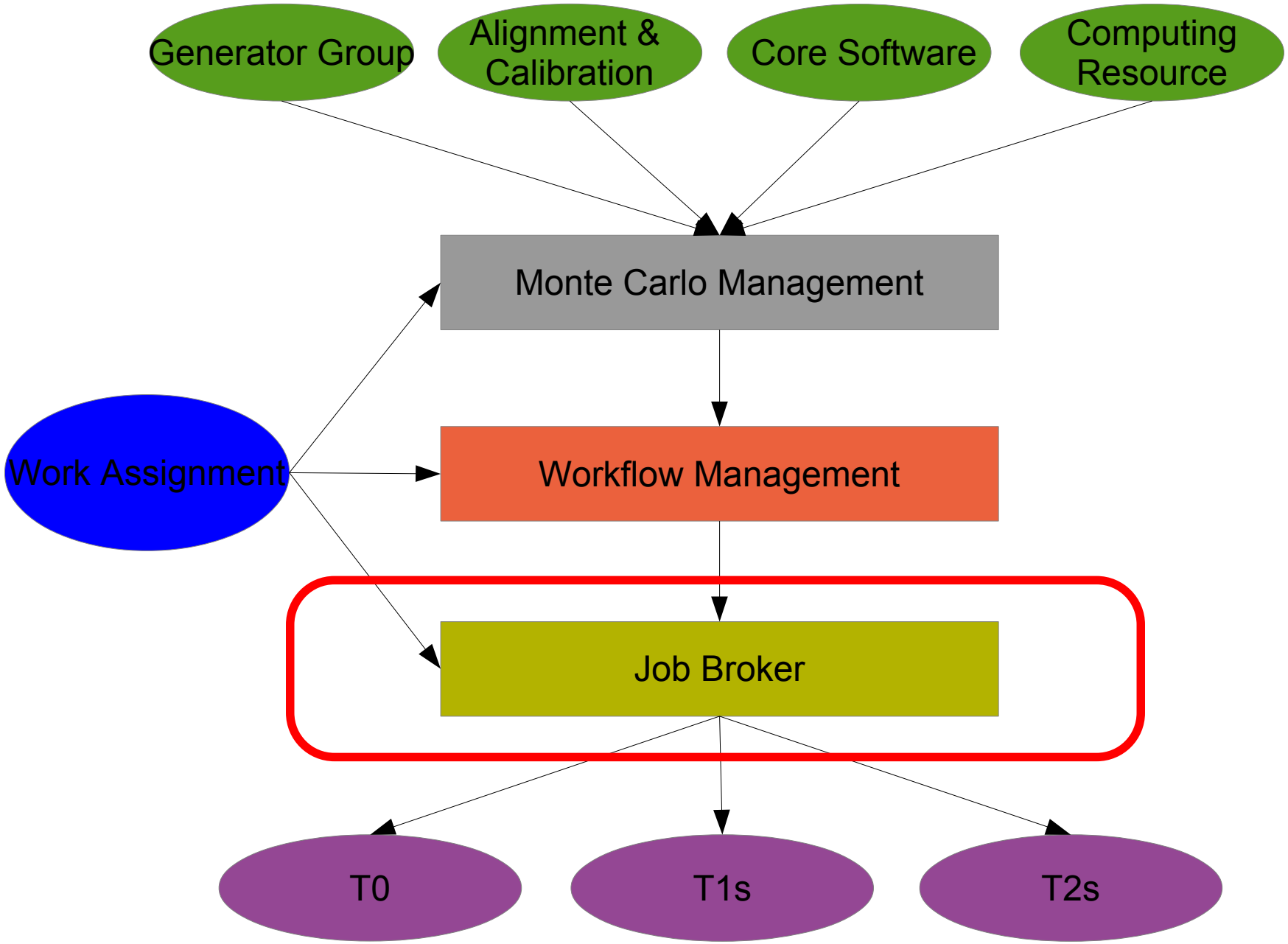
✔ Most workload are without issue (transfer, job failures, site issues, ...) and fully **handled automatically**

✗ Issues are dealt with increasing automation

➔ Automation boosts overall productivity

Unified® state diagram

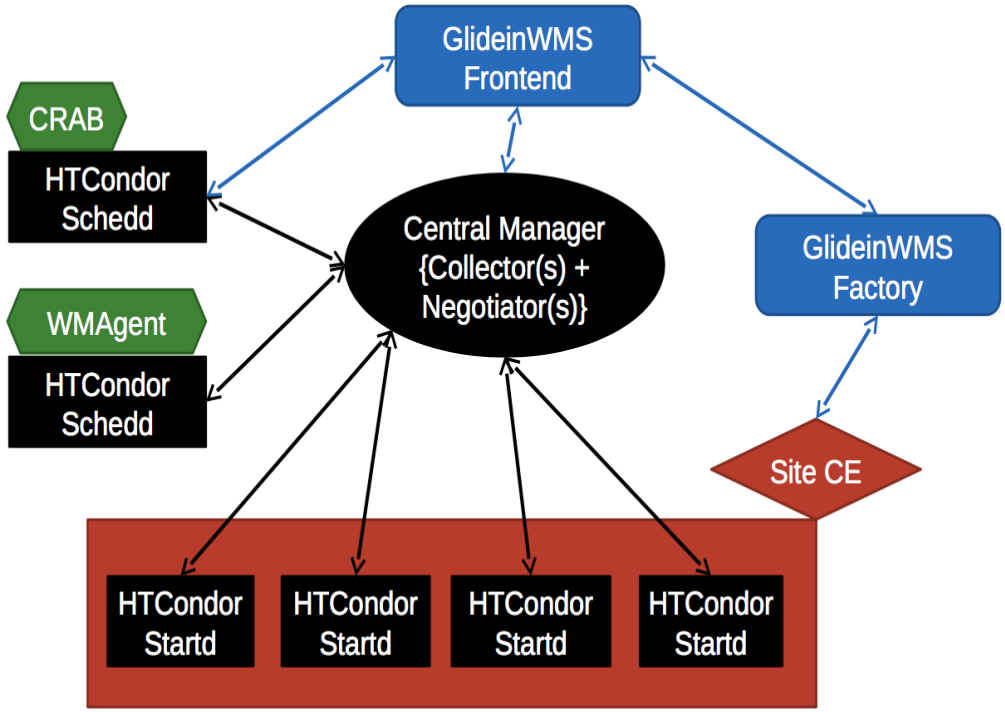



Generator Group
Alignment & Calibration
Core Software
Computing Resource
Monte Carlo Management
Work Assignment
Workflow Management
Job Broker
T0
T1s
T2s

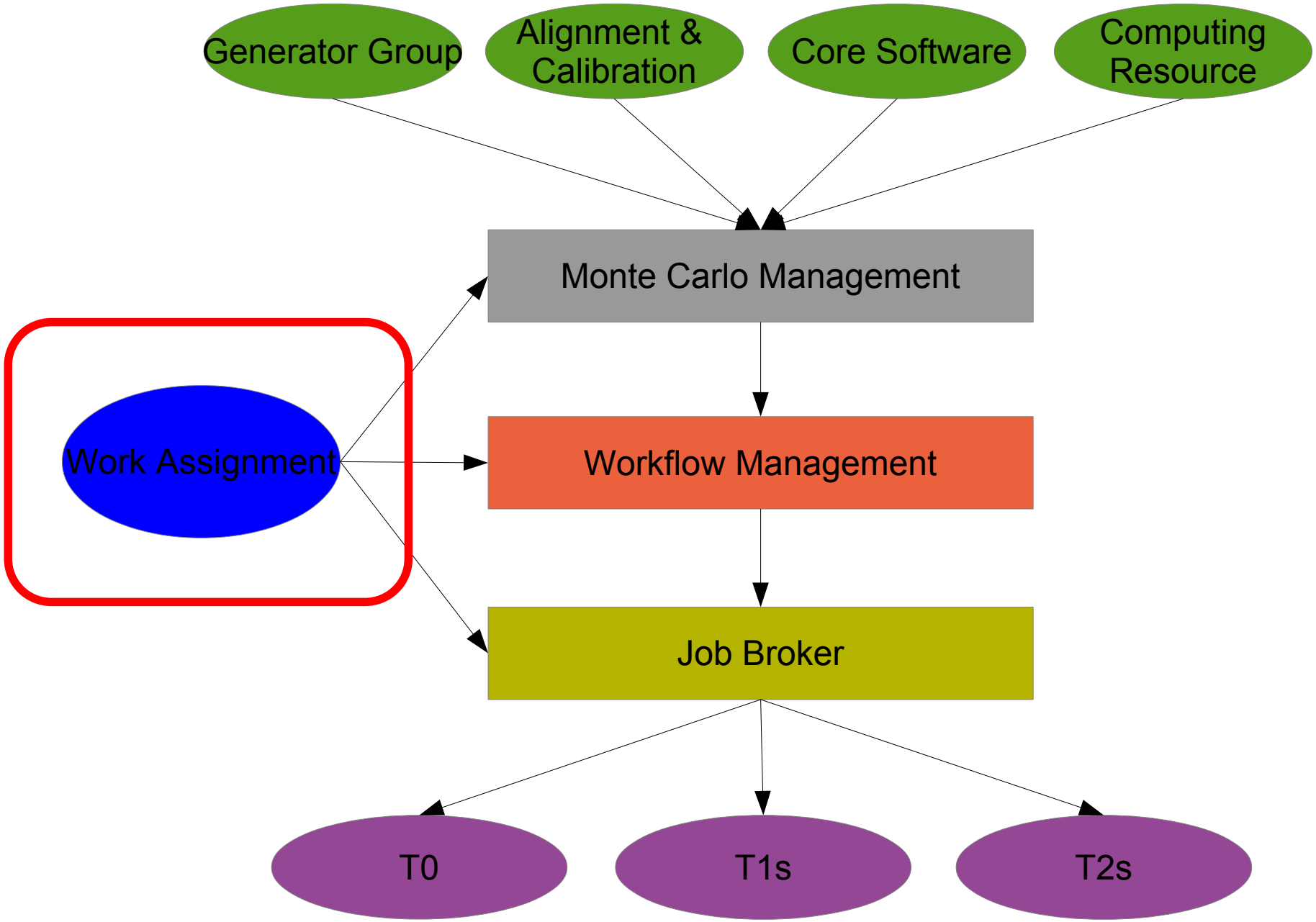
Job Brokering

- **Shared resource** between analyzer and central production in a global pool
  - ✔ T0 production on a specific pool
- Use of HTCondor + glidein mechanism
  - ✔ Wrapper job : pilot running on site
  - ✔ Receive and execute trusted jobs
- Double stage of matchmaking
  - ✔ Jobs to resource (start pilots)
  - ✔ Jobs to pilots (claim pilots)
- Migrated for a large fraction to **multi-core partitionable pilots**
  - ✔ Allows multi-thread application
  - ➔ Moving most workflows to 4+ threads
- High Throughput computing solution

- ~30 schedds for production and analysis with redundancy
  - ✔ Record **200k concurrent jobs**
  - ✔ **Steady >150k job**
- ➔ Constantly working towards scaling up



REFERENCE


Generator Group
Alignment & Calibration
Core Software
Computing Resource
Monte Carlo Management
Work Assignment
Workflow Management
Job Broker
T0
T1s
T2s

Work Optimization

- Site might come out of production status because of schedule intervention, emergency shutdown, intermittent failures, ... (see sites monitoring)
- Workload backlog might develop on local site queue
  - ✓ **Mechanism to overflow** to neighboring site
  - ✓ **Reposition blocks** of data accordingly
  - → Quicken delivery with reliable remote read
  - → Can be used to divert work to resource becoming available

- Jobs requirement are just estimation from limited test-run
  - ✓ Job memory requirement is edited when possible to values observed in running over the grid
  - ✓ Job runtime requirement can be edited
  - → **Better partitioning of resource** into job slots

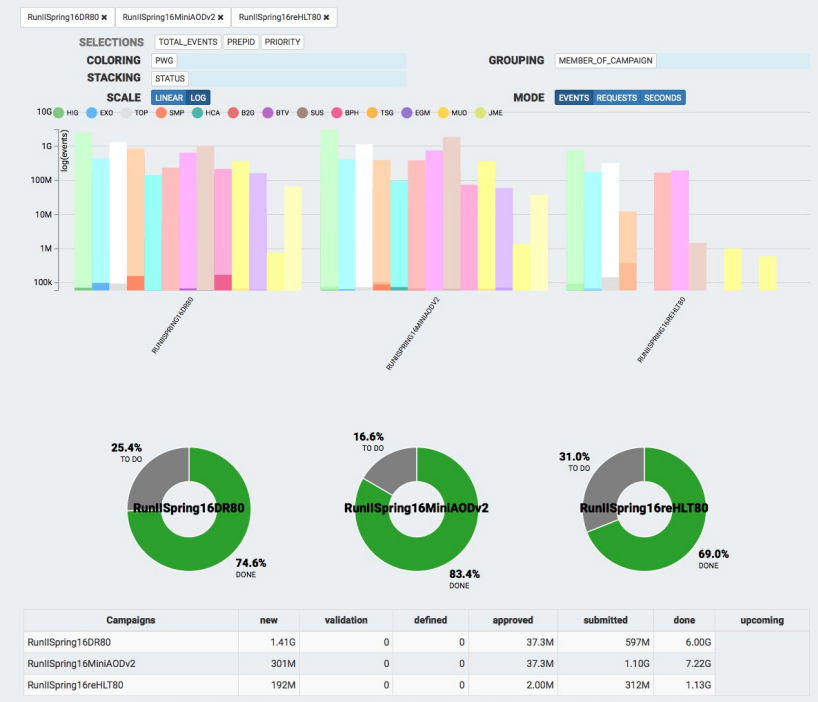
- Shorten workflow processing above agreed completion fraction and running time

→ Working towards much **more flexibility**, using a more granular data-driven processing strategy

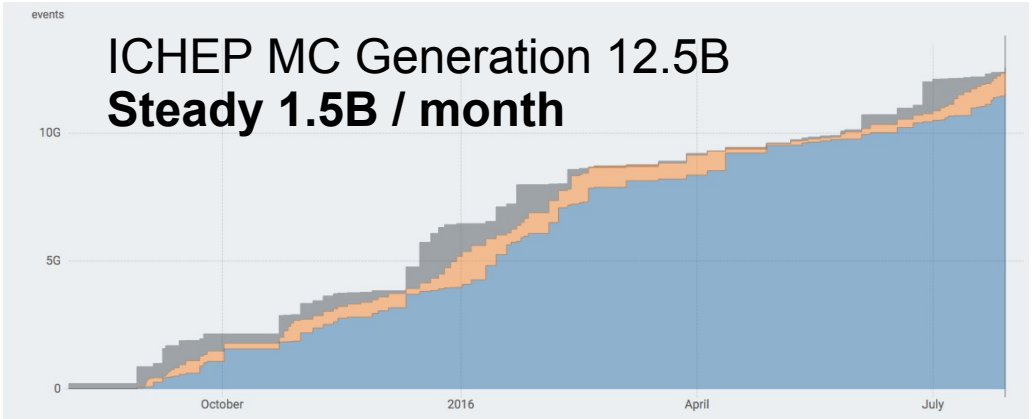
Production Monitoring

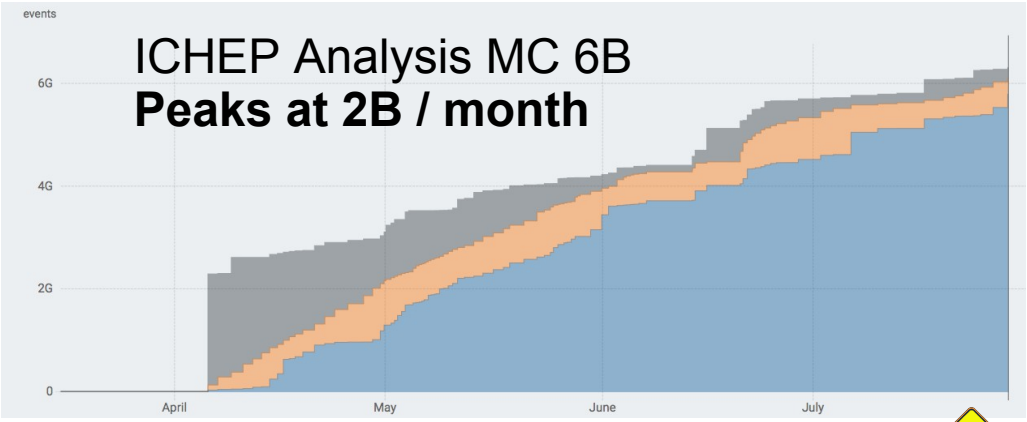
Sample Monitoring

- **P**roduction **M**onitoring **P**latform (pMp)
- Display current statuses of campaigns
- Track **evolution of single requests** and aggregates several ways
- Help guide the user waiting for samples
- Allows for **production planning**



Campaigns	new	validation	defined	approved	submitted	done	upcoming
RunIISpring16DR80	1.41G	0	0	37.3M	597M	6.00G	
RunIISpring16MiniAODv2	301M	0	0	37.3M	1.10G	7.22G	
RunIISpring16reHLT80	192M	0	0	2.00M	312M	1.13G	

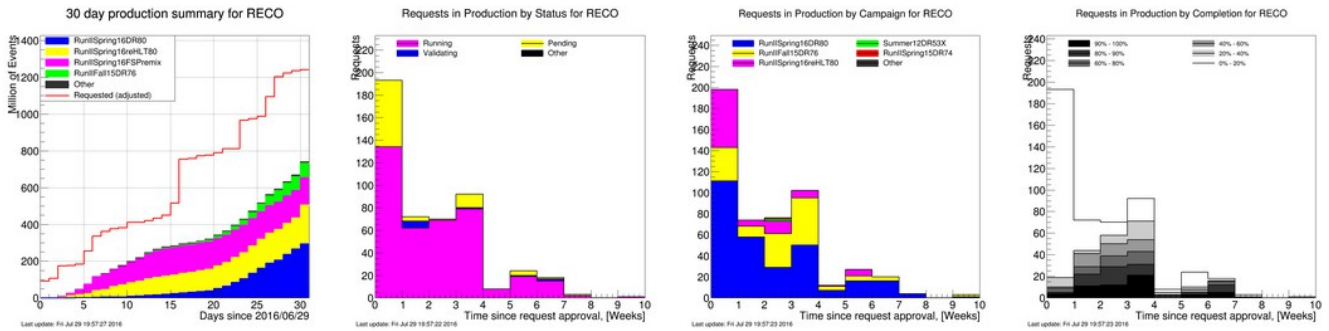


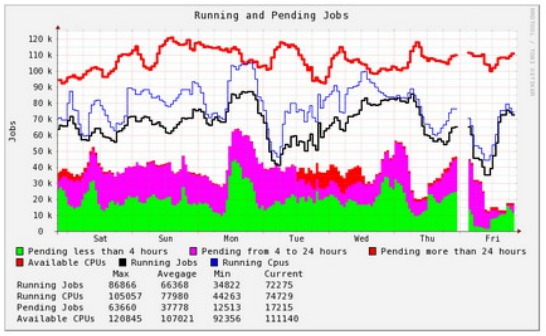


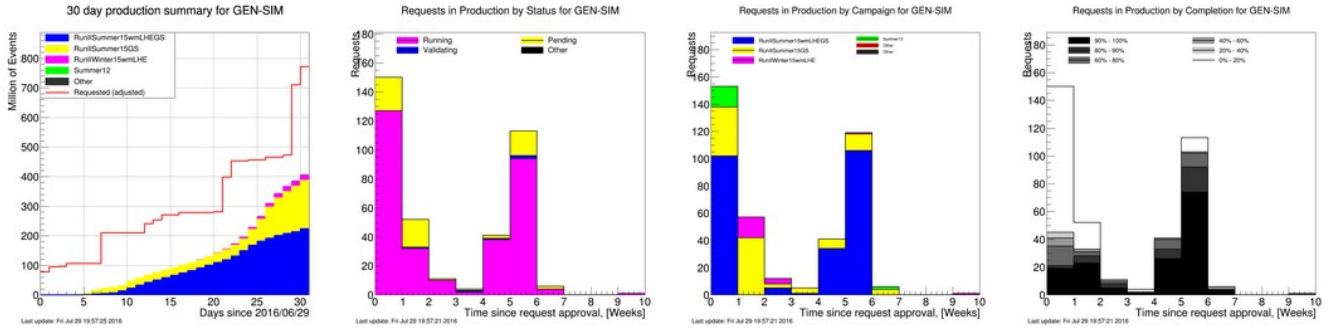
REFERENCE

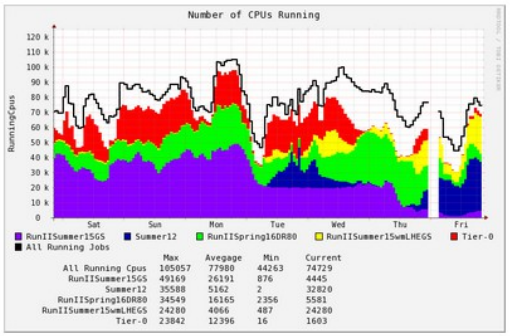
Production Monitoring

- Amount of work left for production at a glance
- Monitors overall resource utilization
- Identifies tails in production
- Aggregate information from several services
- ✔ Average 2000 datasets released per week
- ✔ **Peak 5000 analysis datasets** per week







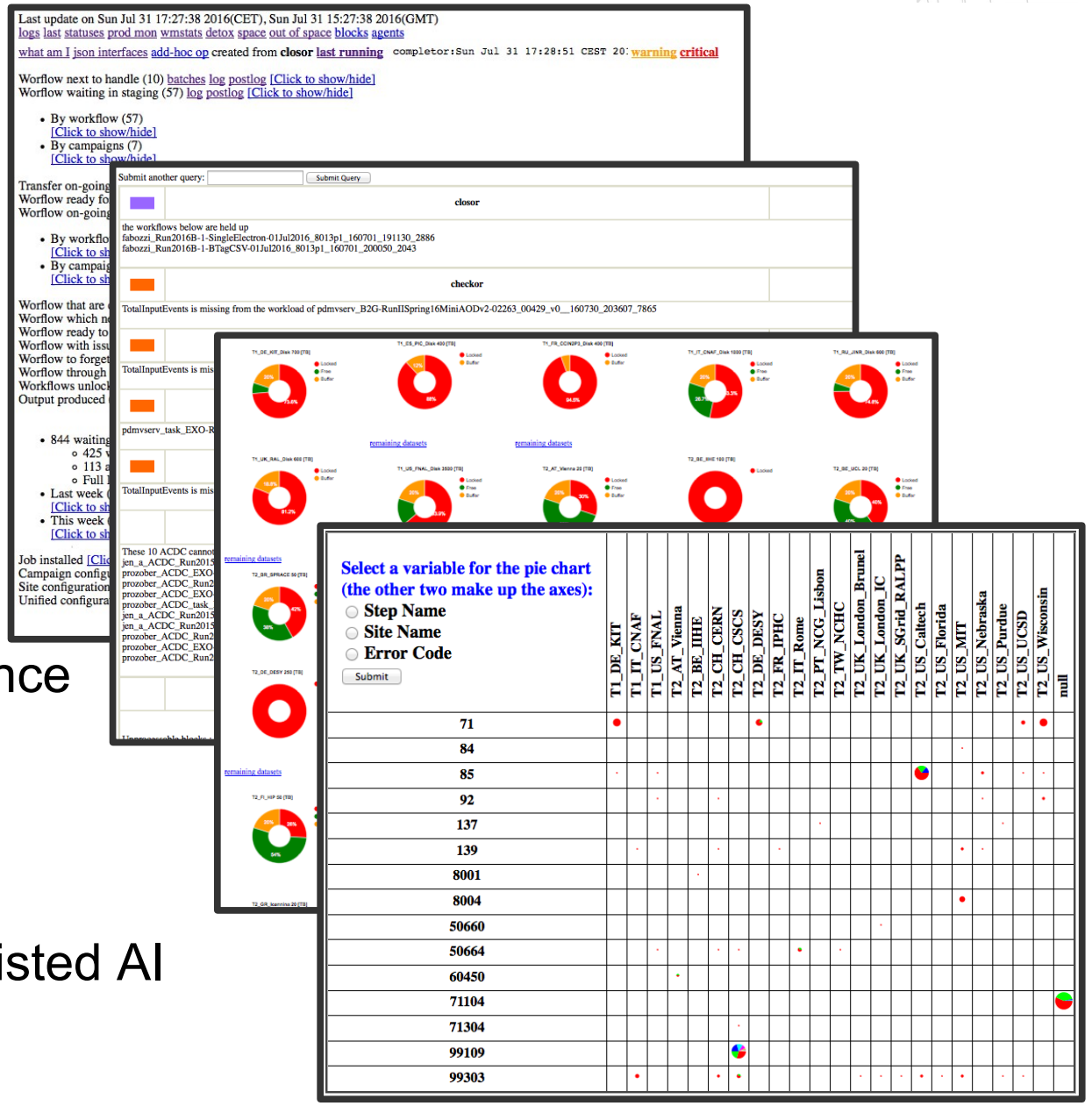


REFERENCE

Operation Monitoring

- Overview of work at each level
  - ✔ Provide links to all relevant services
- Logging **heart beats**
  - ✔ Dashboard of **critical items**
  - ✔ Single **workflow history**

- Expose information relevant to other services in json
- Production disk space at a glance
- **Notification** to requesters
  - ✔ Log redundancy
- Display all relevant errors
  - ✔ **Guidance** to operators
  - ➔ Working towards human-assisted AI operation



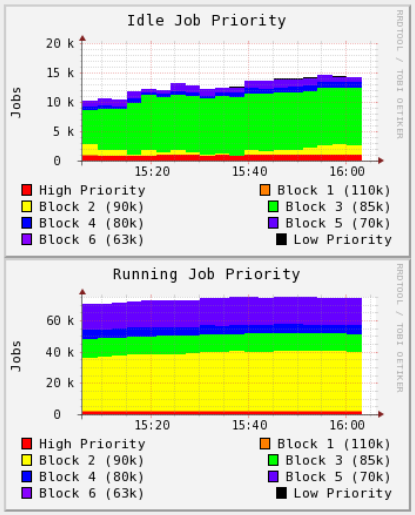
Job Monitoring

- Aggregate and present information from **HTCondor and Glideinwms**
  - ✔ Number of **CPU and jobs per task**, per workload, per site, ...
  - ✔ **Status of sites** with respect to HTCondor
  - ✔ Show the **load on the schedd**
  - ✔ Job **production/analysis share** at sites
- **Feedback loop** on how sites, tasks, and jobs are performing

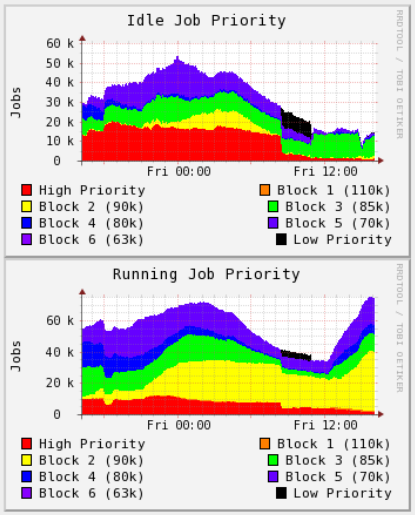
➔ Working on using more of the feedback loop for **processing optimization**

Priority Type	Running	Idle
High Priority	1832	1022
Block 1 (110k)	663	0
Block 2 (90k)	37748	1674
Block 3 (85k)	11187	9762
Block 4 (80k)	5452	934
Block 5 (70k)	17412	788
Low Priority	0	6

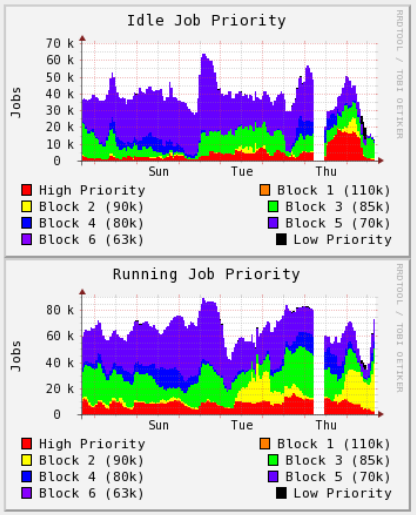
Last Hour



Last Day



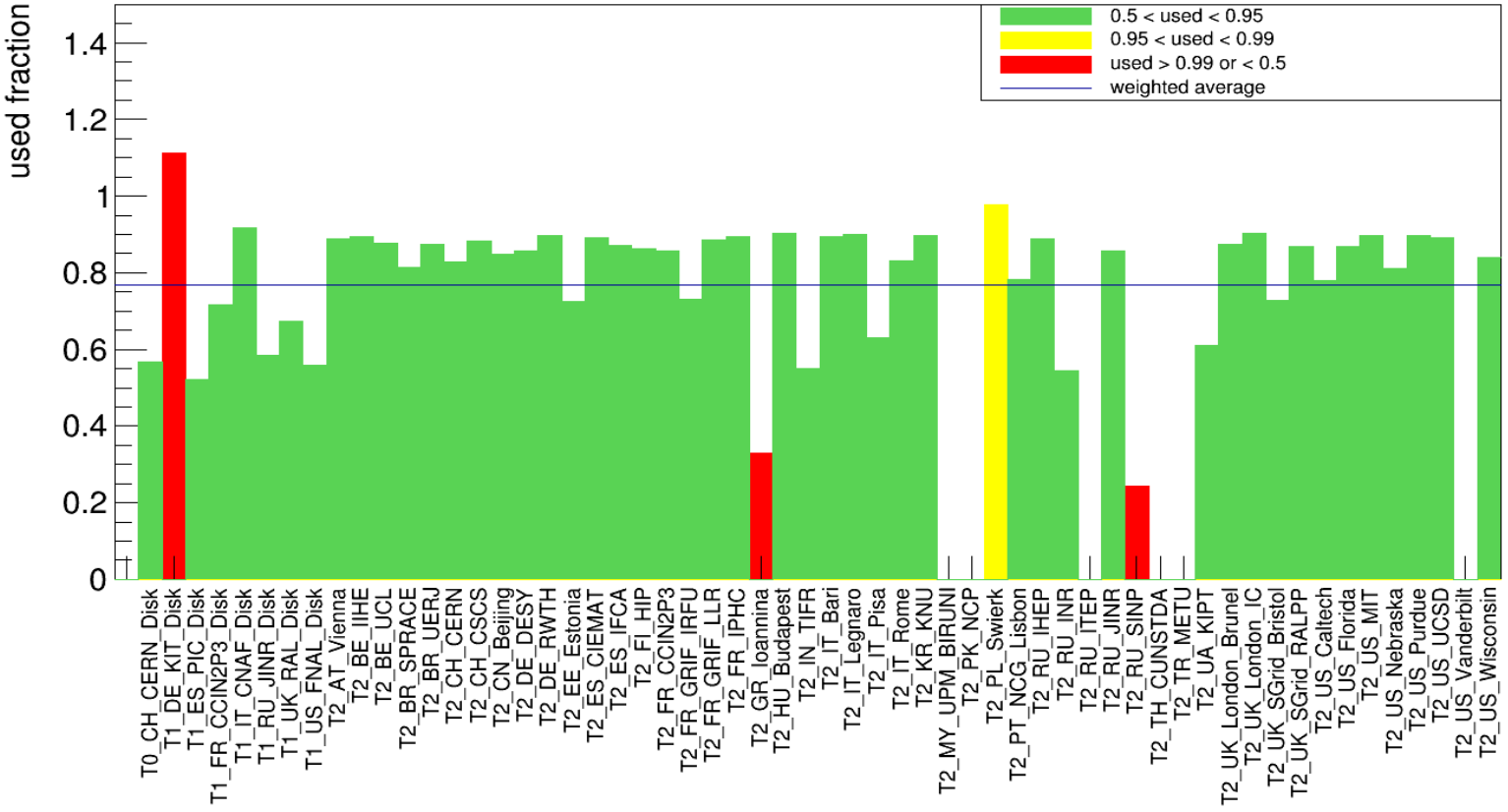
Last Week



REFERENCE

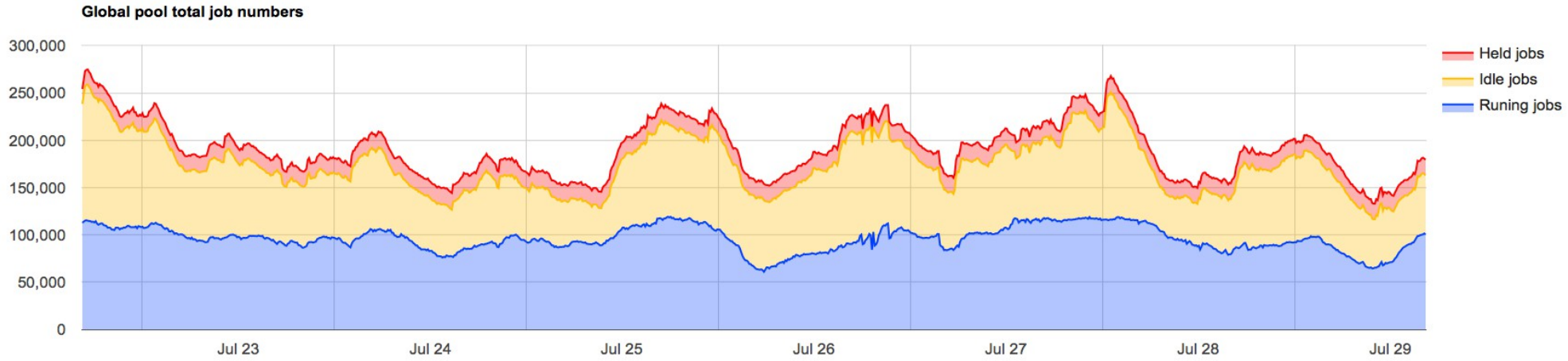
Storage Control

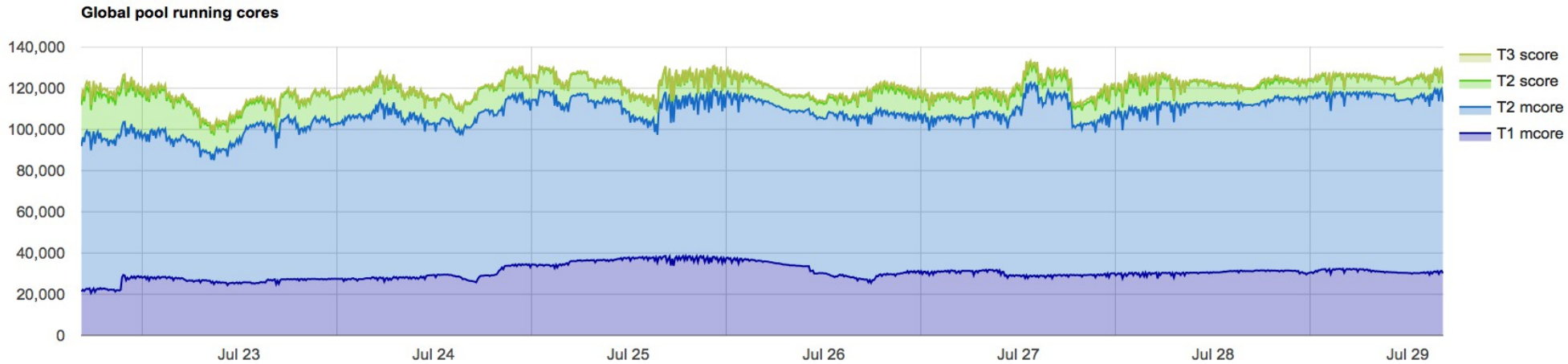
- Available tape space monitored
  - ↙ **Fair-share distribution** to long term storage
- Disk space managed with virtual quota for production and analysis
  - ↙ Automatic transfer and deletion
- ➔ Developing production strategy with a **smaller disk footprint**



REFERENCE

Resource Monitoring





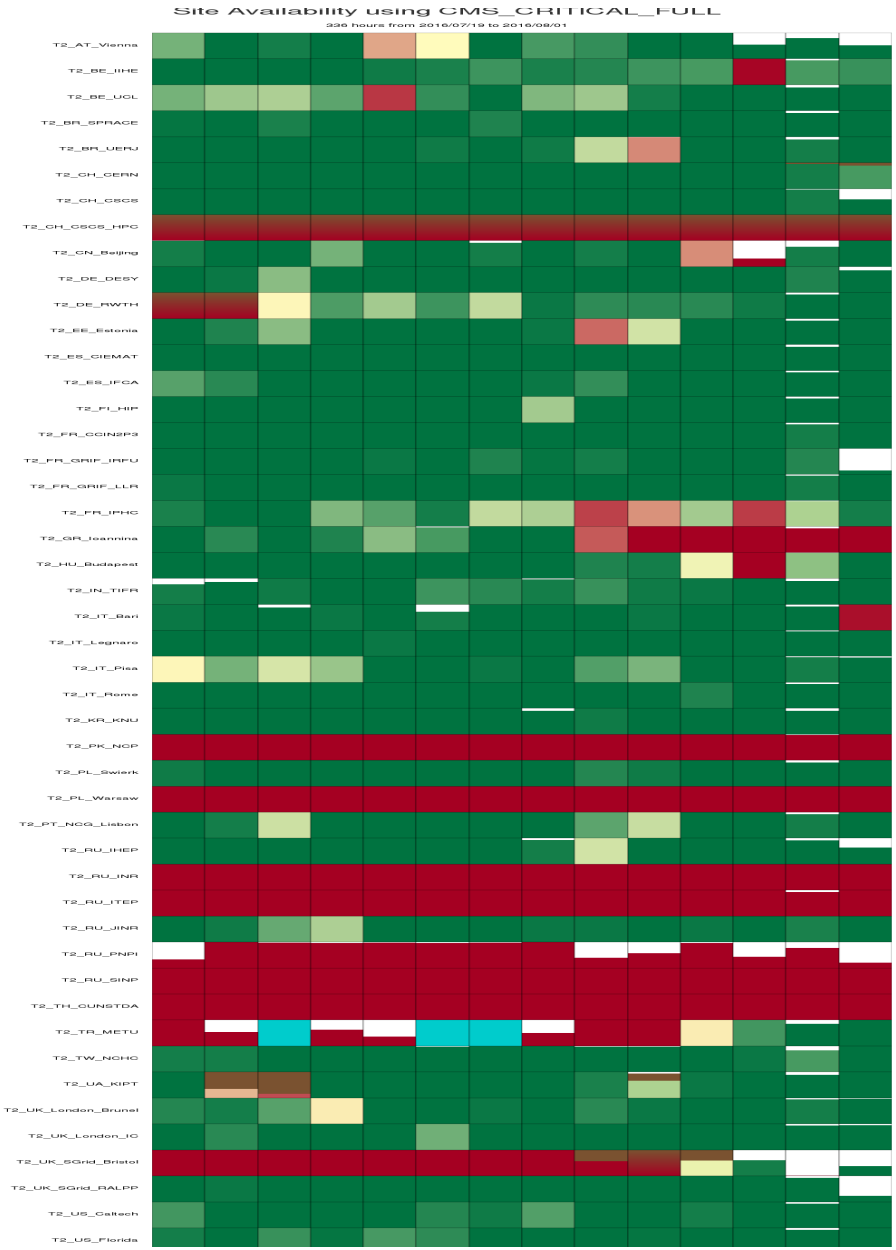
- Steady 100k jobs running for CMS (production and analysis)
- Large contributions from T2
- Large fraction of multi-core pilots
- Spot trend in resource utilization

REFERENCE

Sites Availability

- Aggregate **live information about sites**
  - ✔ Site Availability Monitor (SAM) compute and storage services
  - ✔ Hammer Cloud (HC) ability to run jobs
  - ✔ Data Transfer (PhEDEx) transfer links
- Determine the site ready-ness

➔ Working towards **more dynamic and specific** site status evaluation



SUMMARY

- Large scale production and reprocessing for RunII
- Monte-Carlo Production and Data Reprocessing preparation well orchestrated
- Experience gained invested in development
- Constantly working on improvements