Reaching new scales with the CMS HTCondor Global Pool

June, 2017

Diego Dávila
on behalf of the Submission Infrastructure team of the CMS experiment.

Presentation layout.

1. Introduction to the CMS experiment at CERN
2. Why HTCondor?
3. Our different pools
4. A general view of the Global pool setup
5. Latest scalability limit we faced
6. The solution
7. Scalability test (past and future)
8. Other Challenges

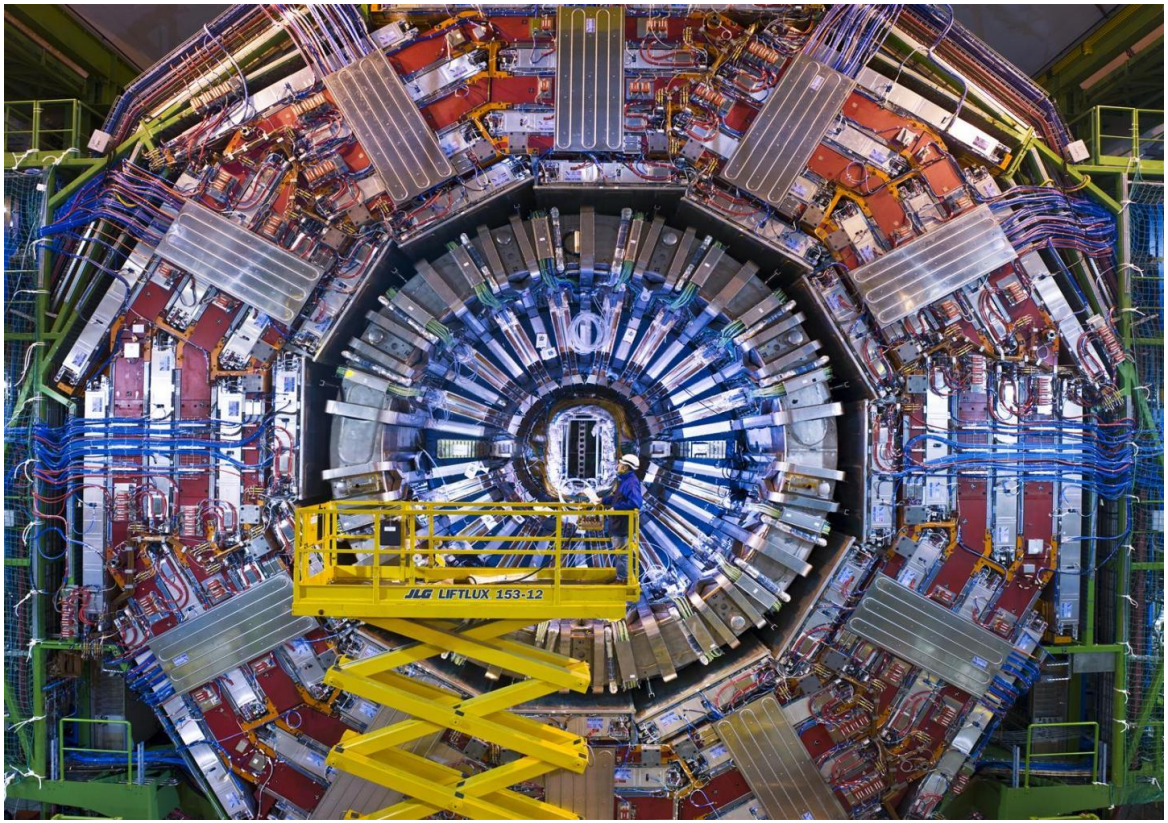
Introduction to the CMS experiment at CERN

- CMS - Compact Muon Solenoid
- One of the main experiments in the LHC (Large Hadron Collider)
- Collaboration of more than 4k people

Introduction to the CMS experiment at CERN

2885	995	279	198	45
PHYSICISTS (922 STUDENTS)	ENGINEERS	TECHNICIANS	INSTITUTES	COUNTRIES & REGIONS

Introduction to the CMS experiment at CERN

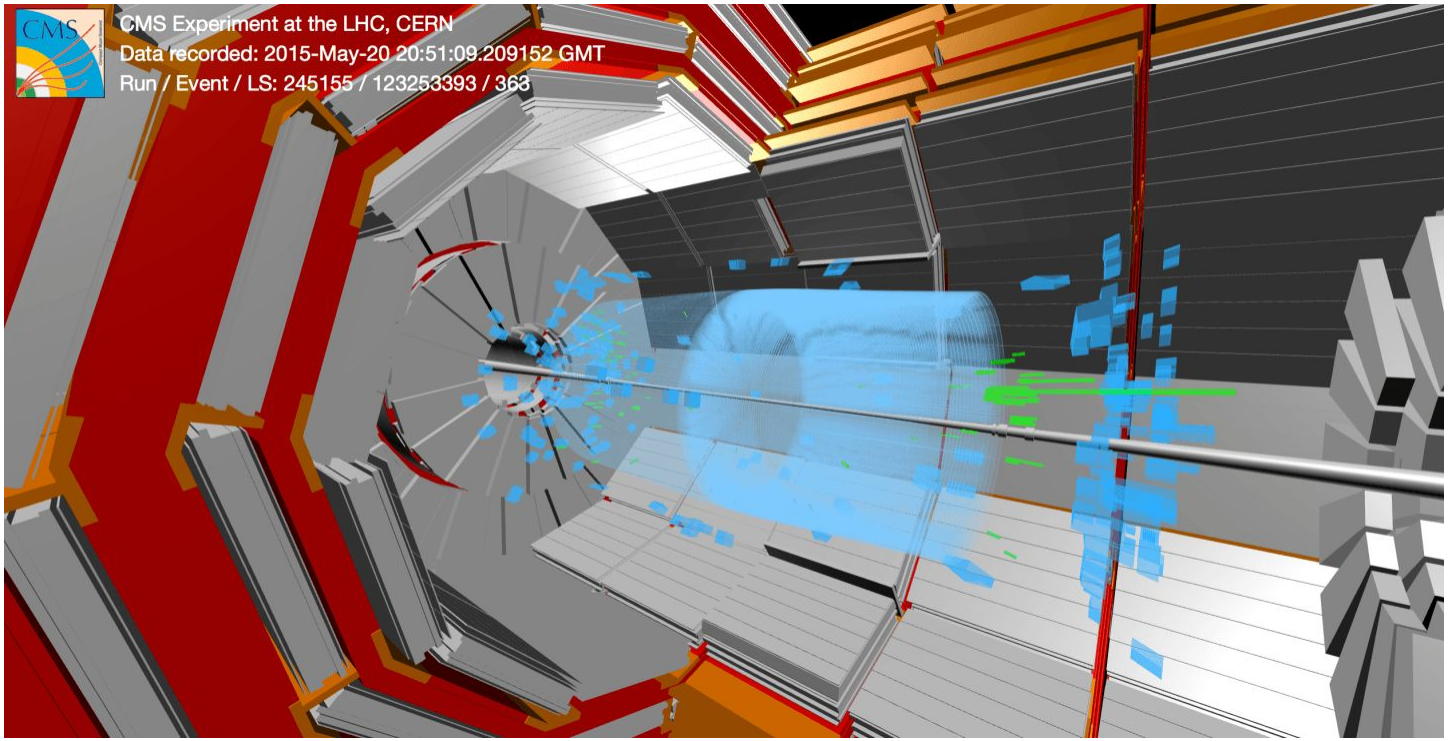


- 15m diameter
- 21m long
- 14k Tons

“Although it is the size of a cathedral, it contains detectors as precise as a Swiss watch”.
Lucas Taylor

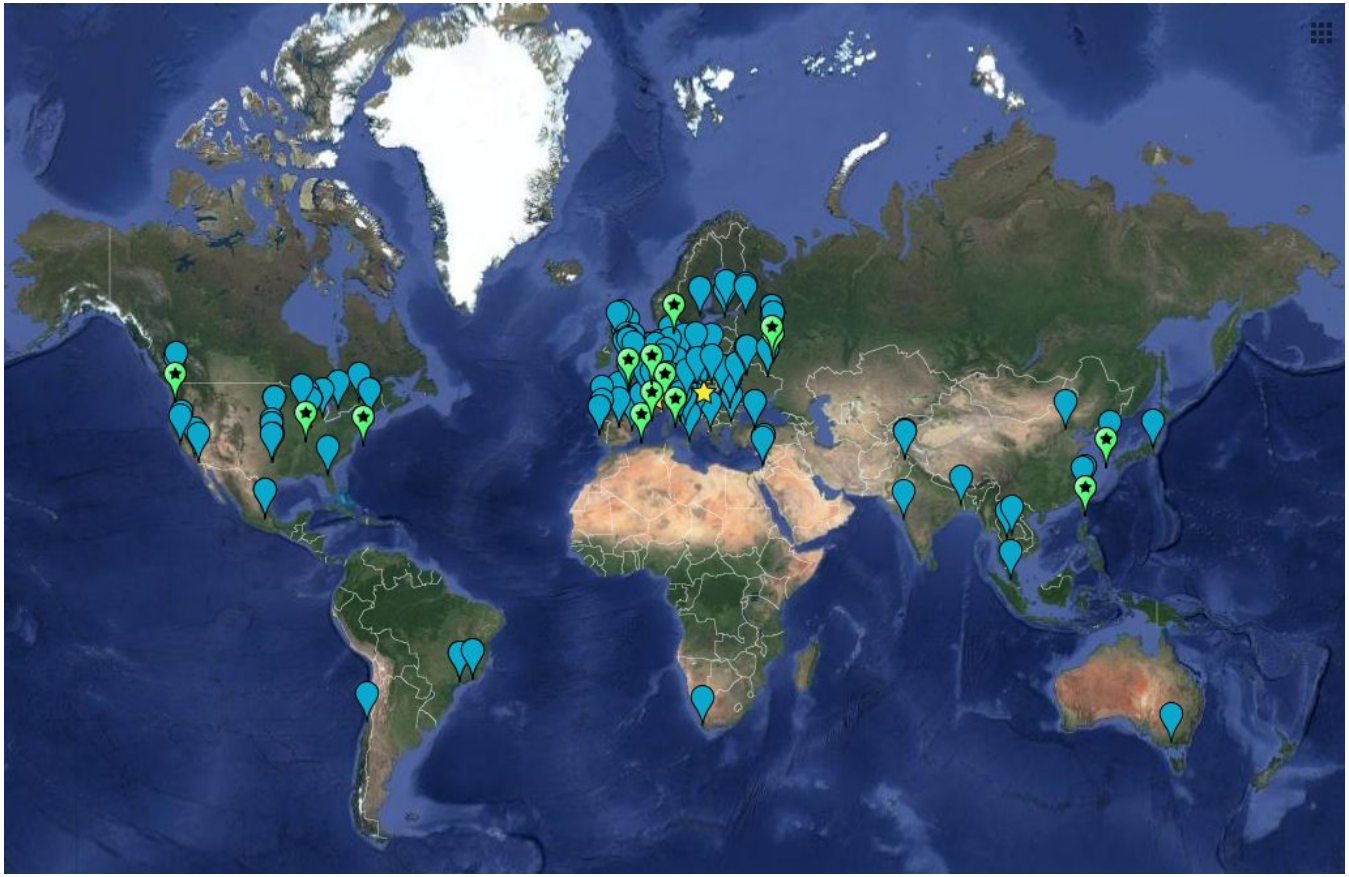
Introduction to the CMS experiment at CERN

- 600 Millions of collisions per second
- After the 2-level trigger system ~1K event per second
- 2MB need to store an event.
- Produces a stream of ~2GB per second
- Tens of petabytes per year to be analyzed



Worldwide LHC Computing Grid

- Hundreds of thousand CPU cores distributed in more than 100 sites worldwide
- Categorized in 4 differents types of tiers



Why HTCondor?

millions of jobs + **hundreds of thousands of CPU cores** = **high throughput computing**

HTCondor
High Throughput Computing

“A scalable and flexible system that allow us to introduce all kind of policies to keep our users happy”

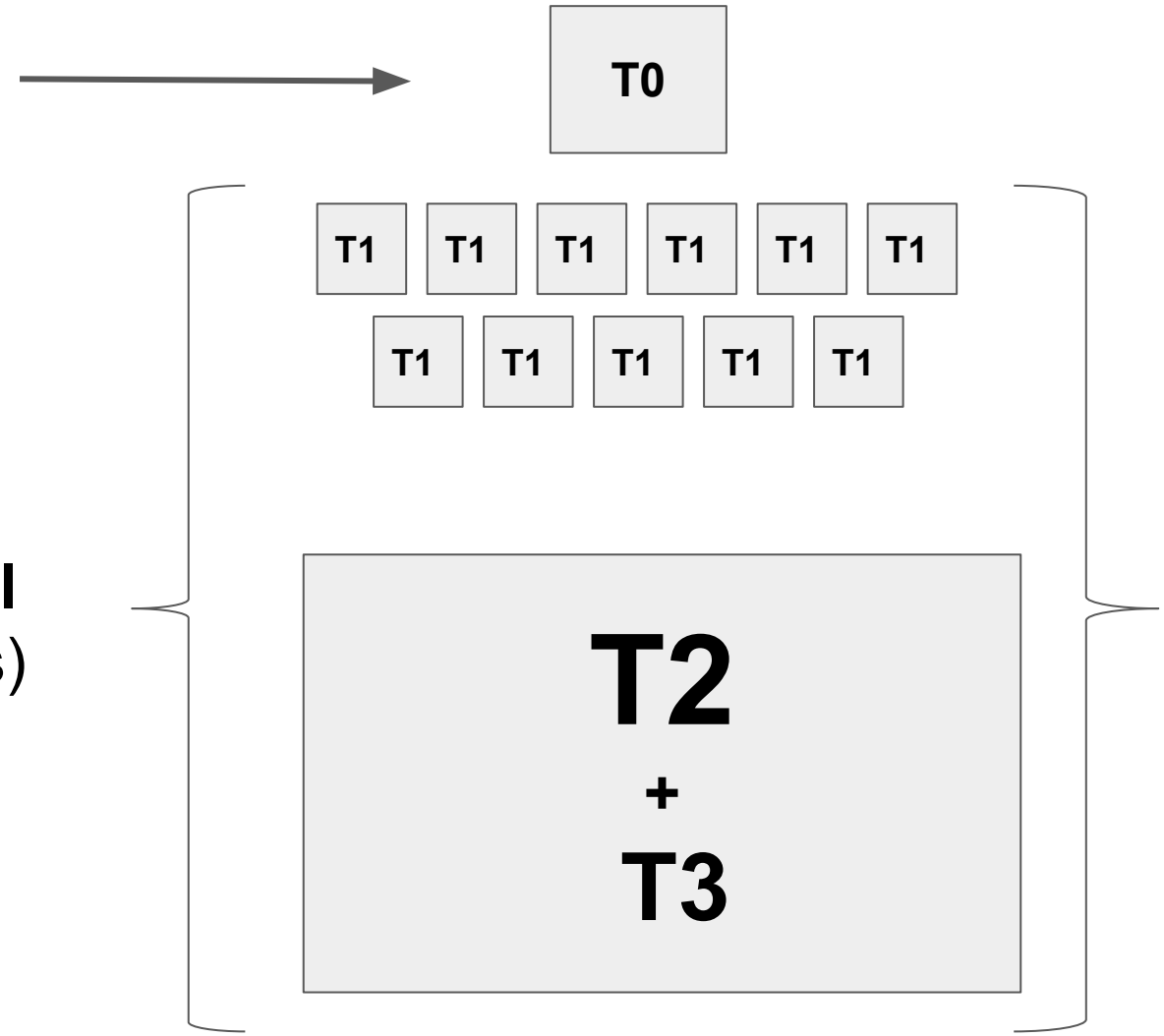
Our different HTCondor pools

- Testing pools
  - **ITB-DEV**
    - Playground
    - Used to “test before testing”
  - **ITB**
    - Replica of the global pool
    - Test new features before going to production

- Production pools
  - **Tier0**
  - **Global pool**

Production pools

Tier 0 pool
(7 schedds)

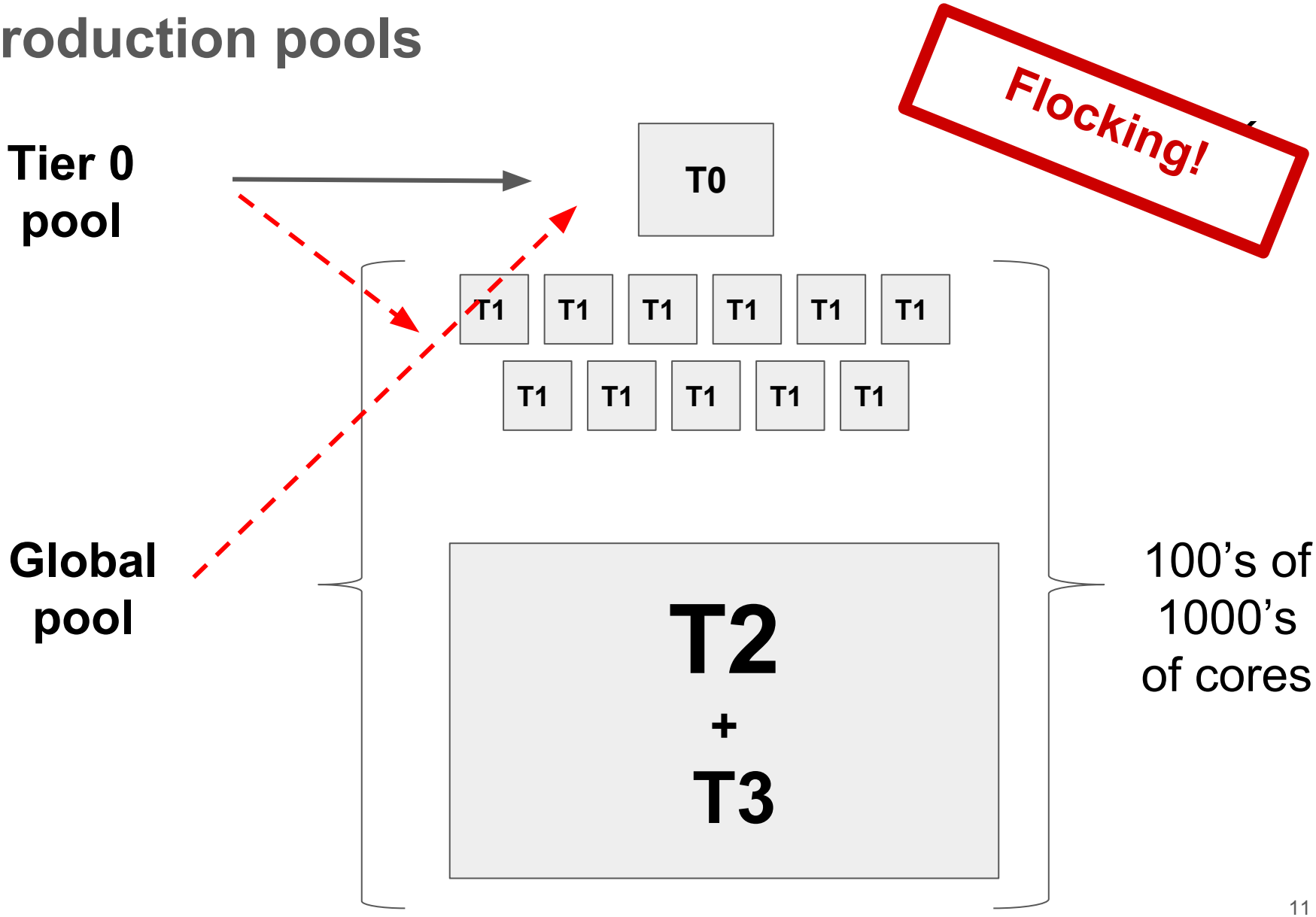


30K
cores

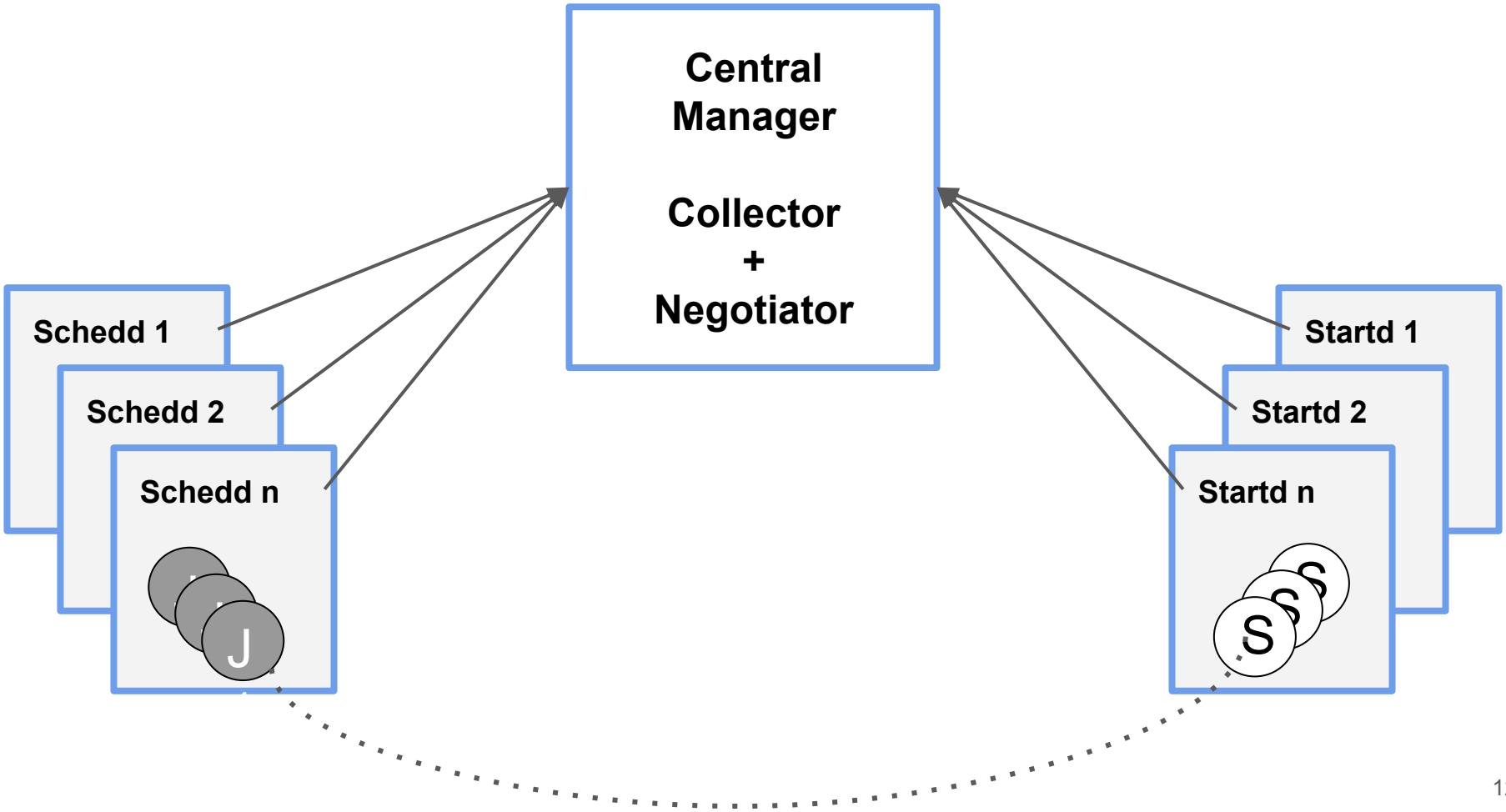
Global pool
(37 schedds)

100's of
1000's
of cores

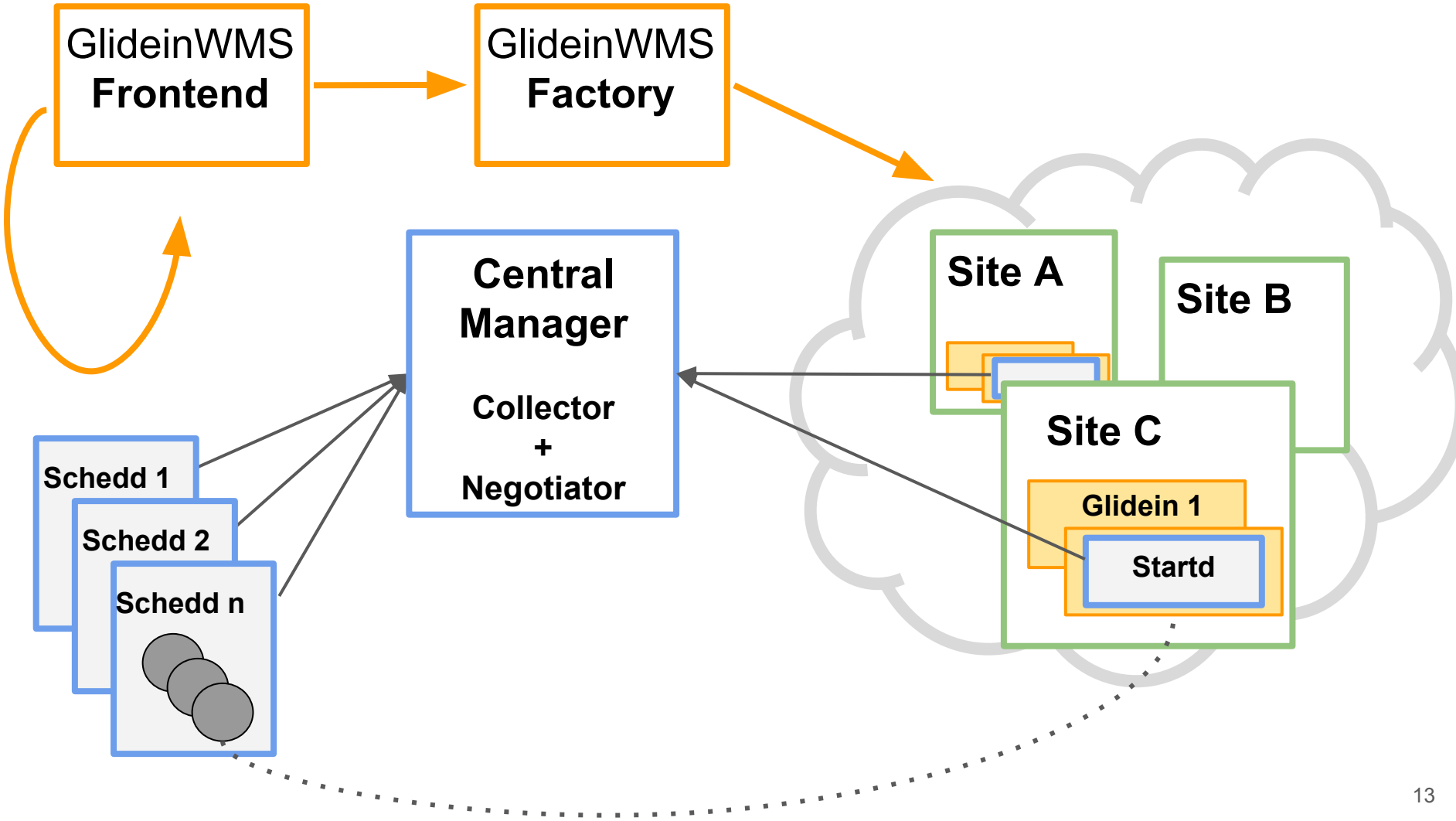
Production pools



The general view of the Global pool setup

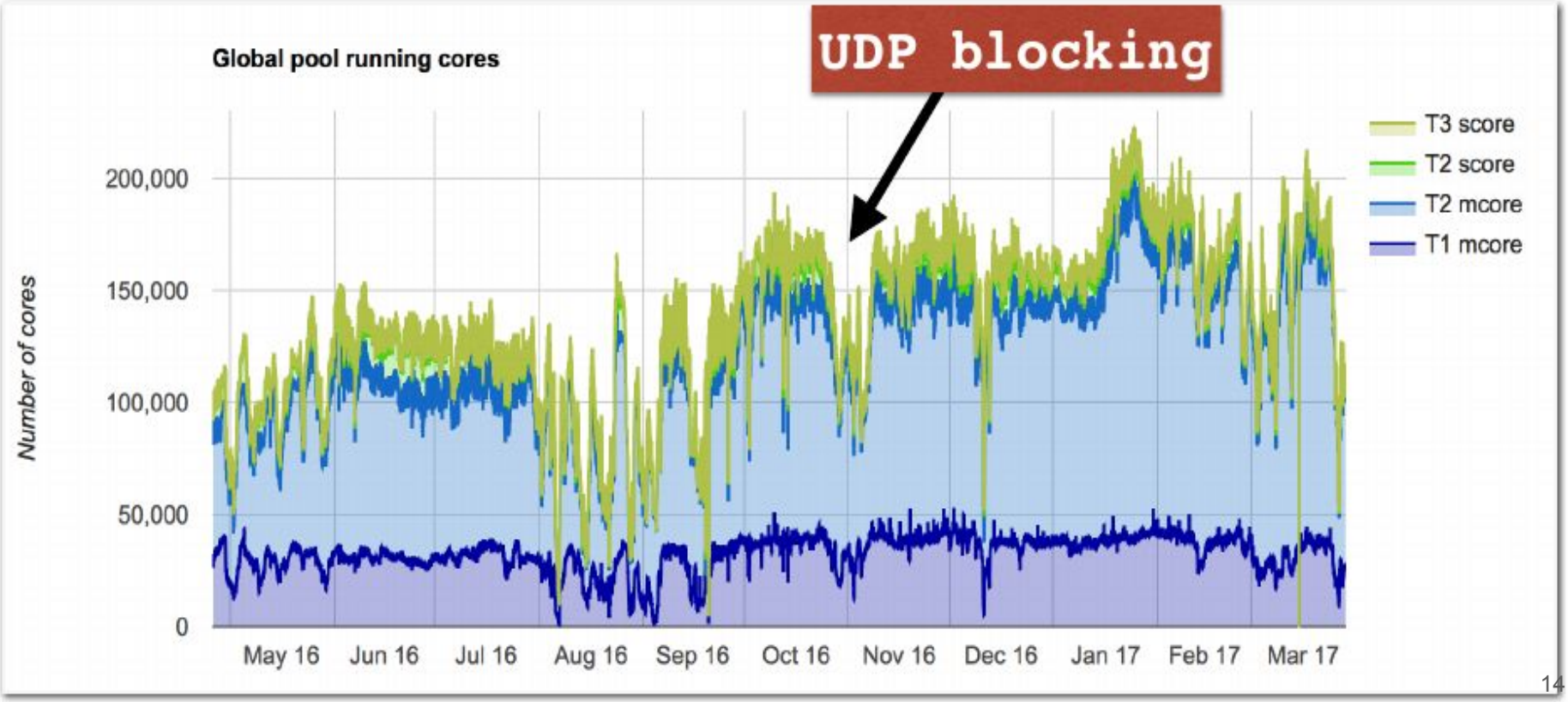


The general view of the Global pool setup



The latest scalability limit we faced

- Going beyond 150k cores
- Collector started to drop updates/queries (including Negotiator ones)



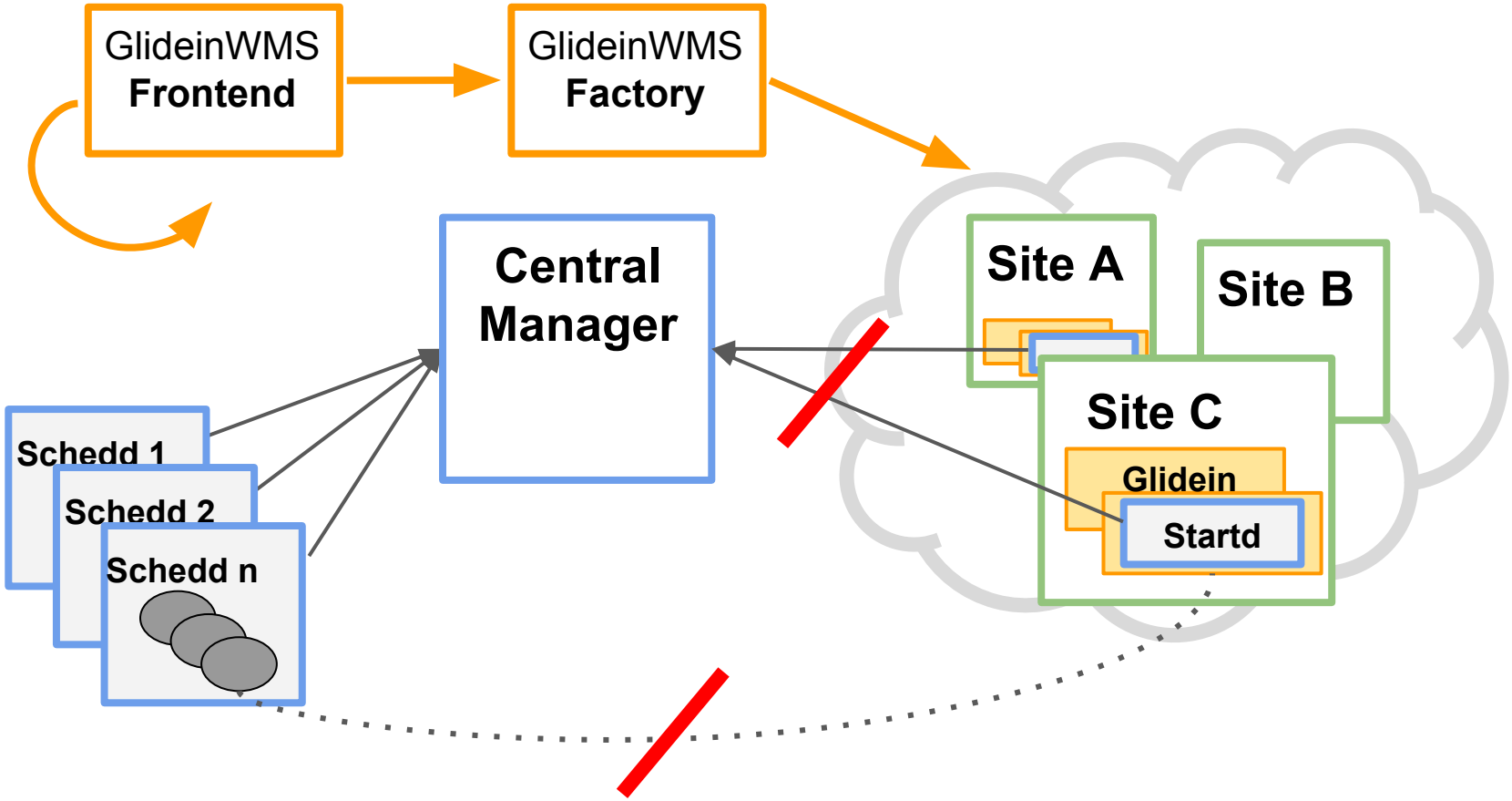
The latest scalability limit we faced

How do you reach that point?

1. A very large pool
2. Lack of memory in the Central Manager
3. Many expensive queries
4. The way Condor process queries
   a. Collector forks child-collectors to process/answer queries
   b. When the limit of child-collectors is hit, the parent-collector will process queries
   c. While the parent-collector is busy, new updates/queries are buffered
   d. When buffer gets full, new updates/queries are dropped.

Which is the effect of dropping queries?

- Dropped queries from the negotiator will prevent new “matchmaking”
- Missing updates in the collector will prevent the pool to add new resources

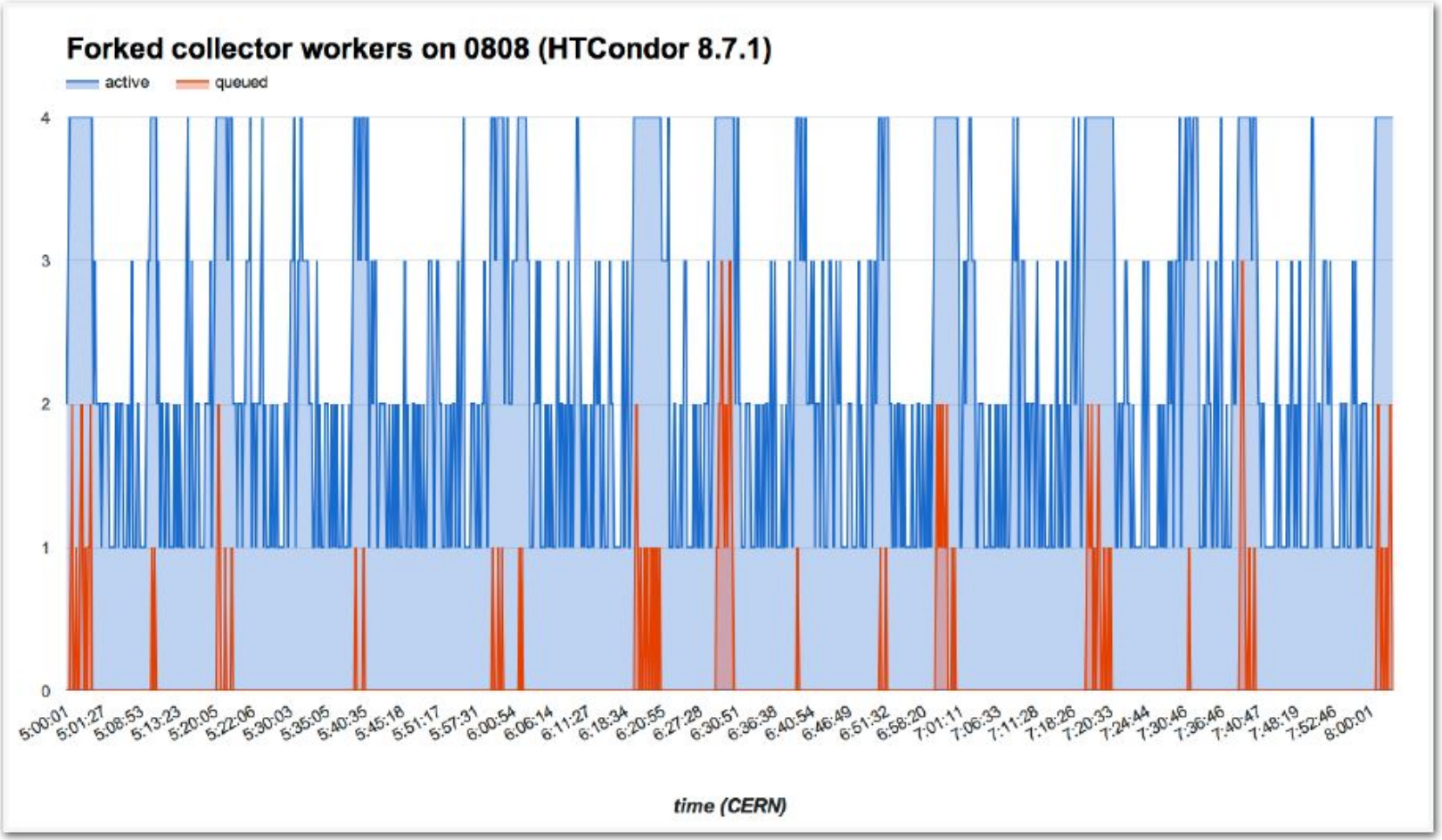


The solution

1. A very large pool
2. Lack of memory in the Central Manager
   a. Move the Central Manager to a machine with 20% more memory
3. Many expensive queries
   a. Adding projection list to the queries
   b. Pointing monitoring to the backup Central Manager
4. The way Condor process queries (8.7.1)
   a. The parent process never process expensive queries
   b. Queries are queued when no more child processes available
   c. Negotiator queries are prioritized

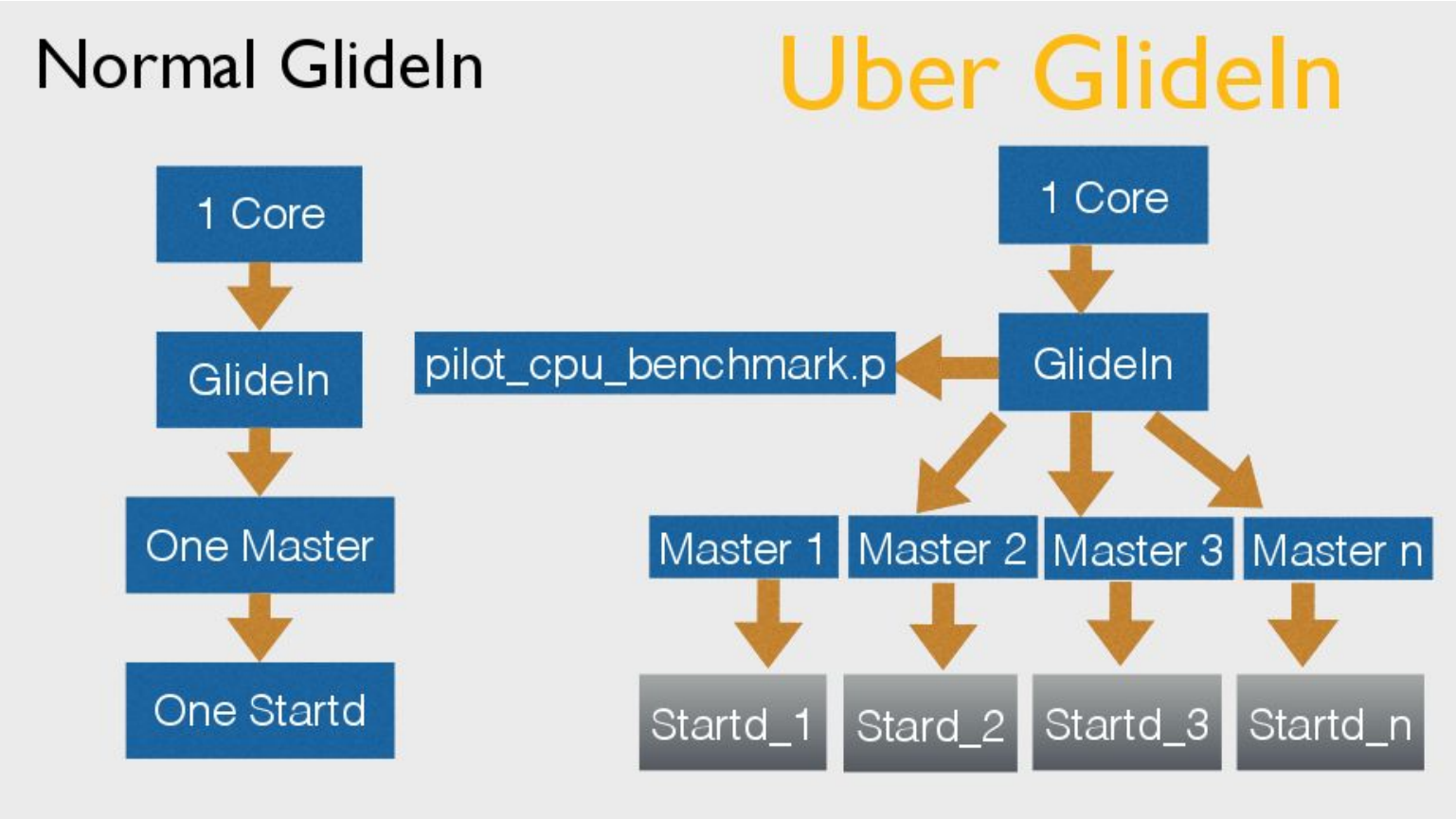
The solution

Queries being processed/queued using HTCondor 8.7.1



Scalability tests

How to emulate a VERY LARGE pool ?



https://research.cs.wisc.edu/htcondor/HTCondorWeek2015/presentations/FajardoE.pdf

Previous scale test

- Conducted by the OSG in 2014
- Objective: to reach 200k cores
- Used 20k physical cores to simulate 200k
- Started observing problems around 150k cores

Solution:

- Moving the CCB to a separated machine
- Increasing the UDP buffer size
- Reducing the frequency of the startd reporting to collector
- Improvements on HTCondor 8.3

https://research.cs.wisc.edu/htcondor/HTCondorWeek2015/presentations/FajardoE.pdf

Next scalability tests

Objective: Test the scalability of an HTCondor pool with 500k cores in a safe environment.

- Using a factor of 32, we could emulate a pool of 500k startds with only ~15k physical cores.
- This test is planned to be carried out during August, commonly a period with low usage.
- The central manager, to be tested, will be deployed within a bigger machine provided by CERN IT (65% more RAM than the current one)

“We are looking forward to see how much can we scale with this new version of HTCondor”

Other challenges

- Increasing the efficiency in the resource utilization
  - DEPTH-WISE Filling
- I/O Scheduling
  - Limit the number of high I/O jobs running PER SITE
- IPV6 support
  - Last week we ran the first test job completely over ipv6 infrastructure

Thank you very much to the HTCondor developers for all the support and collaboration!