Multiple Scenarios Oriented HTC Computing System Based on HTCondor at IHEP

JIANG, Xiaowei (姜晓巍)
On behalf of Computing Team of IHEP CC
ISGC 2022
2022/03/22

OUTLINE

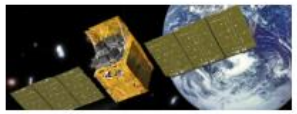
- Background
- Local High Throughput Cluster
- Grid Sites
- Distributed HTC Pool
- Customized Clusters for Edge Sites
- Real-time Computing for Satellite Project
- Summary

Background

- IHEP is hosting or attending in >15 experiments around the field of high energy physics
- Diverse computing requirements
  - Traditional local htc cluster
  - Grid sites (LHC, BELLEII,JUNO)
  - dHTC pool (Glidein-based)
  - Realtime computing for satellite project
  - Customized clusters for brother institutes

BESIII (Beijing Spectrometer III at BECPII)

JUNO (Jiangmen Underground Neutrino Observatory)



HXMT(Hard X-Ray Moderate Telescope)

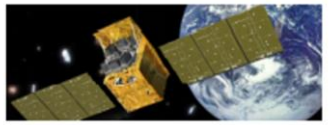
China Spallation Neutron Source

LHAASO (Large High Altitude Air Shower Observatory)

HEPS (High Energy Photon Source)

- All the above requirements are got solutions based on HTCondor

5 Solutions for Multiple Experiments



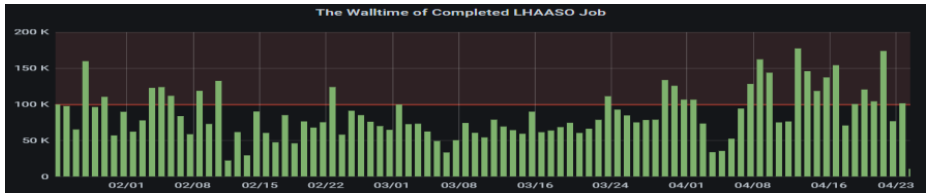
HXMT(Hard X-Ray Moderate Telescope)

HTCondor Software Suite

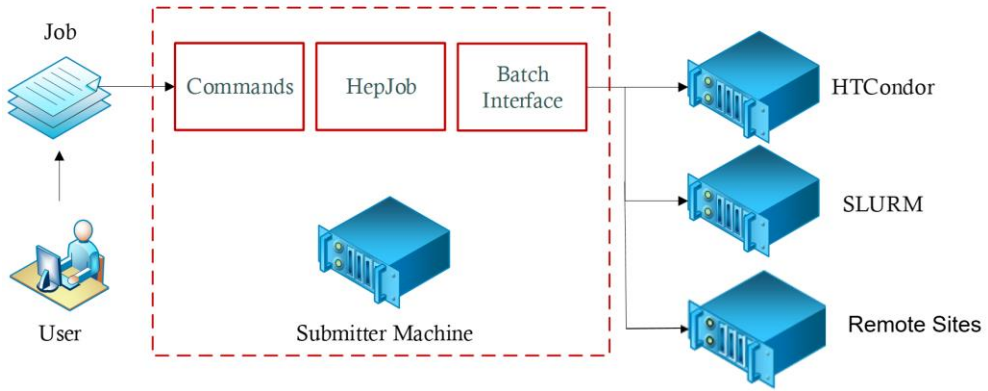
- Local Cluster
- Grid Sites
- dHTC Pool
- Customized Pools
- Real-time Computing

Local HTC Cluster – R&D

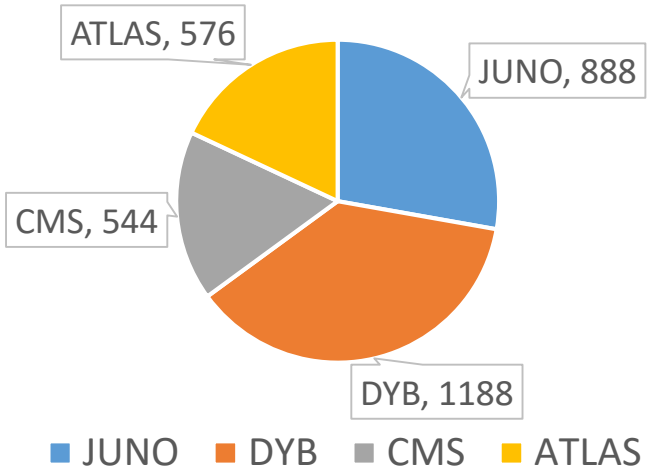
- Share resources between experiments
  - A share policy implemented by accounting group quota

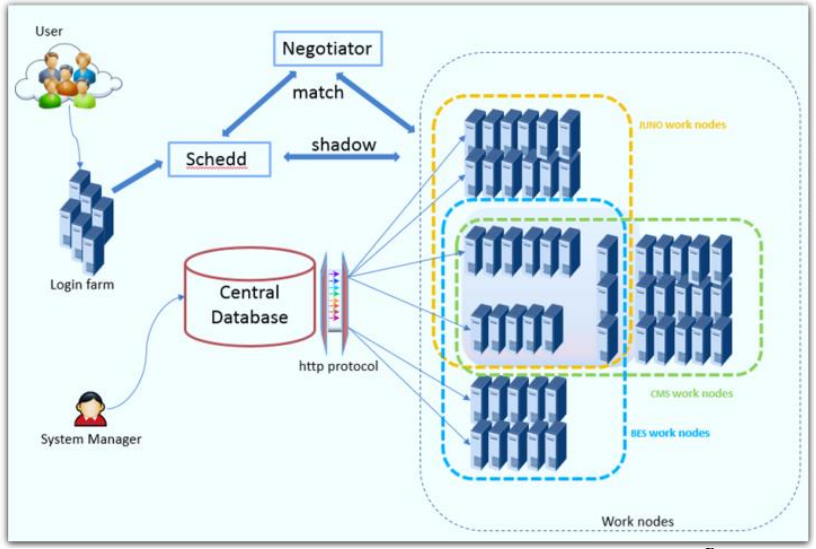


- OMAT
  - Decide the shared group
  - Dynamically add/remove a wn according to the health
- HepJob
  - A unified frontend tool for all experiments



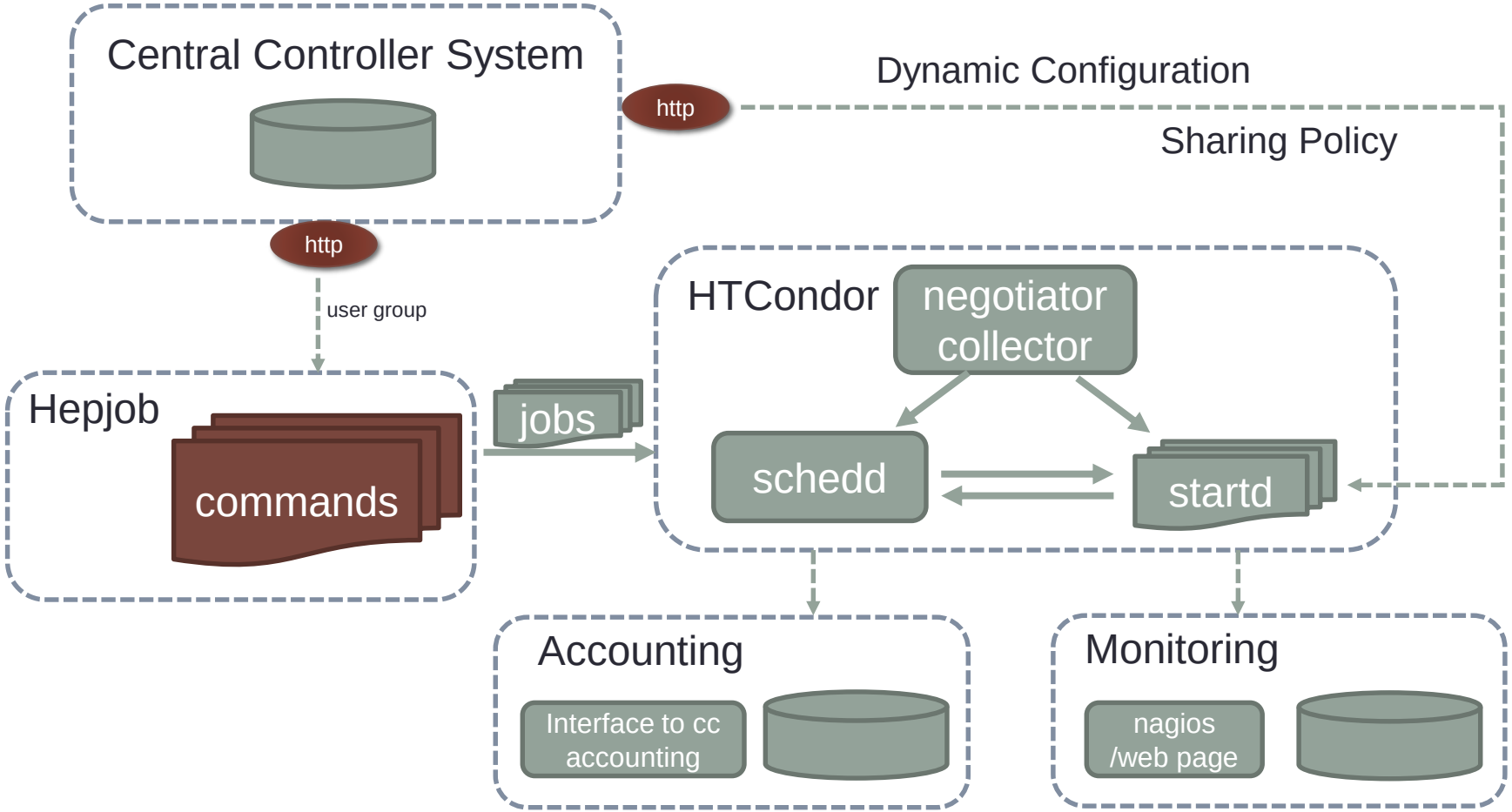
HTCondor Cluster Sharing Policy





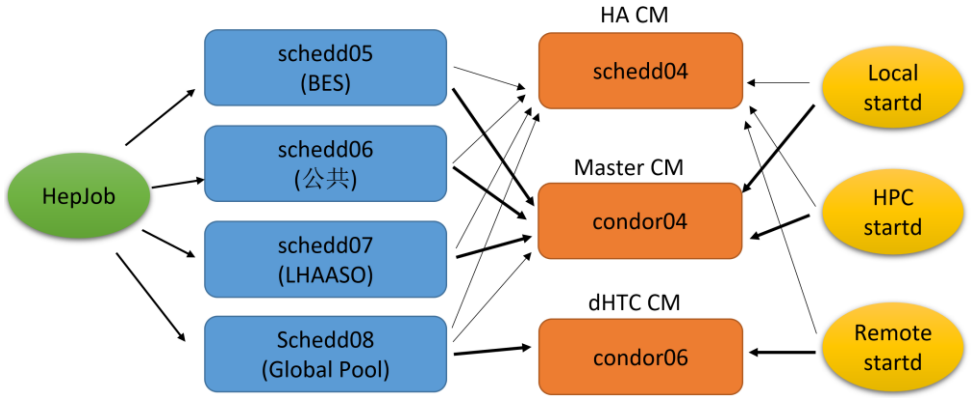
Local HTC Cluster – all in one

- A basic complete HTC system
  - Job Entrance/central controller/accounting/monitoring/…

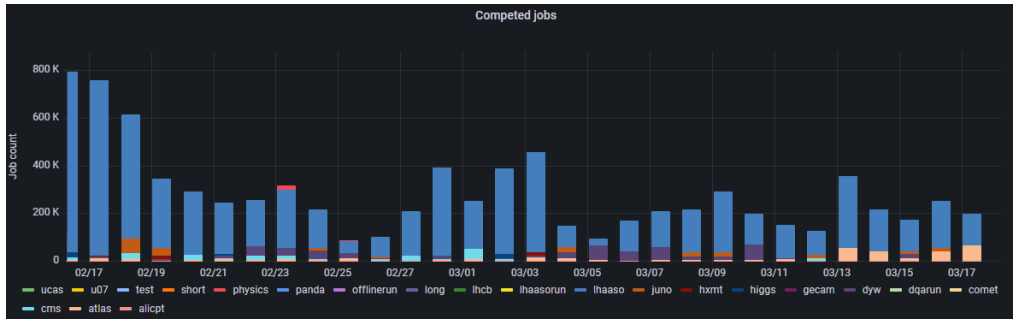


Local HTC cluster – Current Status

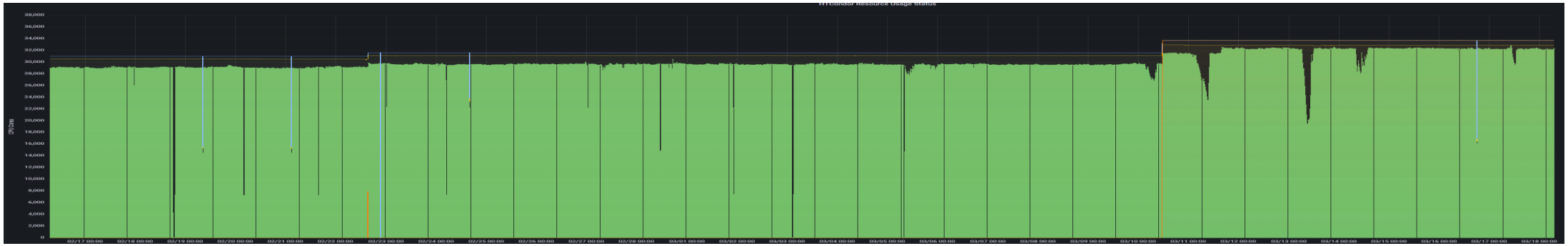
- HTCondor status
  - 4 SchedDs: mapping by specific groups
  - 3 CMs: Main CM&HA CM
  - HPC&HTC&Cloud: share resources



- Job Status (last 30 days)
  - Avg. 524k Jobs completed per day

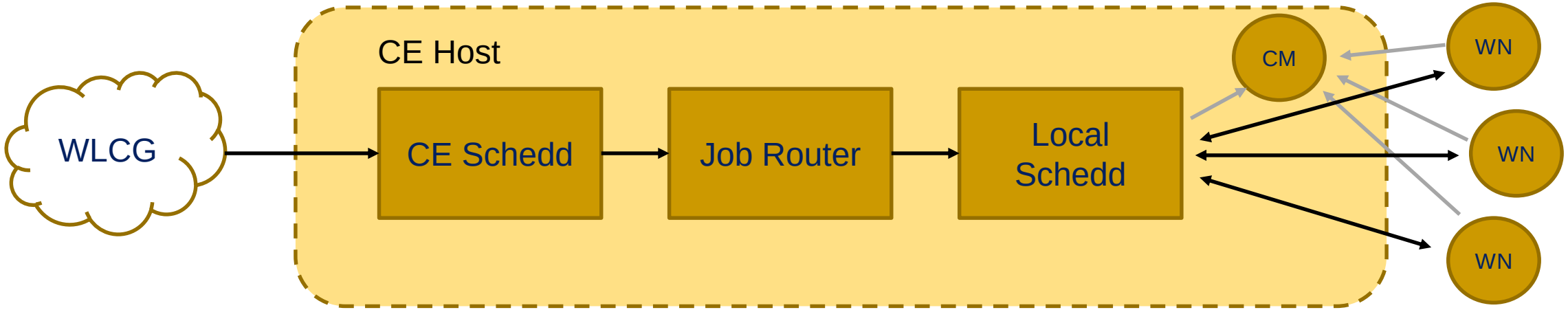


- Resource status
  - >90% resource utilization rate can be counted (~34,000 CPU cores)



Grid Sites

- A traditional CE
  - HTCondorCE as CE, HTCondor as batch system

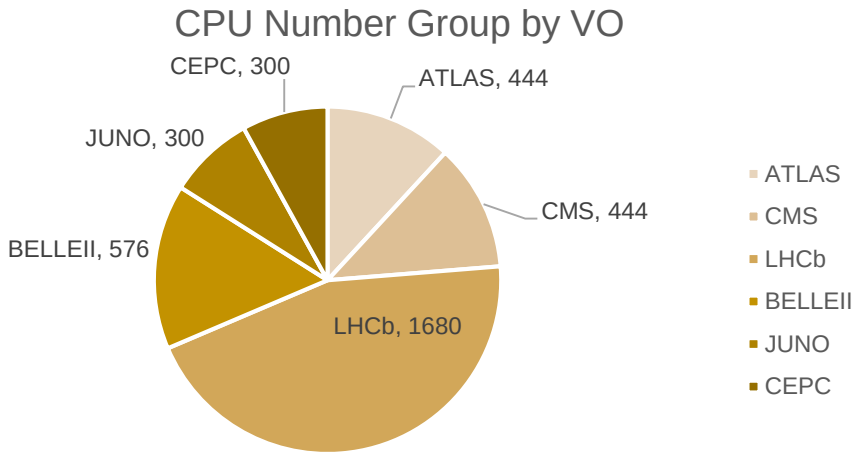


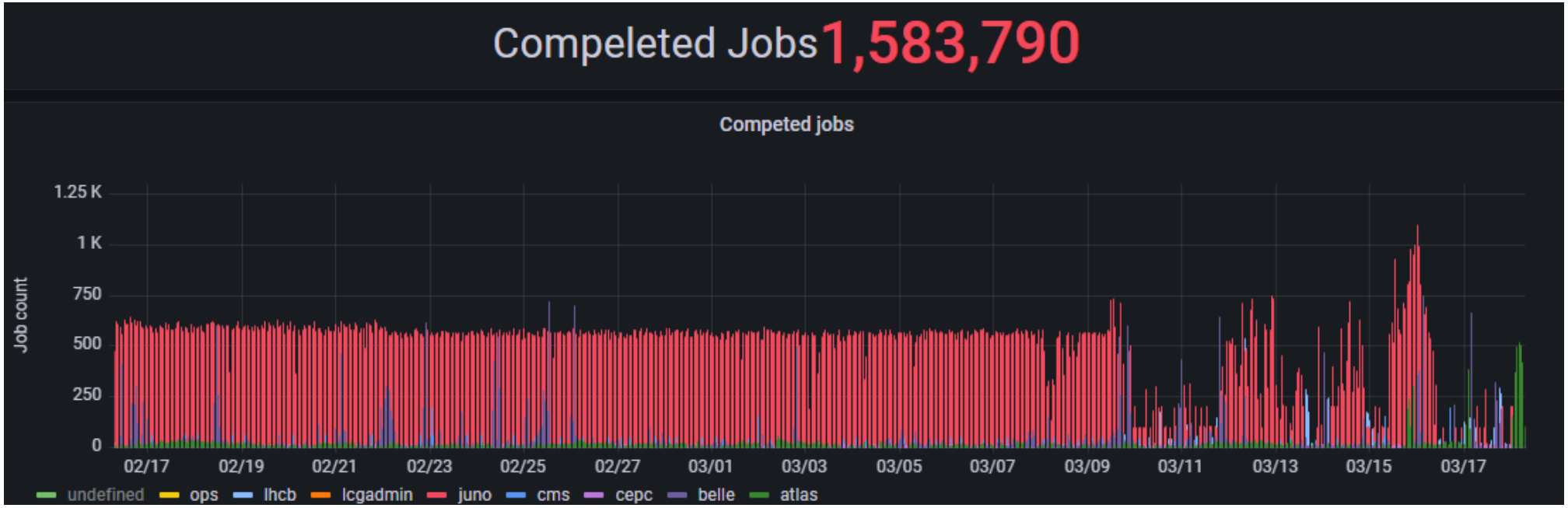
- All the sites are sharing a single HTCondorCE and batch HTCondor
  - Pilot jobs are mapped by VOs in the job router
  - SAM jobs are mapped to a single group

```
JOB_ROUTER_ENTRIES = \
   [ \
     TargetUniverse = 5; \
     name = "cms_pilot"; \
     Requirements = regexp("\/cms\/Role\=pilot", TARGET.x509UserProxyFirstFQAN); \
     eval_set_AccountingGroup = strcat(x509userproxyvoname, ".", Owner); \
     eval_set_AcctGroup = strcat(x509userproxyvoname); \
     eval_set_AcctGroupUser = strcat(Owner); \
     delete_SUBMIT_Iwd = true; \
     set_WantIOProxy = true; \
     set_default_maxMemory = 100; \
     #set_OriginalMemory = 100; \
   ] \
   [ \
     TargetUniverse = 5; \
     name = "lcgadmin"; \
```

Grid Sites – current status

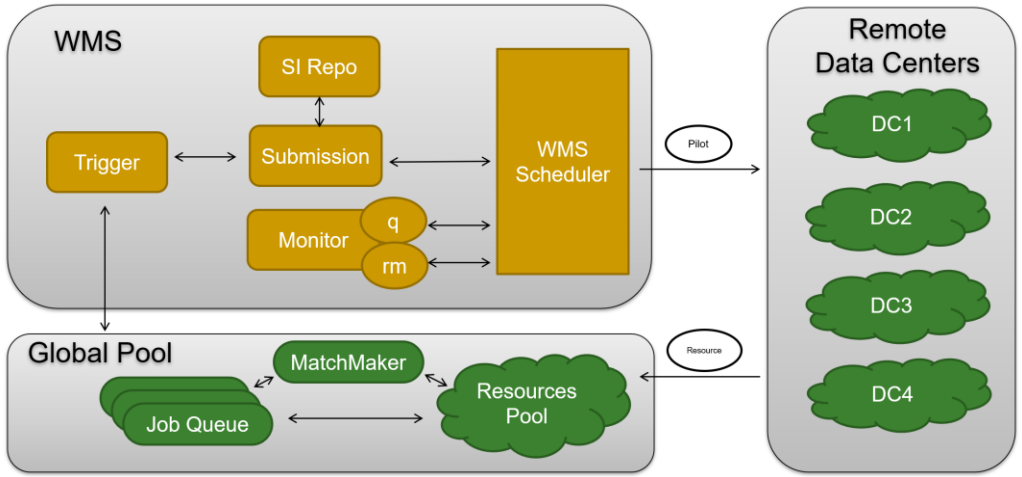
- Serve for several sites
  - ATLAS，CMS，LHCb （WLCG Tier 2）
  - BelleII （Tier 2）
  - JUNO （Tier 1）
  - CEPC (Tier 1)

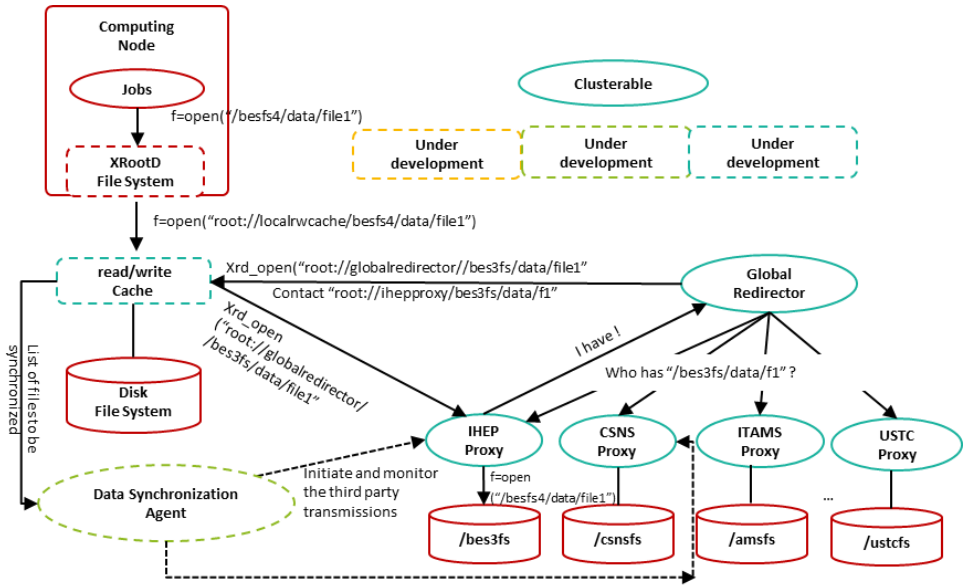




Distributed HTC Pool

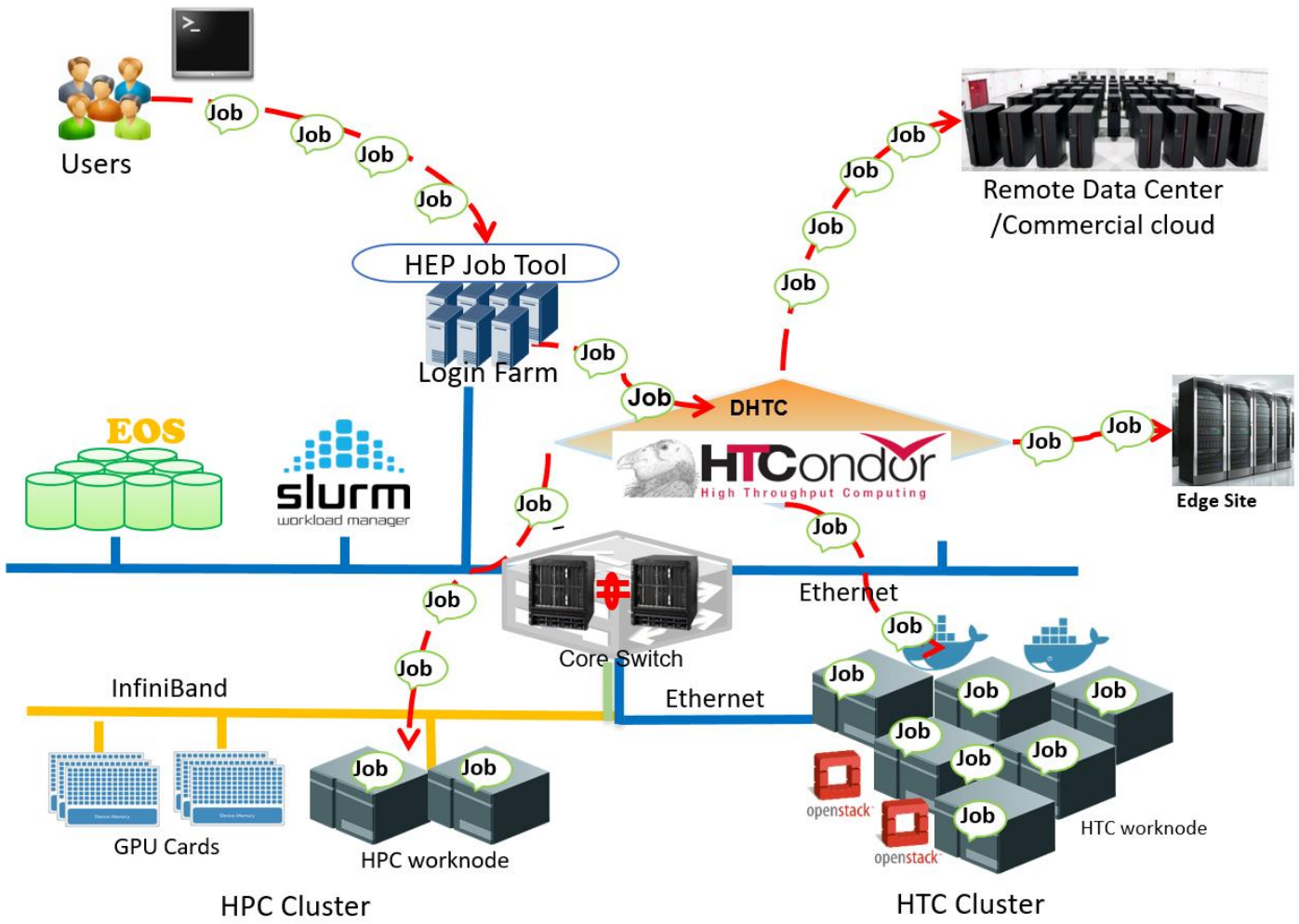
- To share resources between separated data centers and edge sites
- Computing Part
  - Based on HTCondor Glidein
  - User interface provided by HepJob
- Data access and transfer
  - XRootd proxy & Xcache
  - HTCondor transfer
- Certificate
  - Kerberos Tokens for user&job
  - IDTokens for Daemon





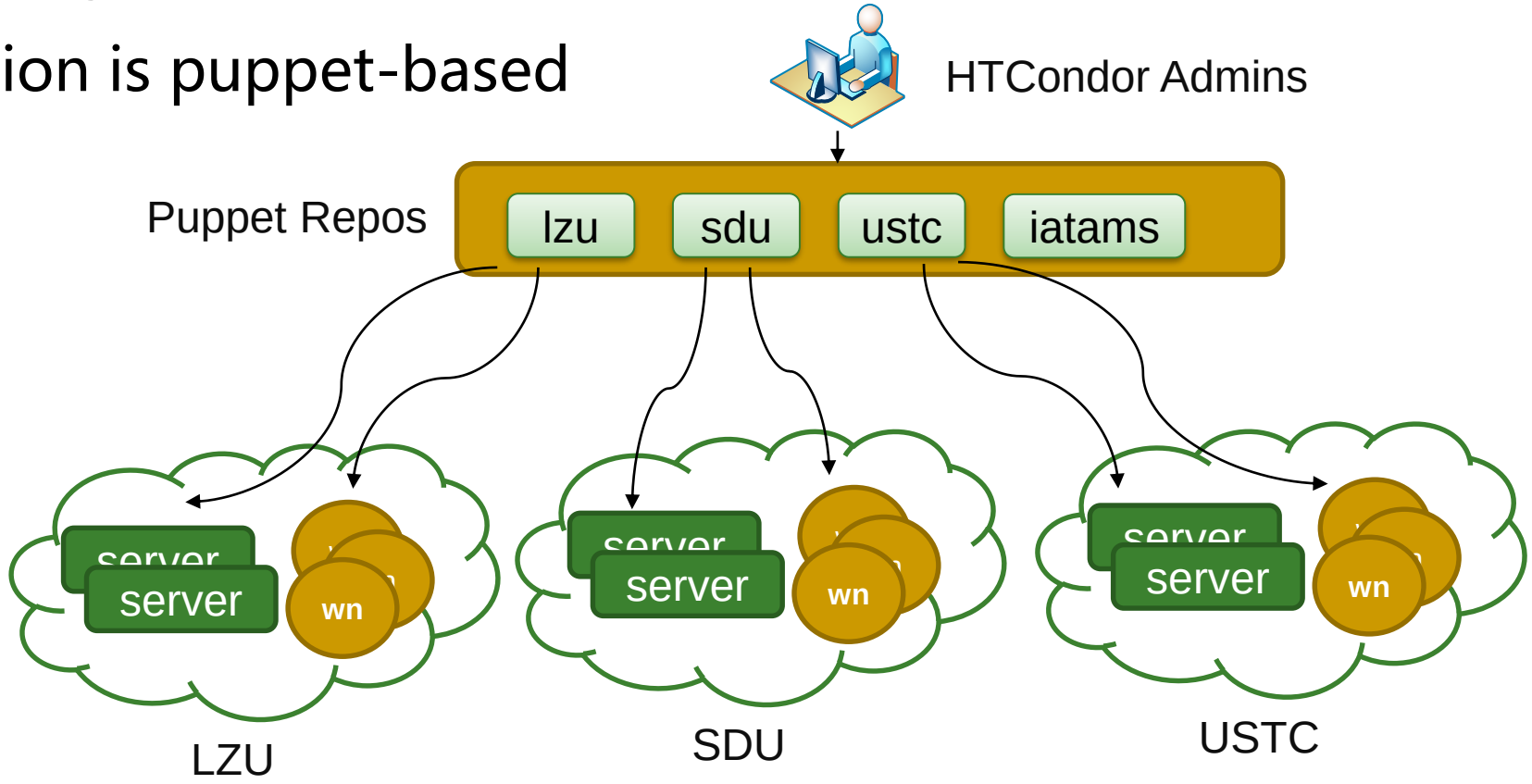
dHTC - Current Status

- Data Centers
  - Beijing Site
  - Dongguan Site
- Edge sites
  - LZU, SDU, IHEP
- Batch system
  - HTCondor & Slurm
- Main application
  - LHAASO WCDA simulation
  - LHAASO WFCTA simulation
  - BES simulation（doing）



Customized Cluster for Edge sites

- The main problem is how to centrally mange the edge sites (for htcondor)
  - Server configuration: login, schedd, cm
  - Startd configuration
- The solution is puppet-based



Customized Cluster for Edge Sites – Current Status

- Edge sites: USTC, SDU, LZU, IATAMS, …
- Support resources: CPU/GPU; single cores/multi cores

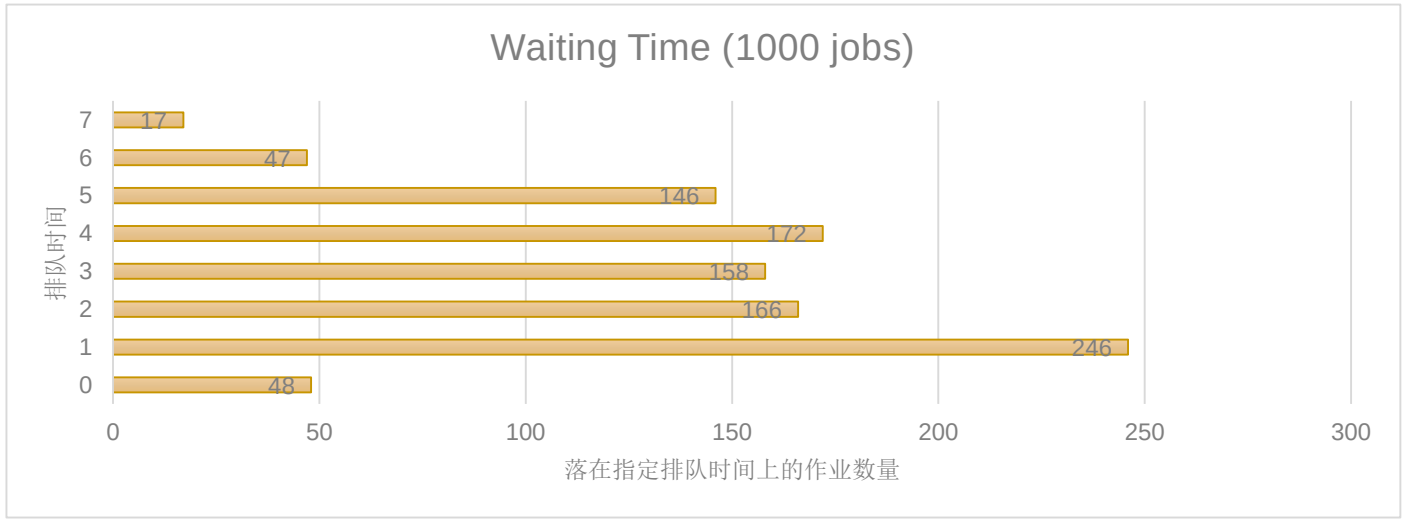
Monitoring of One Platform for Multiple Data Centers
last 1 week

Sites	CPU Resources (CPU Cores)	CPU Resource Utilization	Disk Storage Capacity	Data Storage	Completed Jobs HTC&HPC	Job Run Time (CPU Hour)
IHEPCC	39,996	88.97%	72.12 PB	43.22 PB	2,752,083	5,953,860
DongGuan	31,120	39.51%	6.38 PB	2.30 PB	493	1,399,793
DaoCheng	3,616	47.93%	4.27 PB	3.54 PB	2,036	344,790
CSNS	5,852	31.24%	802.7 TB	375.5 TB	469	290,286
SDU	1,180	8.863%	352.9 TB	238.8 TB	716	16,538
USTC	3,018	26.97%	1.17 PB	767.2 TB	3,539	145,605
LZU	1,768	2.091%	341.8 TB	217.3 TB	585	6,215

Real-time Computing for Satellite Project

- Job should be started in real time
  - A pure cluster dedicated for satellite experiment
  - Some changes on negotiation configurations
- Most Job can be scheduled in short time
  - Average queuing time is ~2.9 seconds

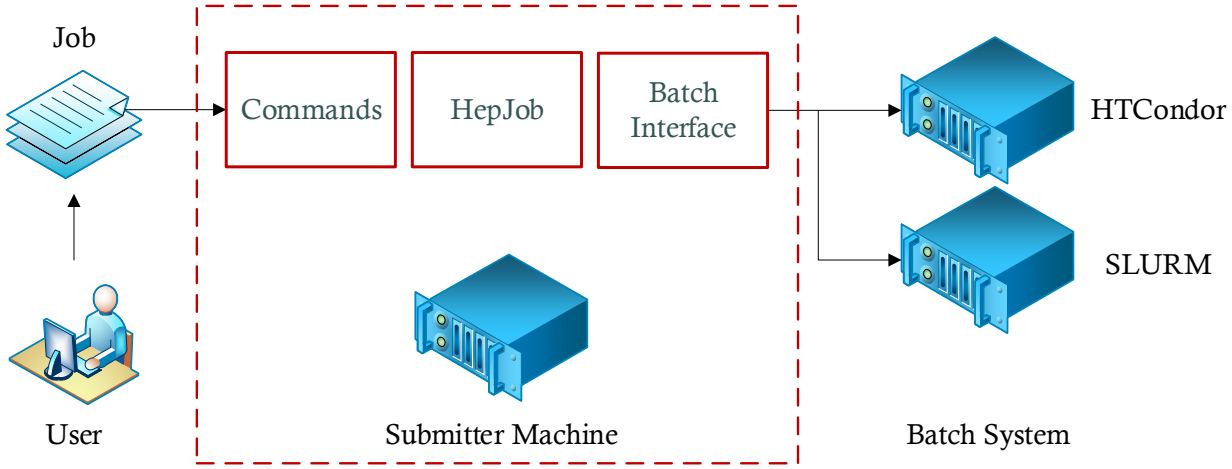
```
-bash-4.2$ cat /etc/condor/config.d/10-negotiator.conf
# Sets how often the condor_negotiator starts a negotiation cycle.
# It is defined in seconds and defaults to 60 (1 minute).
NEGOTIATOR_INTERVAL = 30

# An integer value that represents the minimum number of seconds that must pass
# before a new negotiation cycle may start. The default value is 20.
NEGOTIATOR_CYCLE_DELAY = 5

# A boolean value which defaults to false. When partitionable slots are enabled,
# and this parameter is true, the negotiator tries to pack as many jobs as
# possible on each machine before moving on to the next machine.
NEGOTIATOR_DEPTH_FIRST = True
```

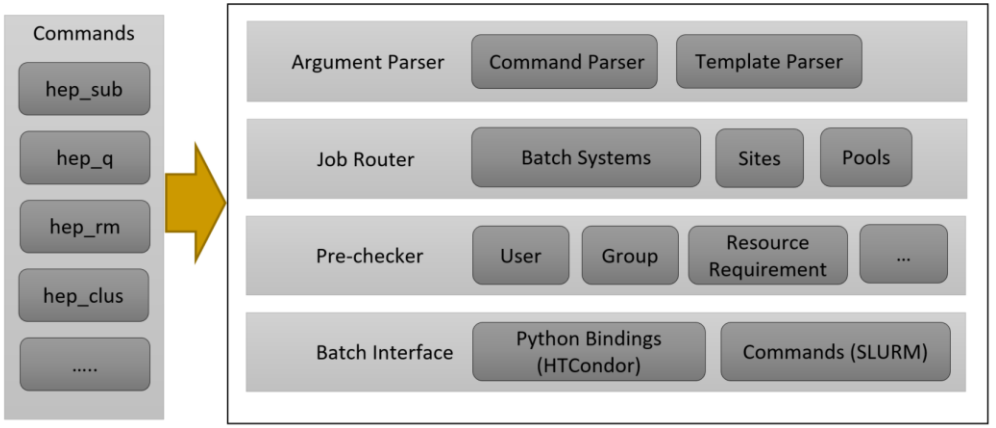


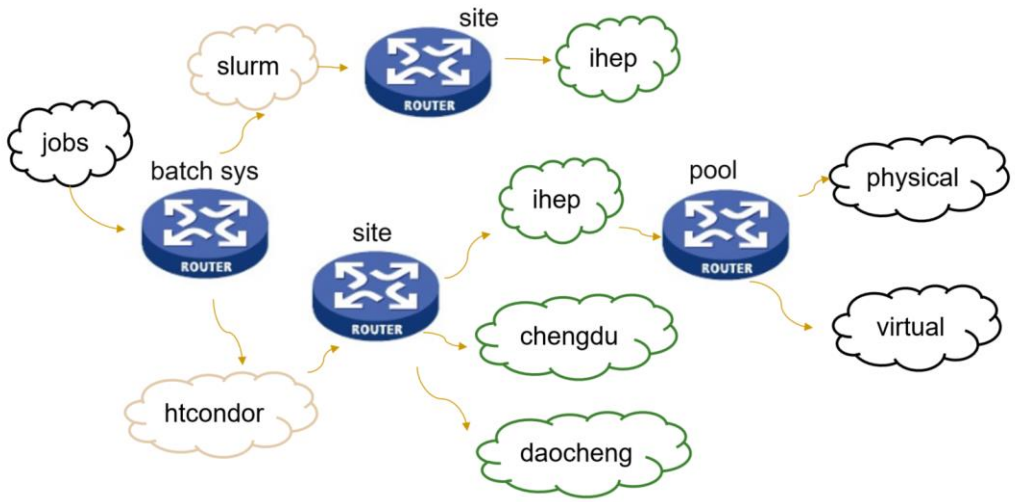
- Serve for GECAM
  - Storm event search
  - Job: single/multi cores

Job Entrance (HepJob)

- A submission frontend toolkit is developed and applied to unify the job interfaces
- A unified submission entrance
- Only simple commands should be learned



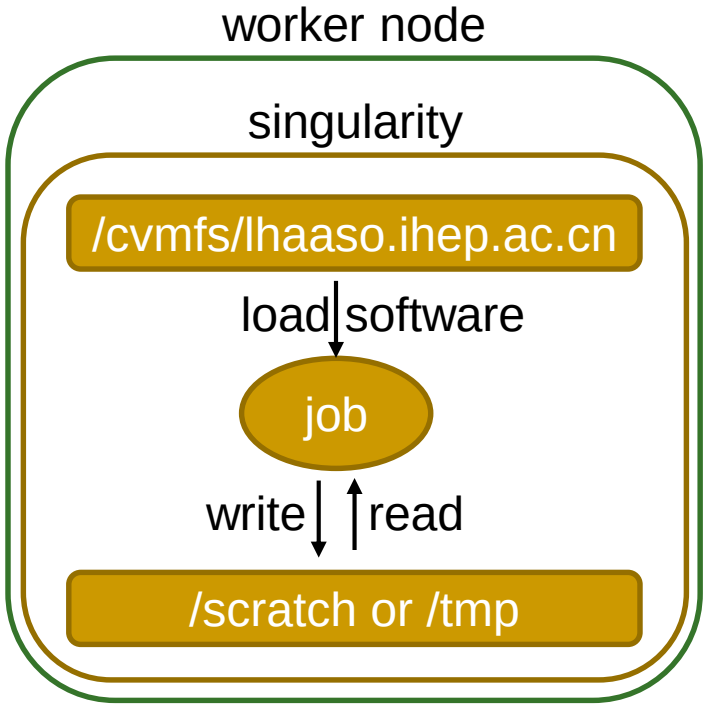




- The Job will be scheduled to the targeted:
  - site, cluster, pool

Job Environment (Singularity)

- Job environment in all solutions are based on singularity
- Operating System
  - Singularity images are published into /cvmfs/container…
  - Glidein job starts up singularity as the given image
- Software
  - Managed and served by CVMFS (recommended)
  - Transferred with job, as part of job input
- Temporary storage
  - The local scratch on the worker node
  - The global storage shared in the whole distributed infrastructure



Summary

- Several solutions based on HTCondor were made for the multiple scenarios at IHEP
  - Local High Throughput Cluster
  - Grid Sites
  - Distributed HTC Pool
  - Customized Clusters for Edge Sites
  - Real-time Computing for Satellite Project
- The next goal is to connect the separated sites/clusters together
  - The possible solution is HTCondor glidein

Thanks
Q&A