HammerCloud as a commissioning tool

Possibilities for the ESCAPE communities

Jaroslava Schovancová, Aristeidis Fkiaras
(CERN)

Further read: a document

HammerCloud at a glance

HammerCloud is a framework to

- commission,
- run continuous tests or on-demand large-scale stress tests, and
- benchmark

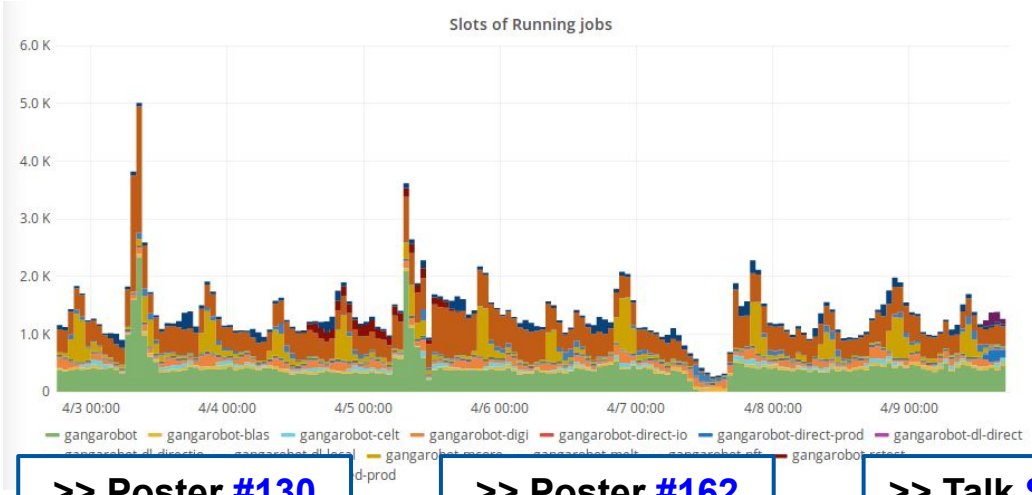
computing resources and *components of various distributed systems* with *realistic full-chain* experiment workflows

HammerCloud at a glance II

- Functional and stress tests of WLCG resources: ATLAS, CMS; Batch
  - **Functional:** steady flow of test jobs
  - **Stress:** on demand tests, configure load intensity
- Part of automation suite of the Experiments
- Testing the **full chain** of an Experiment job
  - Same environment as any “real” analysis/production jobs
- Utilization
  - ATLAS: 80k jobs/day, ~30 tests/day
  - CMS: 39k jobs/day, 36 tests/day
  - Batch: 150-750 jobs/day, ~1-2 tests/day
- About HC: CHEP2018

HammerCloud activities

ATLAS: functional testing & auto-exclusion of resources; ESblacklist; commissioning of new resources; commissioning of new components of distributed computing systems (Pilot, Rucio, new data access protocols, …); FT of services (ObjectStore, Dynafed testing)



CMS: functional testing; commissioning of new resources; commissioning of new components of distributed computing systems

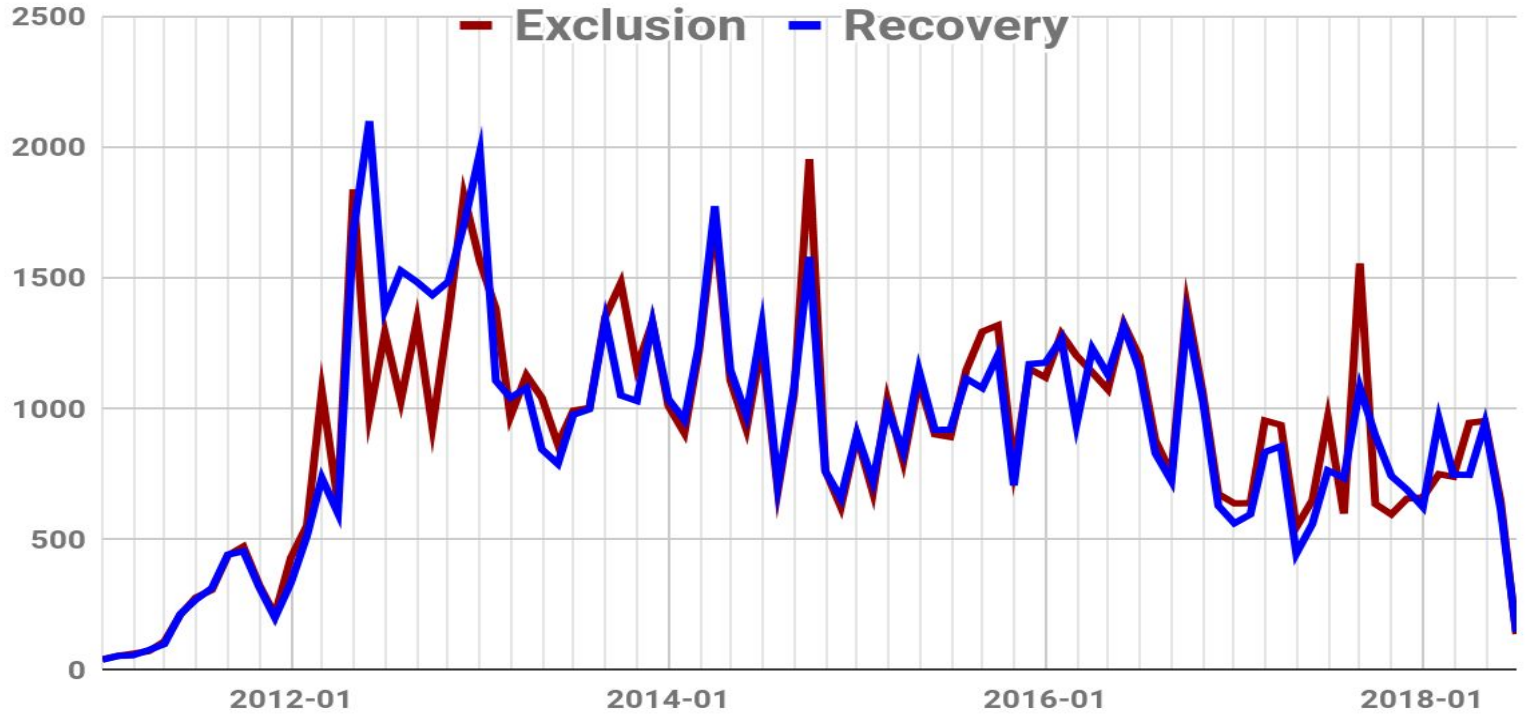
Batch: BEER, external cloud, CI/CD, containers usability, ...

>> Poster #130

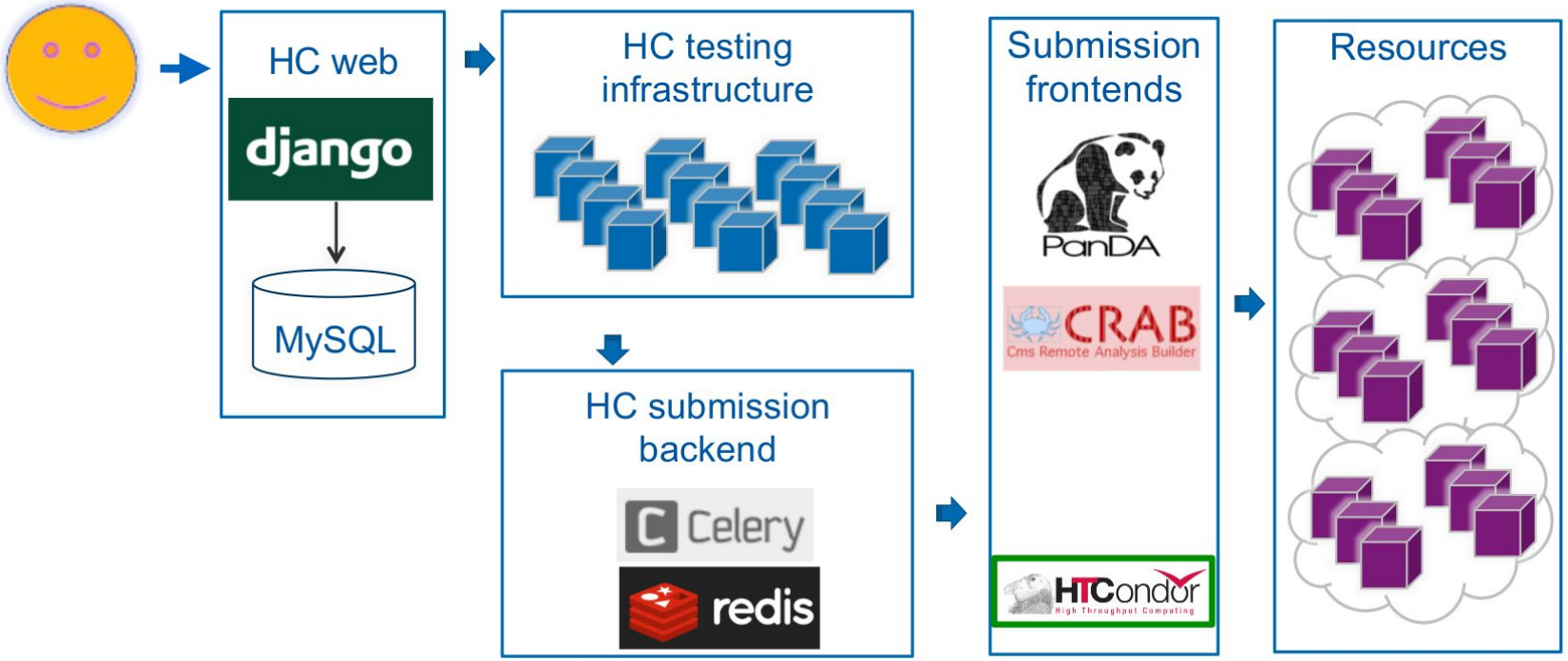
>> Poster #162

>> Talk Sharing server nodes for storage and compute

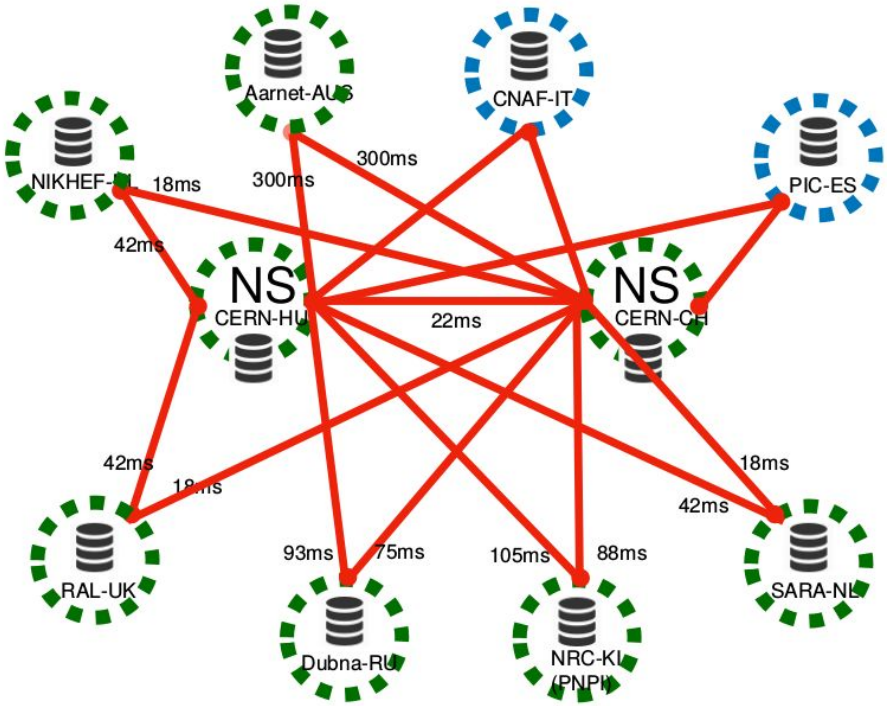
ATLAS HammerCloud auto-exclusion



HammerCloud from far away



Data Lake Prototype



- Goal: testbed to test and demonstrate some of the ideas
- Deployed a Distributed Storage prototype, based on EOS
  - distributed storage
  - network links: latency, bandwidth
  - storage media: disk/cache/tape
  - evolving data access protocols: driven by the changes in networks
  - evolving inter-storage communication

The core metric: event throughput

- Compute side of things ⇒ boils down to the **event throughput at the same cost**

⇒ Are we able to support the same or even better event throughput at the same cost with the evolving storage configuration?

- Easier said than done!
  - Which events? Which SW? How much I/O? How much memory? ...
  - How to measure job performance? Storage performance?
  - How to benchmark?
  - What to take into account for the storage configuration?
  - Topology of resources? its transparency?
  - (Co-)location of data vs. compute resources?
  - Types of storage media vs. access policies?
  - Direct vs. remote access to data?
  - How to evolve tools to support the core mission

Measurements

- Methodology, how to measure and benchmark
- What to measure: event throughput
  - I/O rate
  - Stage-in / Stage-out time
  - SW init time
  - Time spent in the event loop
- Production and Analysis workflows
- Core count preferences: MCORE (production) vs. SCORE (analysis)
- Local vs. remote data access

Workflows types - ATLAS

- G4 simulation
  - CPU intensive, not so much RAM demanding, not much I/O intensive
  - ttbar full simul, reference workflow to compare HS06
- Digi+reco
  - some I/O (not that much IOwaits for jobs), RAM-demanding, sensitive to latency
  - Event mixing, digitization, trigger, trigger reconstruction
  - 50 GB in
- Production derivation
  - More I/O intensive
  - Skim, slim, …
  - 5 GB in
- Analysis - focusing on analysis derivation

Data Lake and HammerCloud

- ✓ We integrated the Data Lake Prototype with HammerCloud
- We can test real workflows and data access patterns of ATLAS and CMS

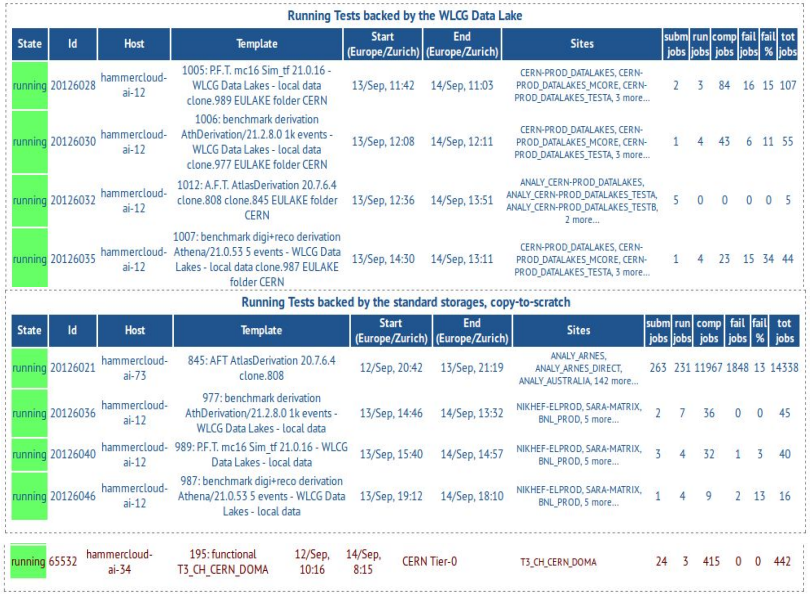
Initial focus on ATLAS

(Data is copied from storage to WN)

4 test scenarios, stage-in from

1. Base: Local access (no data lake)
2. A: DLP, data @CERN, WN @CERN
3. B: DLP, data NOT @CERN, WN @CERN
4. C: DLP, 4+2 stripes, WN @CERN

PoC with a CMS “1-chain job” running.



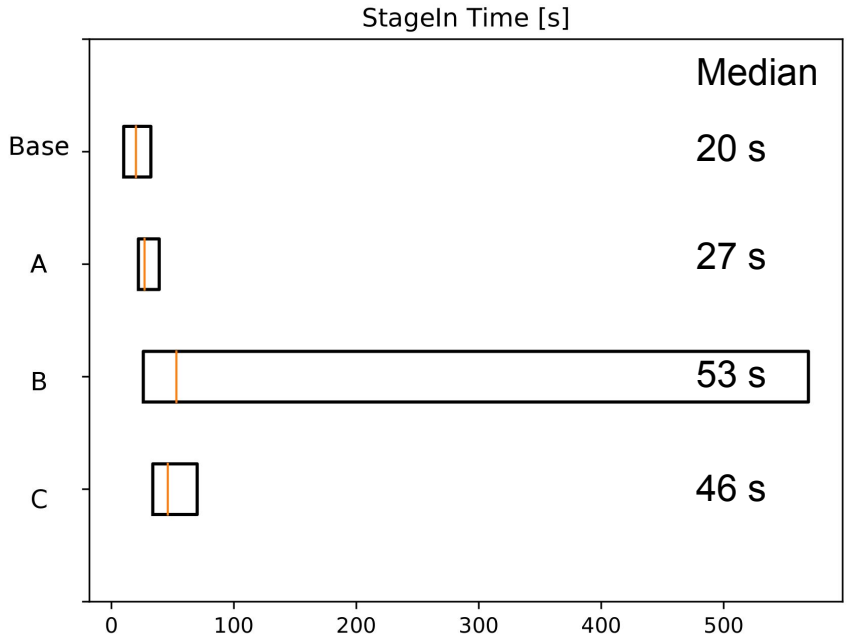
Running Tests backed by the WLCG Data Lake

State	Id	Host	Template	Start (Europe/Zurich)	End (Europe/Zurich)	Sites	subm jobs	run jobs	comp jobs	fail jobs	fail %	tot jobs
running	20126028	hammercloud-ai-12	1005: P.F.T. mc16 Sim_tf 21.0.16 - WLCG Data Lakes - local data clone.989 EULAKE folder CERN	13/Sep, 11:42	14/Sep, 11:03	CERN-PROD_DATALAKES, CERN-PROD_DATALAKES_MCORE, CERN-PROD_DATALAKES_TESTA, 3 more...	2	3	84	16	15	107
running	20126030	hammercloud-ai-12	1006: benchmark derivation AthDerivation/21.2.8.0 1k events - WLCG Data Lakes - local data clone.977 EULAKE folder CERN	13/Sep, 12:08	14/Sep, 12:11	CERN-PROD_DATALAKES, CERN-PROD_DATALAKES_MCORE, CERN-PROD_DATALAKES_TESTA, 3 more...	1	4	43	6	11	55
running	20126032	hammercloud-ai-12	1012: A.F.T. AtlasDerivation 20.7.6.4 clone.808 clone.845 EULAKE folder CERN	13/Sep, 12:36	14/Sep, 13:51	ANALY_CERN-PROD_DATALAKES, ANALY_CERN-PROD_DATALAKES_TESTA, ANALY_CERN-PROD_DATALAKES_TESTB, 2 more...	5	0	0	0	0	5
running	20126035	hammercloud-ai-12	1007: benchmark digi+reco derivation Athena/21.0.53 5 events - WLCG Data Lakes - local data clone.987 EULAKE folder CERN	13/Sep, 14:30	14/Sep, 13:11	CERN-PROD_DATALAKES, CERN-PROD_DATALAKES_MCORE, CERN-PROD_DATALAKES_TESTA, 3 more...	1	4	23	15	34	44

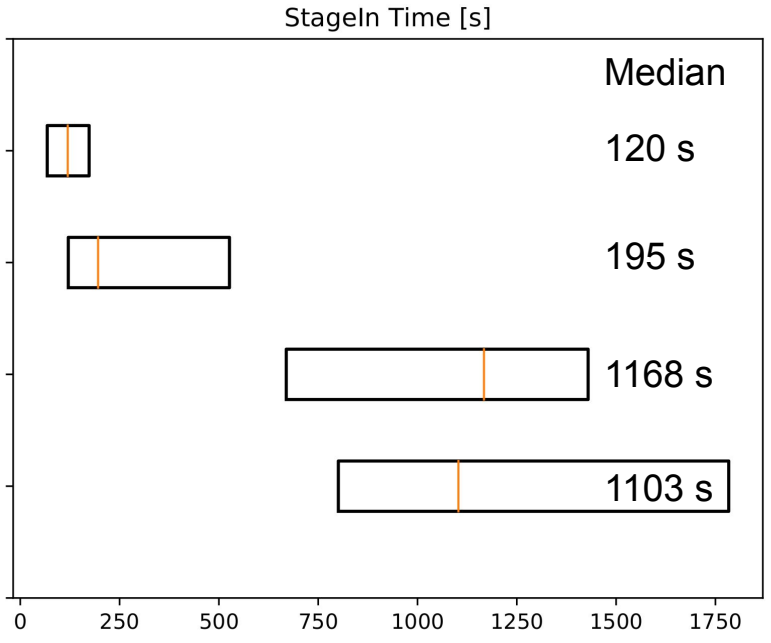
Running Tests backed by the standard storages, copy-to-scratch

State	Id	Host	Template	Start (Europe/Zurich)	End (Europe/Zurich)	Sites	subm jobs	run jobs	comp jobs	fail jobs	fail %	tot jobs
running	20126021	hammercloud-ai-73	845: AFT AtlasDerivation 20.7.6.4 clone.808	12/Sep, 20:42	13/Sep, 21:19	ANALY_ARNES, ANALY_ARNES_DIRECT, ANALY_AUSTRALIA, 142 more...	263	231	11967	1848	13	14338
running	20126036	hammercloud-ai-12	977: benchmark derivation AthDerivation/21.2.8.0 1k events - WLCG Data Lakes - local data	13/Sep, 14:46	14/Sep, 13:32	NIKHEF-ELPROD, SARA-MATRIX, BNL_PROD, 5 more...	2	7	36	0	0	45
running	20126040	hammercloud-ai-12	989: P.F.T. mc16 Sim_tf 21.0.16 - WLCG Data Lakes - local data	13/Sep, 15:40	14/Sep, 14:57	NIKHEF-ELPROD, SARA-MATRIX, BNL_PROD, 5 more...	3	4	32	1	3	40
running	20126046	hammercloud-ai-12	987: benchmark digi+reco derivation Athena/21.0.53 5 events - WLCG Data Lakes - local data	13/Sep, 19:12	14/Sep, 18:10	NIKHEF-ELPROD, SARA-MATRIX, BNL_PROD, 5 more...	1	4	9	2	13	16
running	65532	hammercloud-ai-34	195: functional T3_CH_CERN_DOMA	12/Sep, 10:16	14/Sep, 8:15	CERN Tier-0	T3_CH_CERN_DOMA	24	3	415	0	0	442

Data Lake, Stage-in Time



Low I/O intensity workflow



High I/O intensity workflow

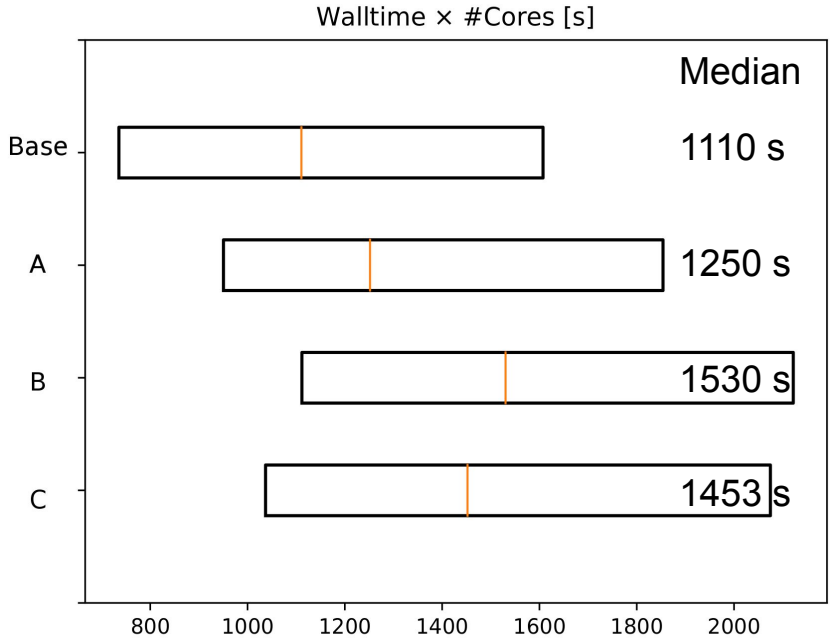
Base:
No data lake

A: Data lake;
data @CERN
WN @CERN

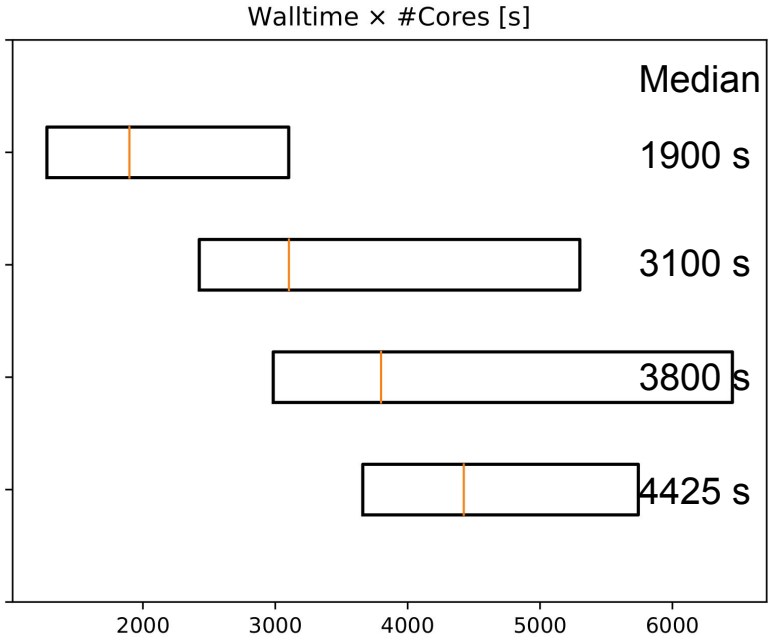
B: Data lake;
data not @CERN
WN @CERN

C: Data lake;
data 4+2,
WN @CERN

Data Lake, WallTime x cores



Low I/O intensity workflow



High I/O intensity workflow

Base:
No data lake

A: Data lake;
data @CERN
WN @CERN

B: Data lake;
data not @CERN
WN @CERN

C: Data lake;
data 4+2,
WN @CERN

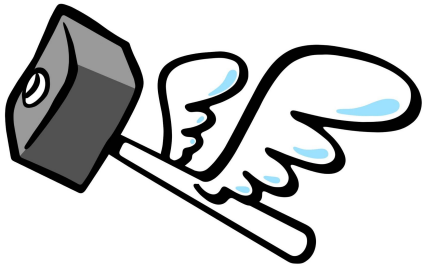
Full-chain experiment job

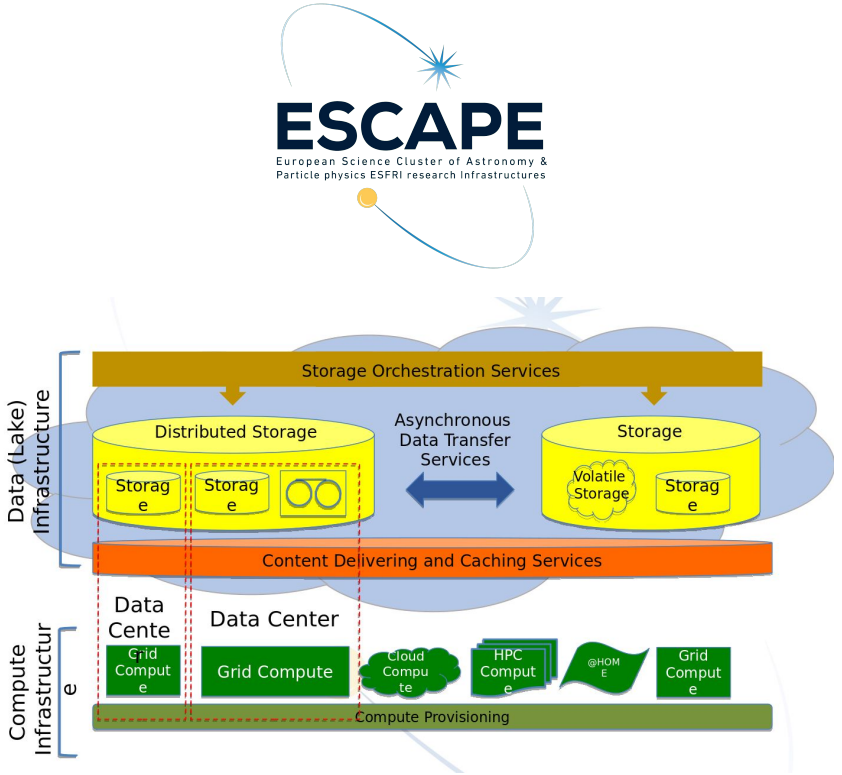
- Full-chain: perform all activities as the standard jobs of the Experiment
  - “Standard candle” jobs, controlled environment
  - Commission resources, components of distributed computing systems
- Examples:
  - WLCG Data Lake commissioning & prototyping of ideas CHEP2018, IEEE eScience 2018
  - CRAB3 commissioning in 2014
  - ATLAS Pilot commissioning; ATLAS SW installation system; Rucio mover
  - CERN BEER (Batch on Extra EOS Resources, CHEP2018)
  - Sim@P1: ATLAS HLT farm (infrastructure ramp up, CHEP2016)
  - Integration of Cloud Computing resources
  - Commissioning of new (or non-standard) resources configuration
- Benefits: controlled environment, “standard candle” jobs, interact with infrastructure in the same way as any other experiment job

A service commissioning

- We have a ***service endpoint*** and we issue ***a generic request*** to communicate with that service
    - in a controlled environment, at reasonable/desired scale
- Examples: ATLAS ObjectStore, ATLAS Dynafed tests
- Benefits:
    - Controlled environment
    - Can commission a new endpoint “on demand”
    - Can commission a service
    - Can monitor & health-check the service endpoint

HammerCloud & WP2 DataLake





Submission backends

- Workload Management Systems (WMS):
  - PanDA
  - CRAB3
  - Dirac

- Batch Systems/Computing Elements (CE):
  - HTCondor
  - HTCondor-CE
  - ARC-CE / other CEs
  - Other Batch Systems

- Already available in HC
- HC will require some development to make available

Building blocks

- SW distribution
  - CVMFS
  - Local installation on the Compute cluster
  - Package in tarball/RPM
  - Containers

- Already available in HC
- HC will require some development to make available

- (Distributed) Data Management
  - WMS/Batch instrumented to perform DDM tasks
  - Locally available data
  - DDM client

- What about licencing policies?

Job definition & configuration

- If run at CERN, via HTCondor-CE, we need a bash script to do
  - setup SW, setup clients (e.g. DDM)
  - stage-in (or access the input data),
  - run the payload,
  - stage-out (or upload the output data).
- A job can be defined with a bash script, a container, mix of both,...
- We may need a proxy registered in proper VO

- If run outside CERN, what WMS/CE/Batch System interaction do we need?

HammerCloud as a commissioning tool

- Commission resources and components of distributed computing systems
  - With full-chain experiment jobs
  - Commission service endpoints
  - Automation of computing operations
- Submit to WMS PanDA & CRAB3, and HTCondor(-CE)
  - What other submission backends do we need?
- Payload
  - Same as in a full-chain job
  - Request to a service endpoint
  - A “bash script” or a “container” with some payload activity
  - Something else?
- Volunteers? :)

Jaroslava Schovancová, Aristeidis Fkiaras (CERN)


CERN