System Efficiency Improvement for Distribute Cloud

Felix Lee, Eric Yen

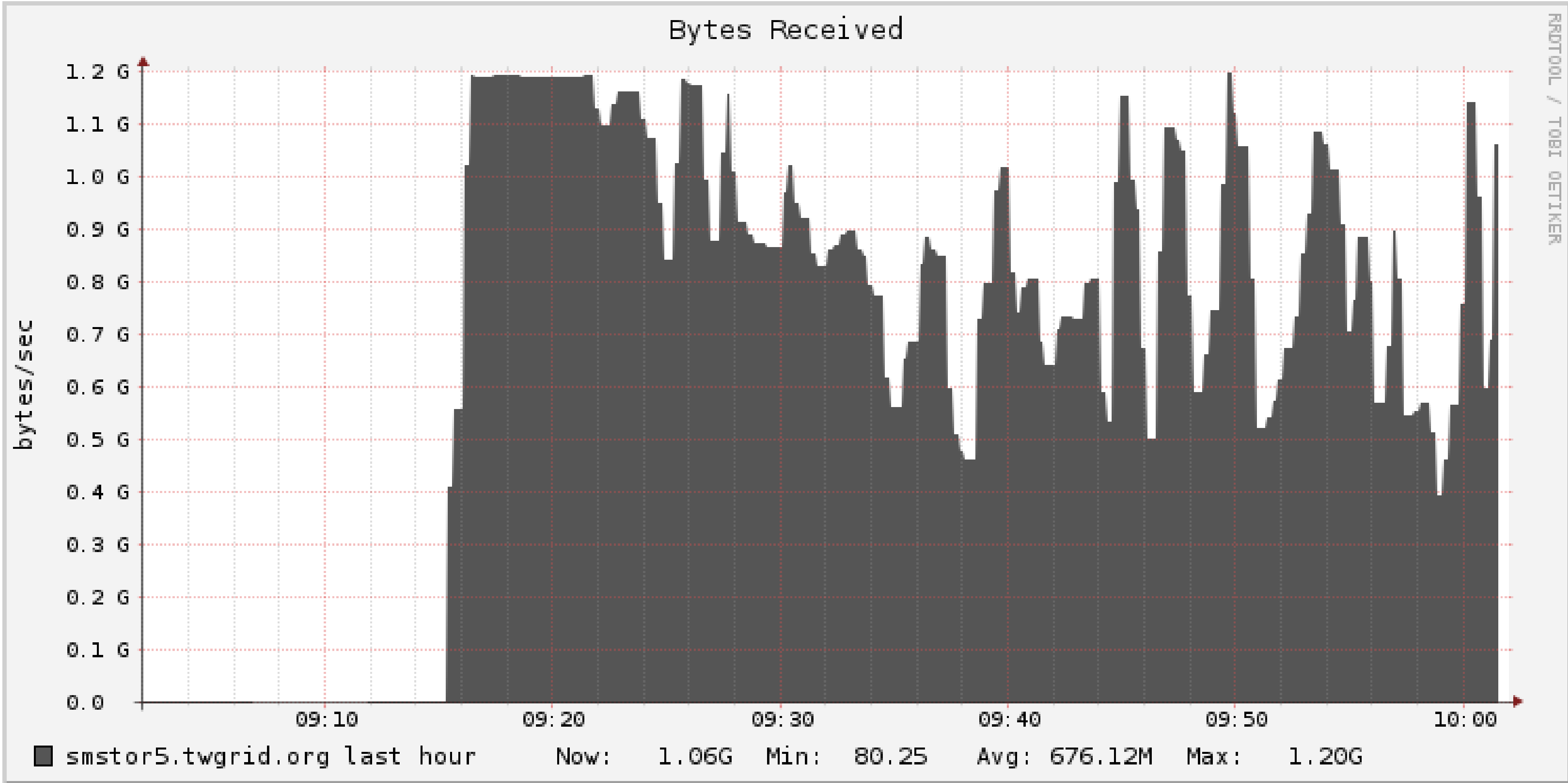
ASGC is Supporting Scientific Computing by Leveraging Distributed Computing Infrastructure and Technology, as well as International Collaboration

- Application Workflow Optimization: Support Big Data and Long-Tail Scientific Applications More Efficiently
- Improve System Efficiency Optimization: Power, Thermal, Operation
- Enhance the Implementation of Computing Models by User/Application Experiences and Innovative Technologies
- Automatic Monitoring and Control
- Employing Machine Learning
- Broaden Application Domains and User Communities
- Extend the Local and Regional Research Infrastructure by Federating Resources

System Efficiency Improvement is a very Essential Internal Big Data Analytics Project

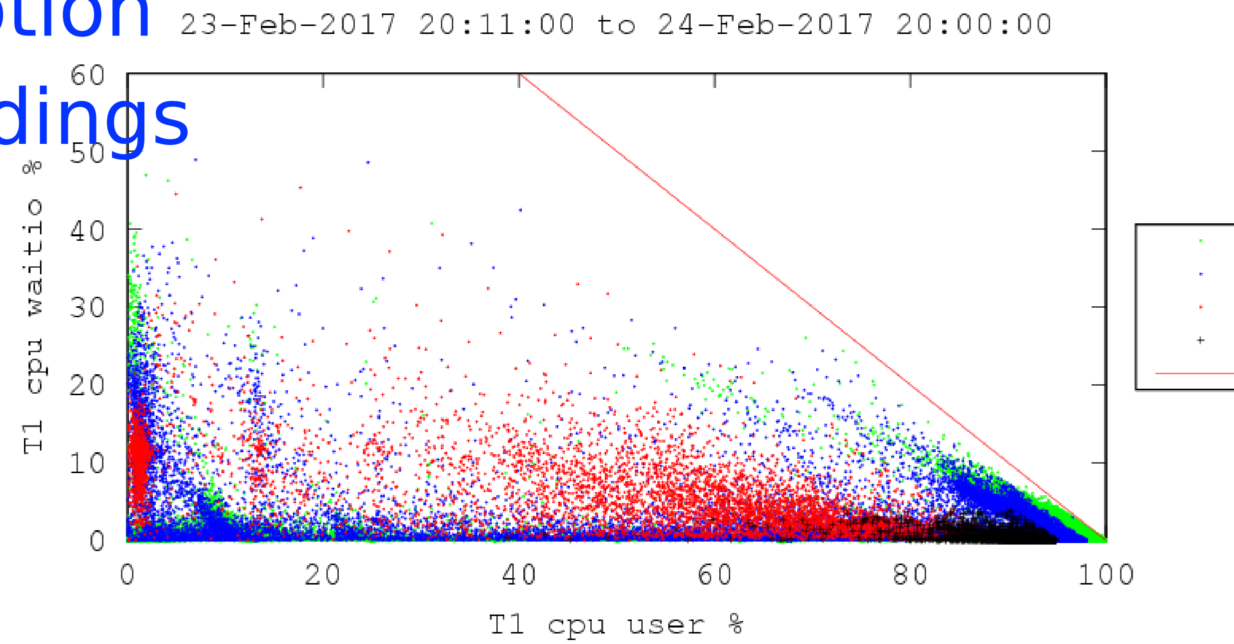
Long-Distance Data Transmission Performance Optimization is Fundamental to the Distributed Computing Infrastructure

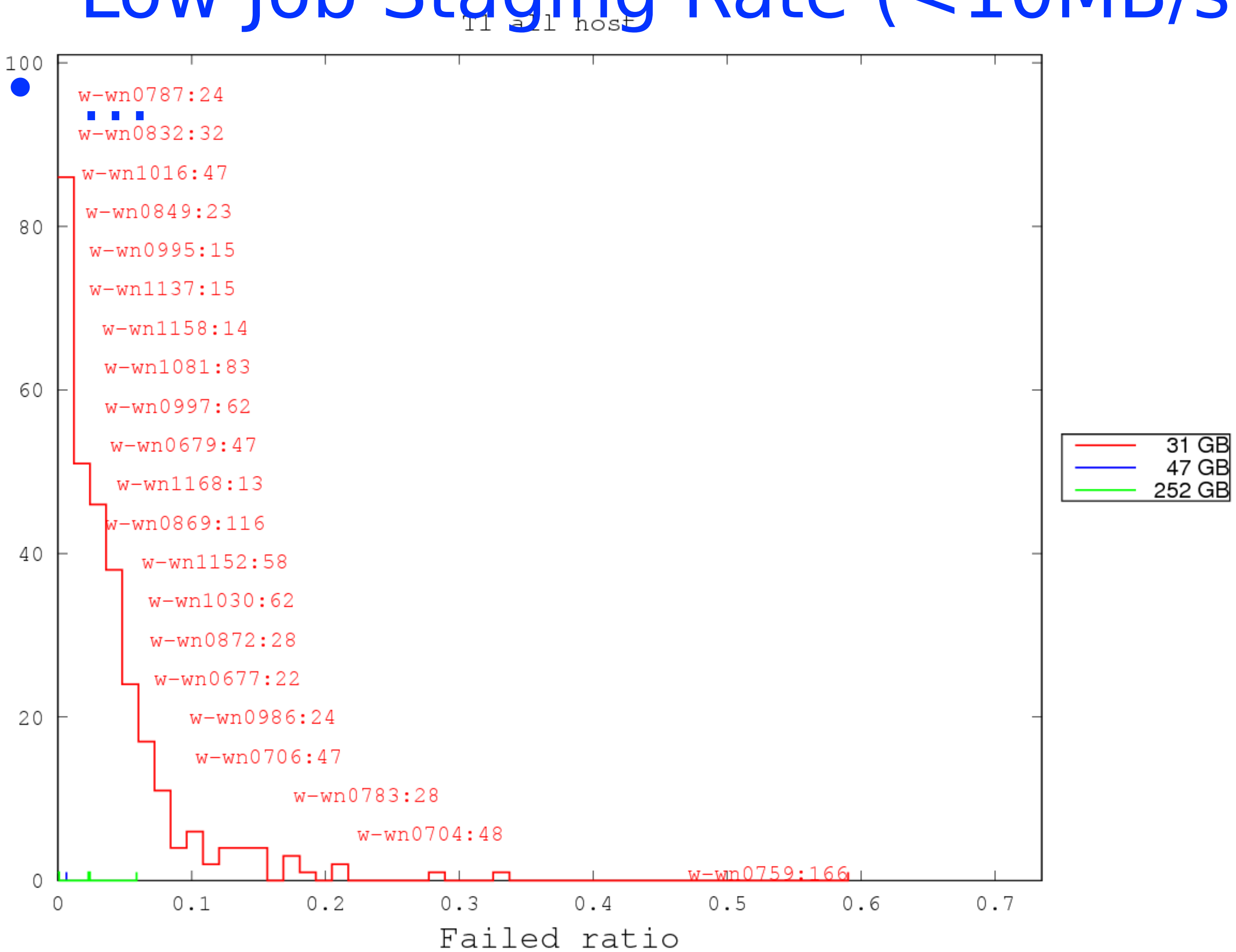
- Single Storage Server (10GbE) is able to achieve 1GB+ write performance with data input from CERN
- How about 100Gb?

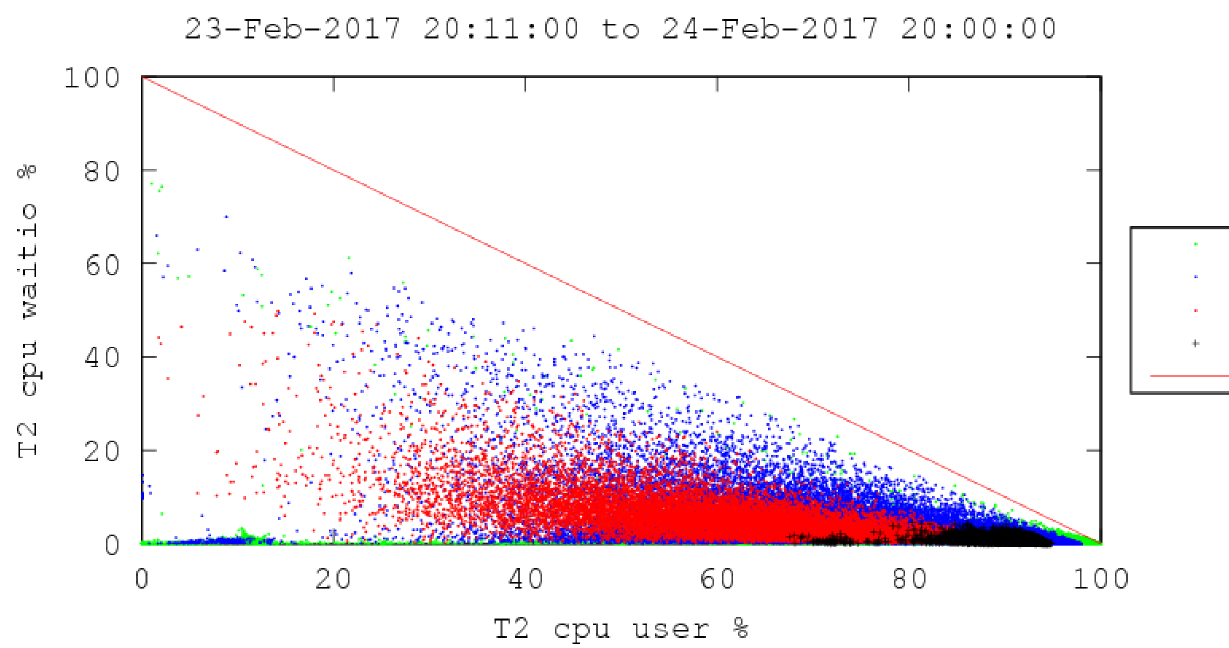


Case Study 1, from WLCG resources: Single Out the Problematic Box and Detect System Bottleneck

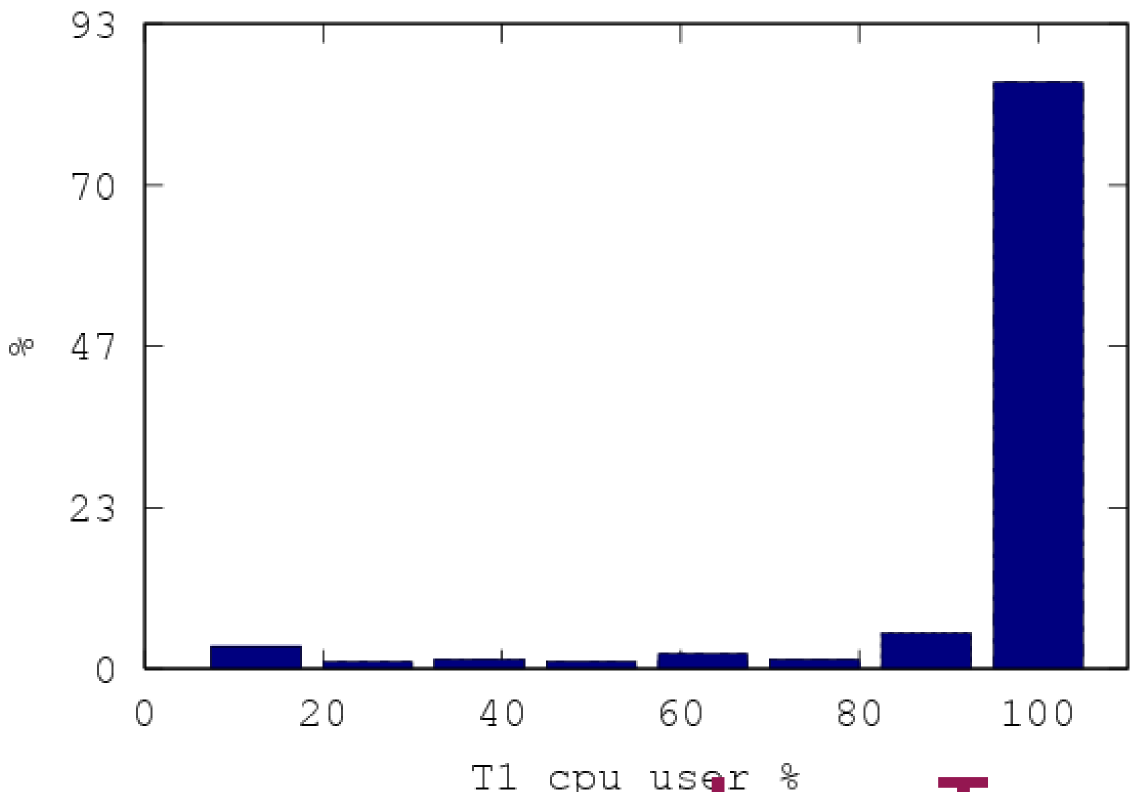
- High Waitio (higher than average, or higher than NxSigma)
- High System Time ratio: contributed by Merge jobs and/or system bottleneck ?
- High Failure Job Rates
- Unable to reach 100% CPU consumption
- Low Free Memory without heavy loadings
- High Swap Usage
- Low Job Staging Rate (<10MB/s)
- ...





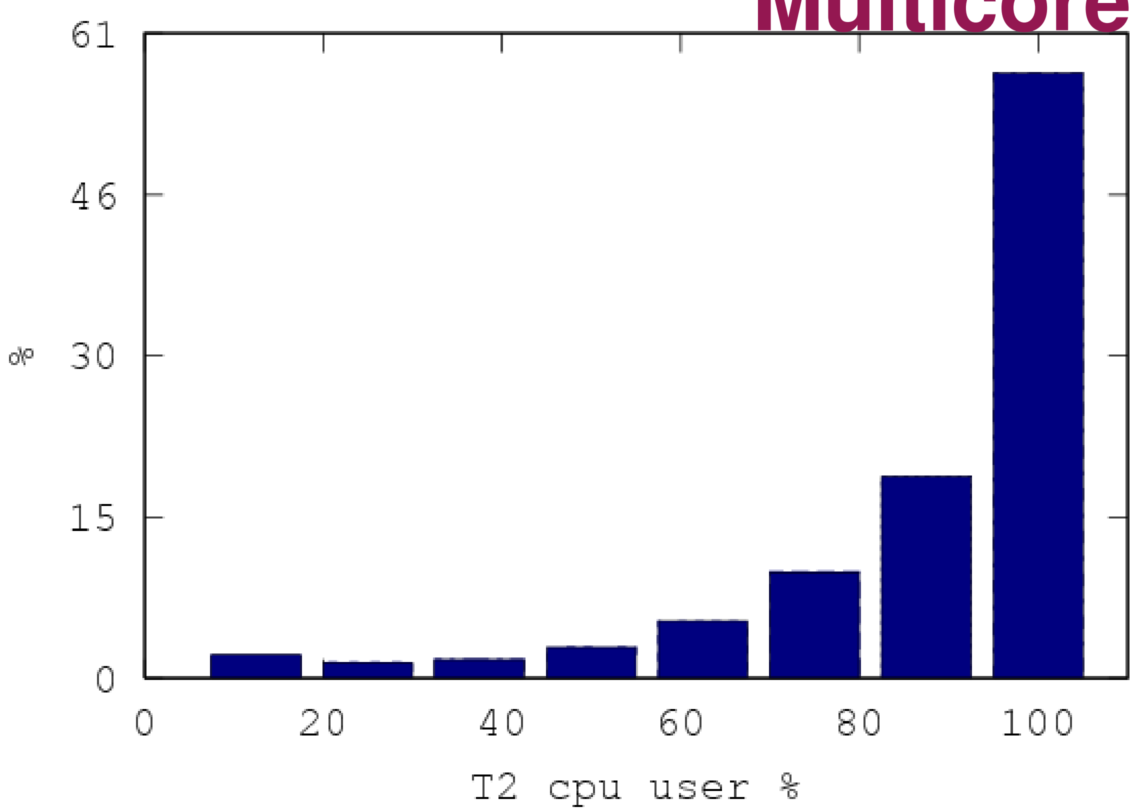


Optimization of System Resources by Investigating Behaviors of Jobs in the Mixed Single-Core and Multi-Core Jobs Environment



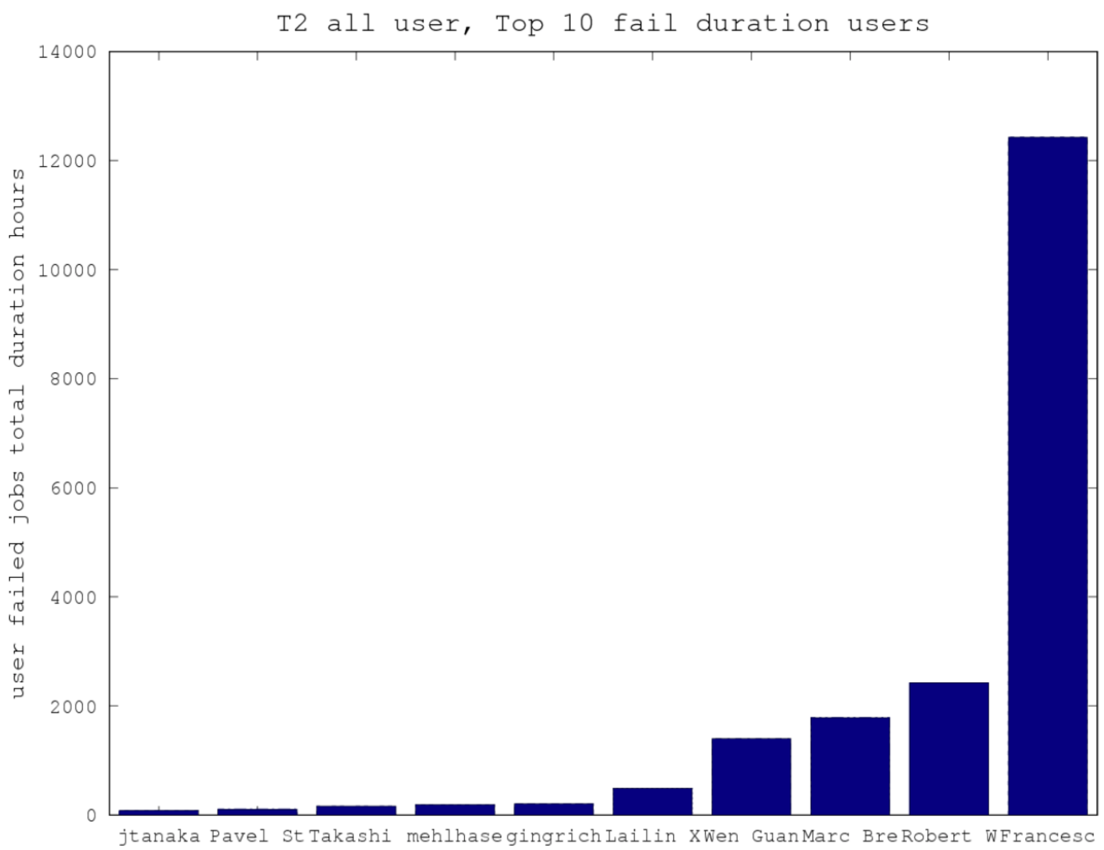

```
Single core jobs :
    finished duration time percentage : 96.13
    failed duration time percentage :  3.68
    cancelled duration time percentage :  0.19
    jobs efficiency :  0.93
    stage in time percentage :  0.46
    files transfer error time percentage :  0.13
Multi-core jobs :
    finished duration time percentage : 95.96
    failed duration time percentage :  4.01
    cancelled duration time percentage :  0.02
    jobs efficiency :  7.21
    stage in time percentage :  0.67
    files transfer error time percentage : [illegible]
single to multi core job slot idle         47 hours/WN
multi to single core job node idle       0.29 hours/WN
```

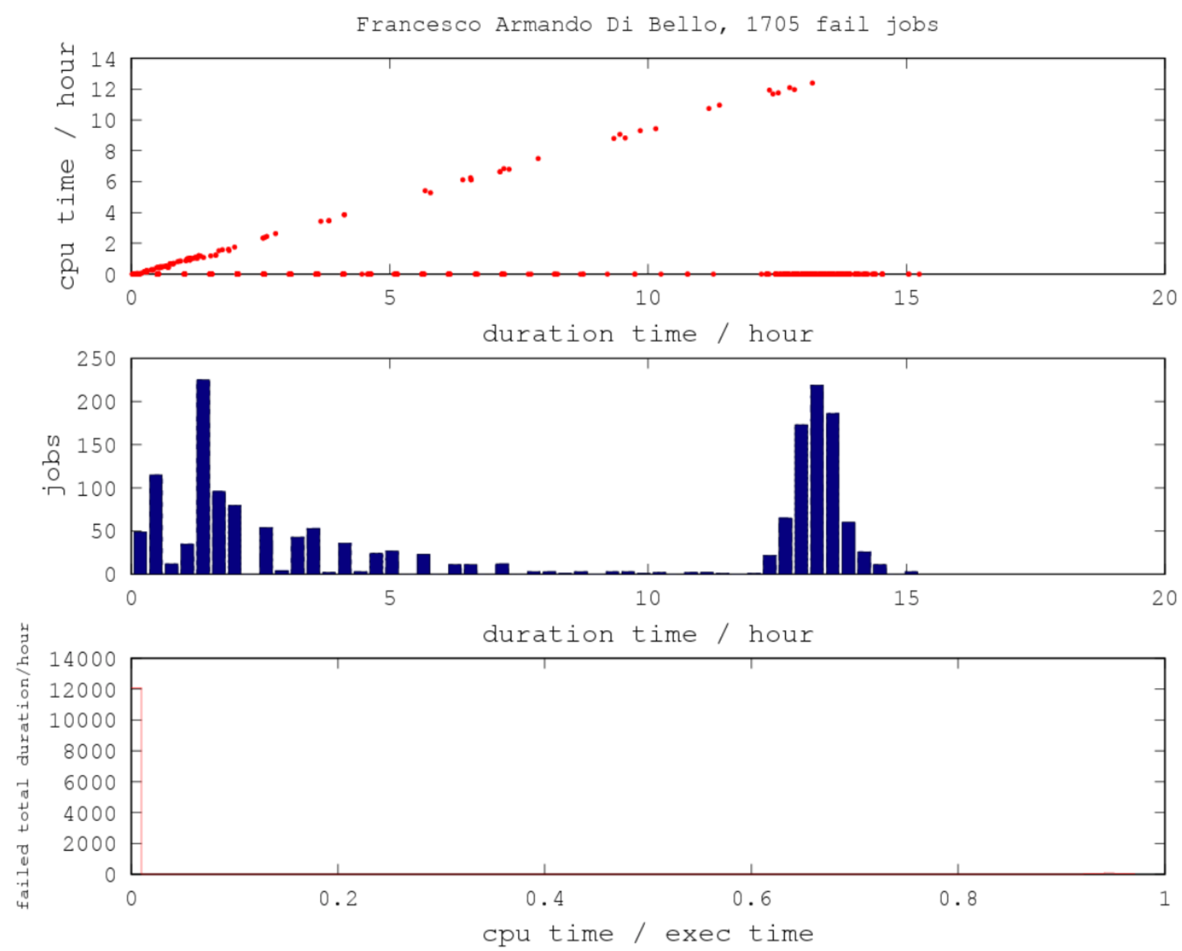
Long Transition Time between Single and Multicore Jobs Means CPU Time Waste

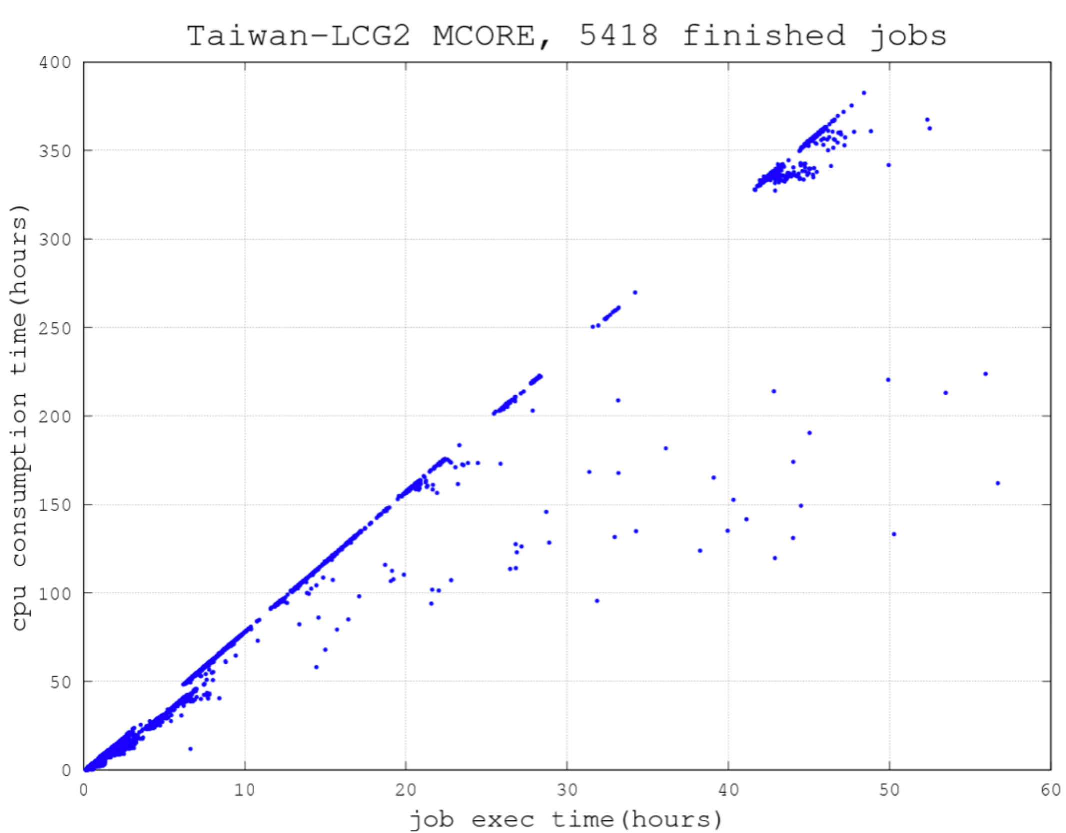


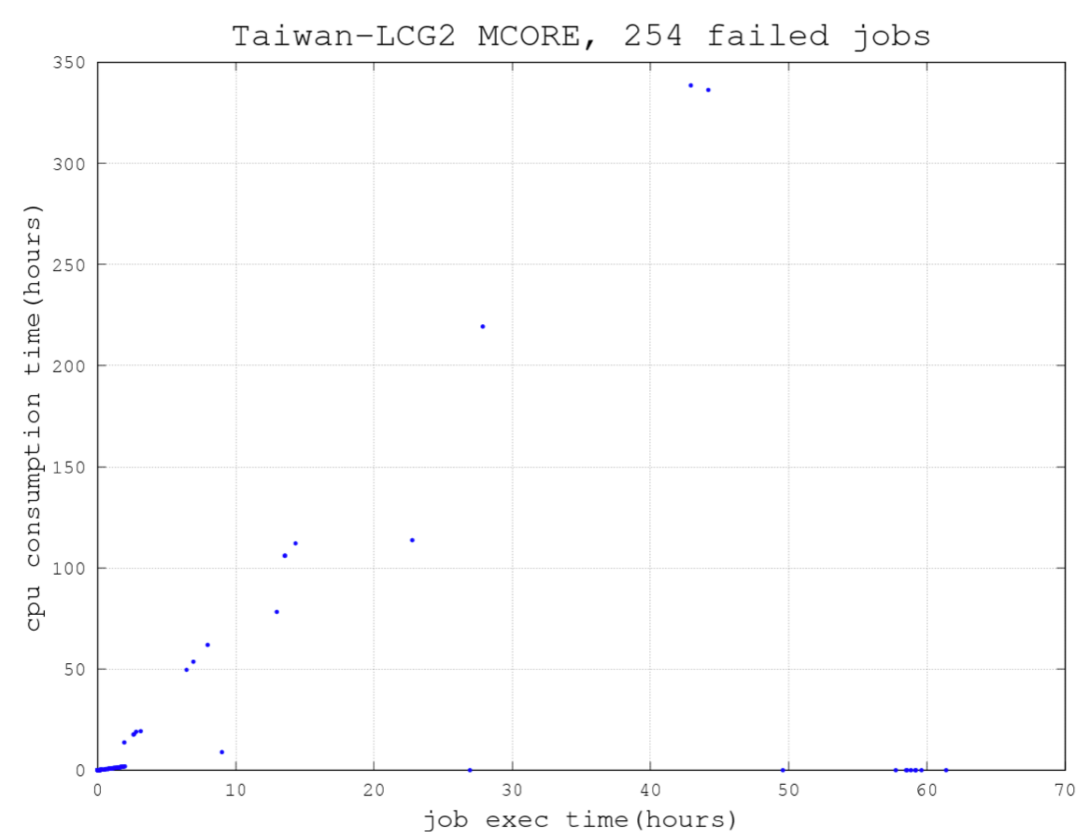

```
Single core jobs :
    finished duration time percentage : 67.52
    failed duration time percentage : 32.15
    cancelled duration time percentage :  0.33
    jobs efficiency :  0.66
    stage in time percentage :  3.63
    files transfer error time percentage :  8.27
Multi-core jobs :
    finished duration time percentage : 92.41
    failed duration time percentage :  7.58
    cancelled duration time percentage :  0.01
    jobs efficiency :  7.76
    stage in time percentage :  0.06
    files transfer error time percentage : [illegible]
single to multi core job slot idle :        3 hours/WN
multi to single core job node idle :     0.19 hours/WN
```

T2 Jobs Not Consuming CPU Should be Purged



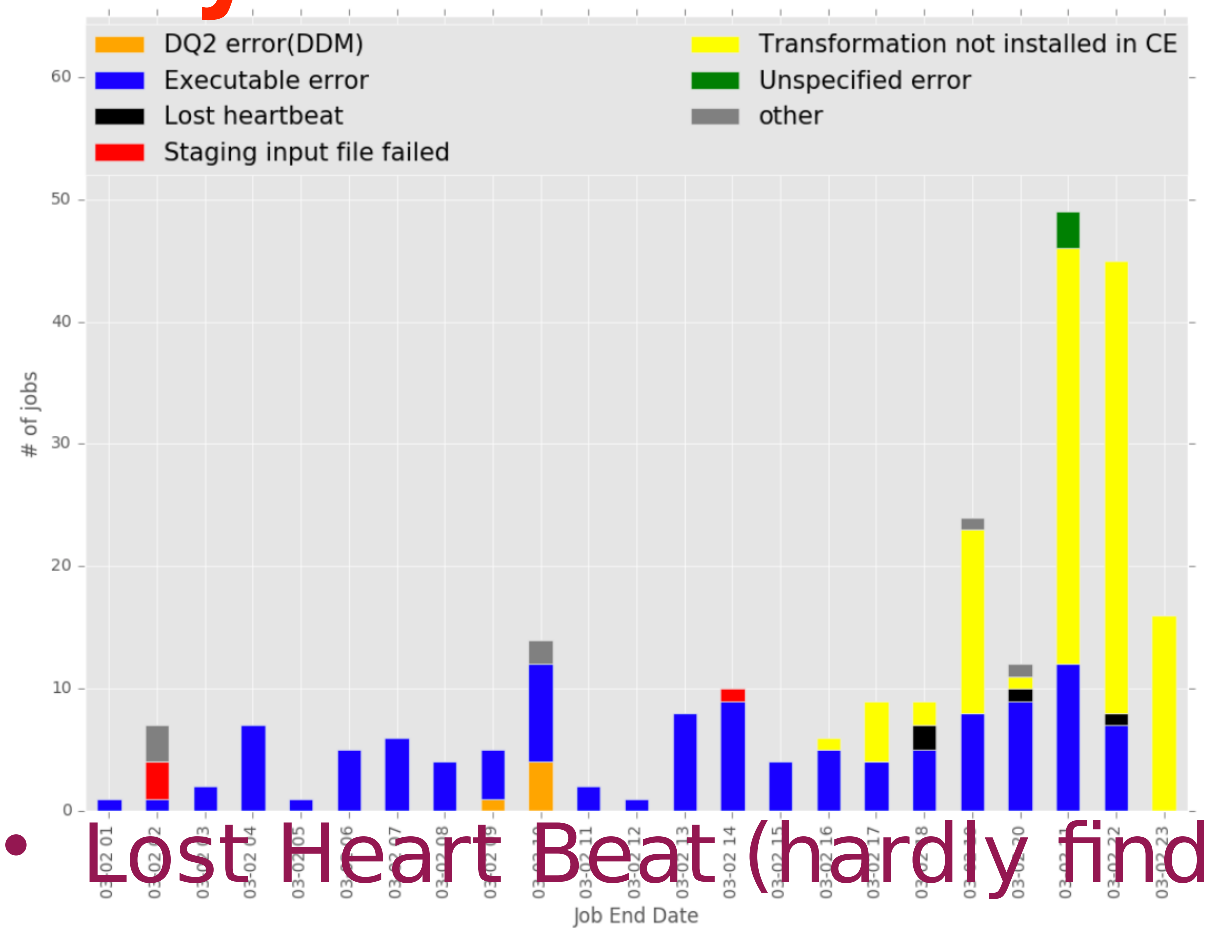


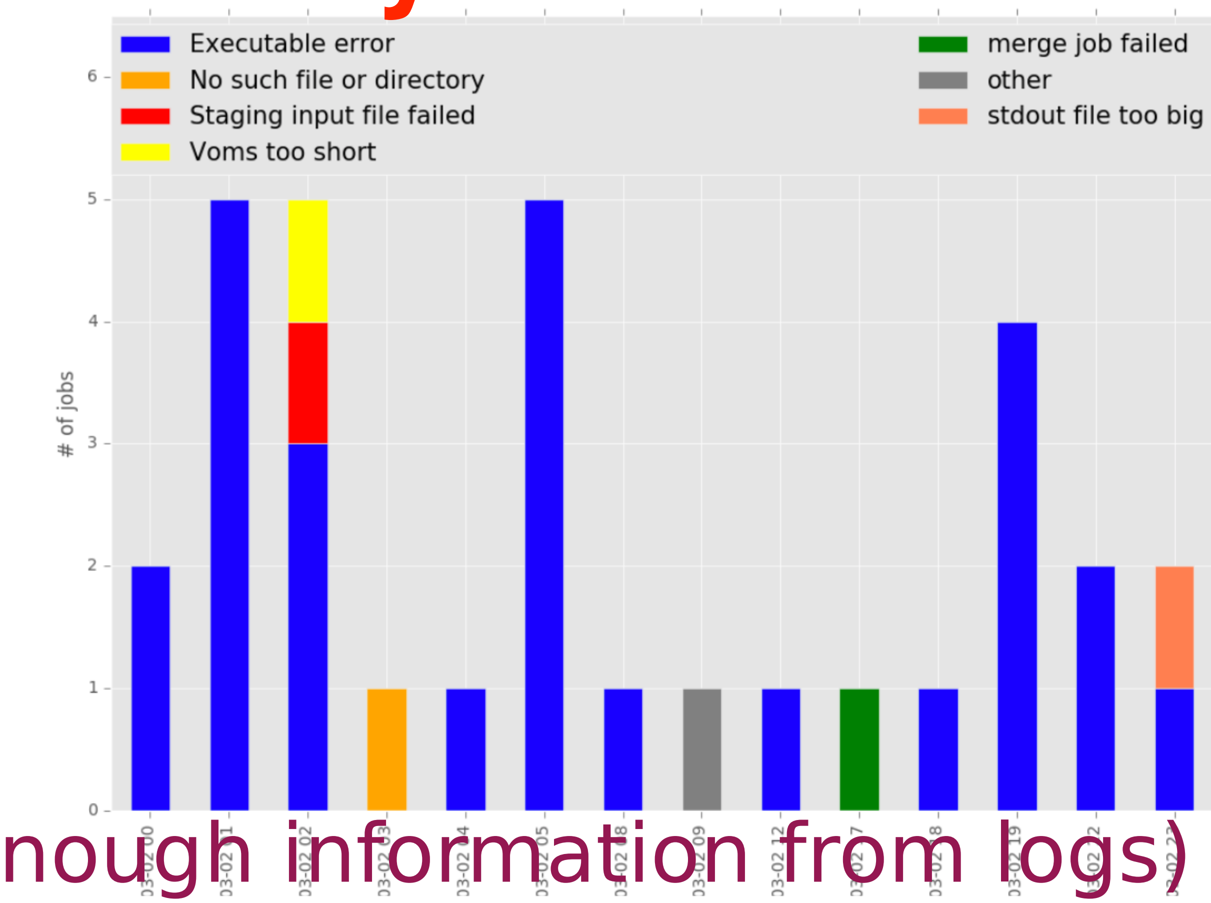




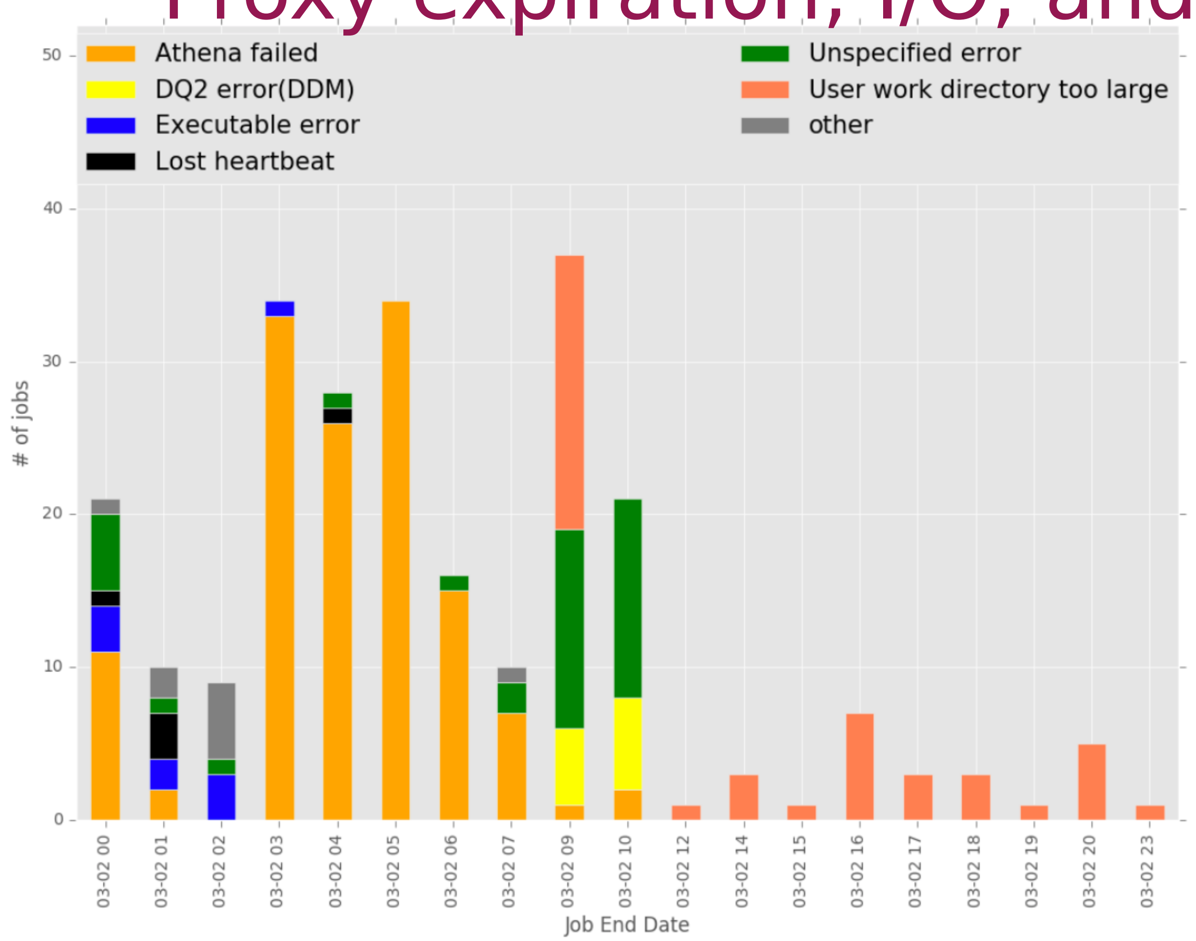
Case Study 2(WLCG resources): Job Failure Analysis to Understand Workload Characteristics and to Enhance Job Efficiency (by avoiding Failure Again due to Local HW/SW faults)

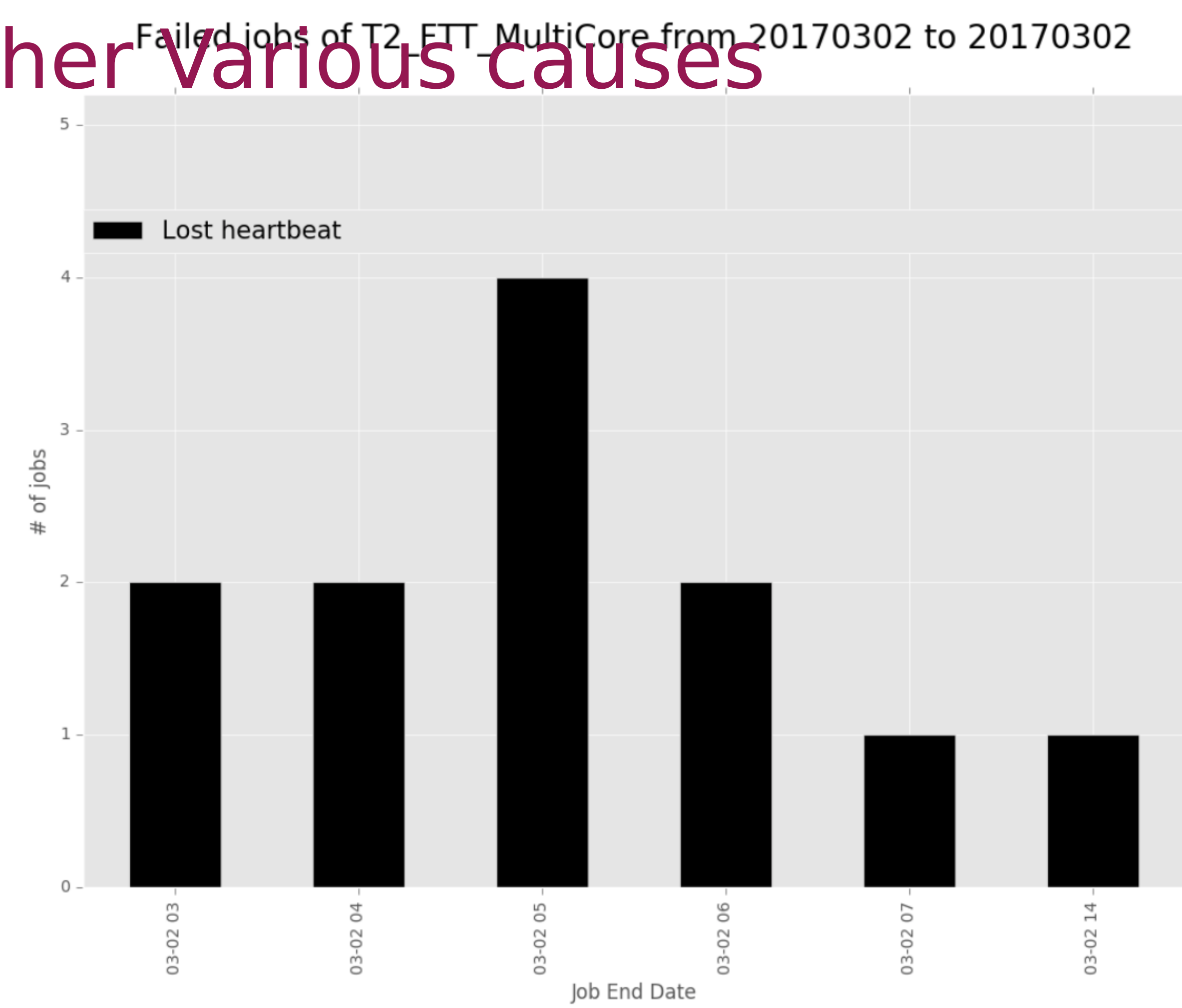
Analysis of Failure Jobs for System Efficiency



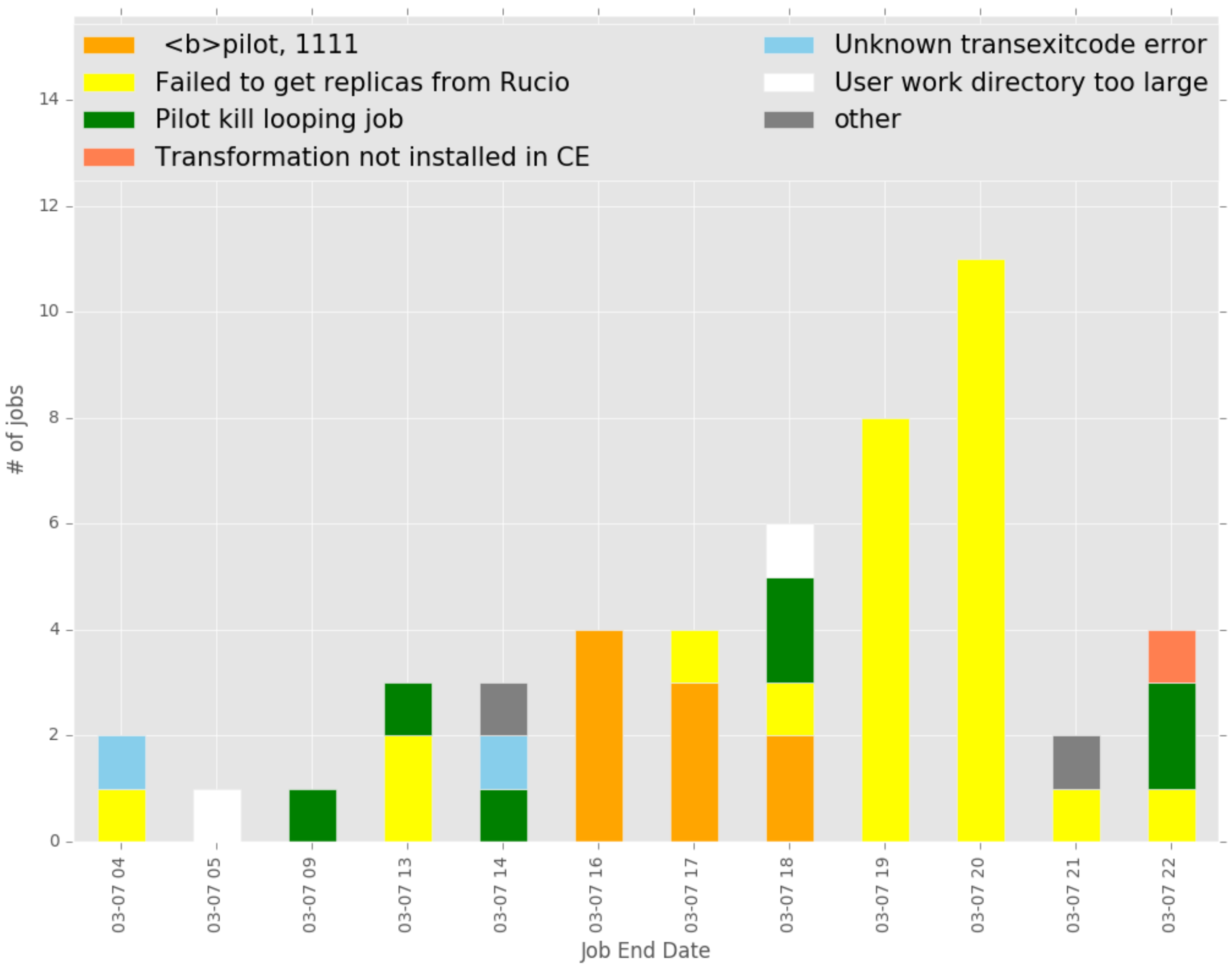


- Lost Heart Beat (hardly find enough information from logs)
- Proxy expiration; I/O; and other Various causes

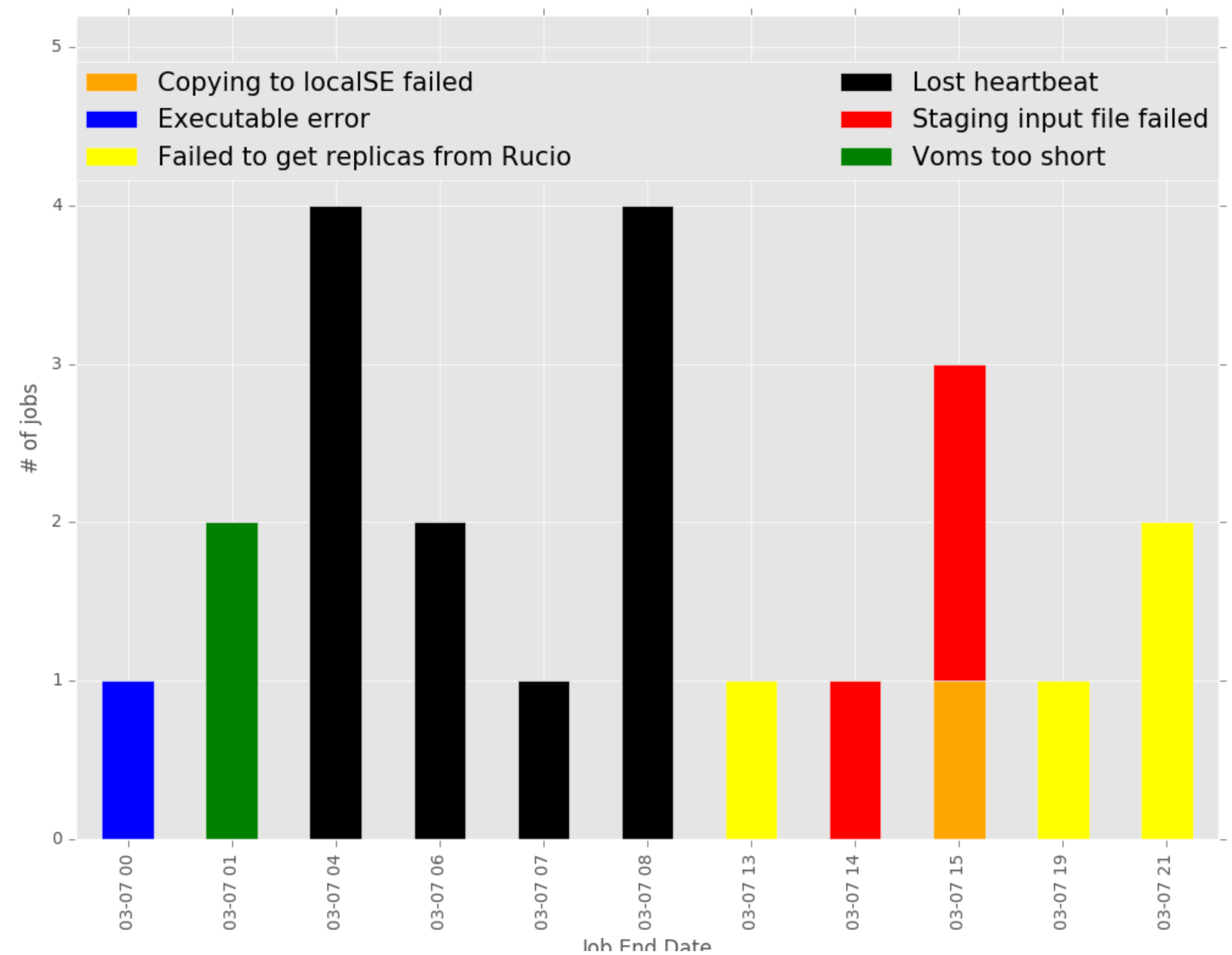




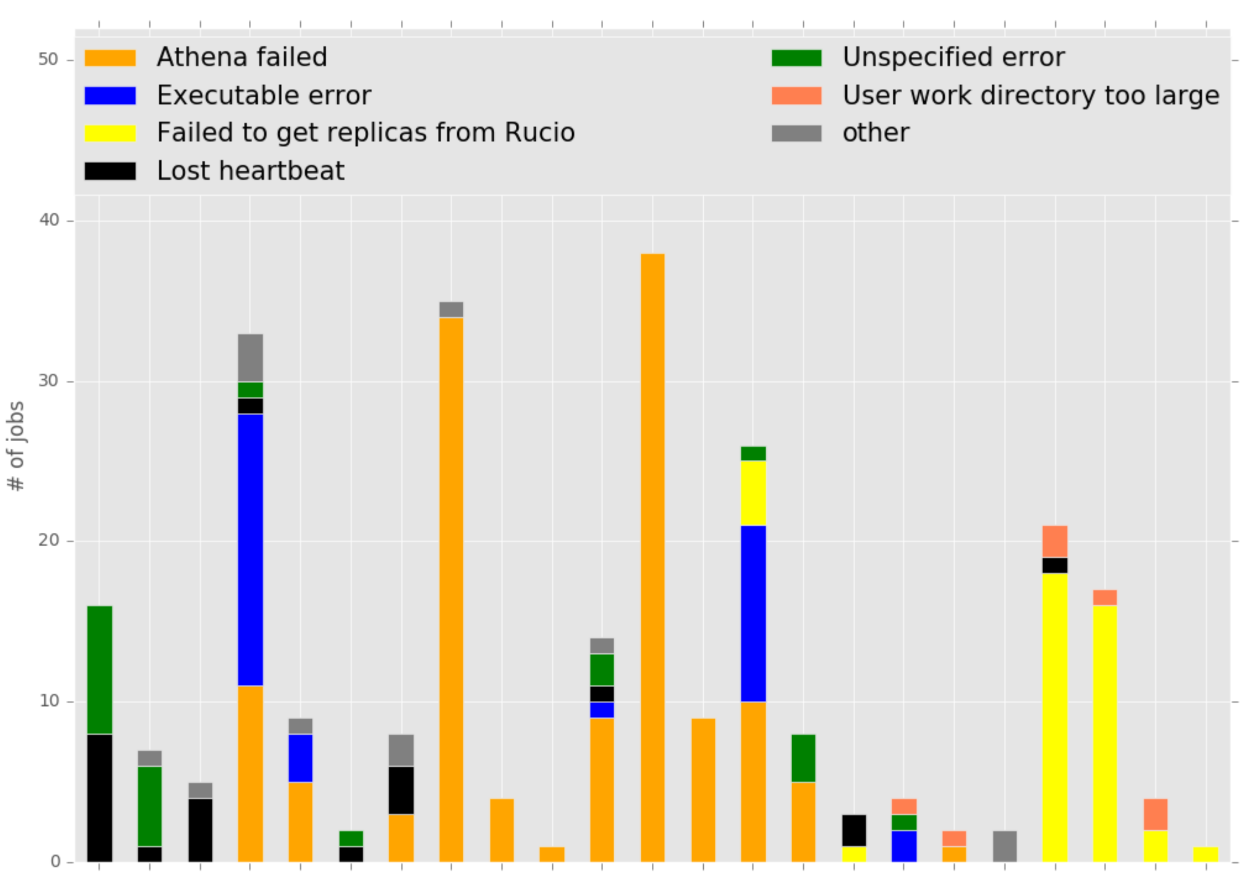
Failed jobs of T1_SingleCore from 20170307 to 20170307



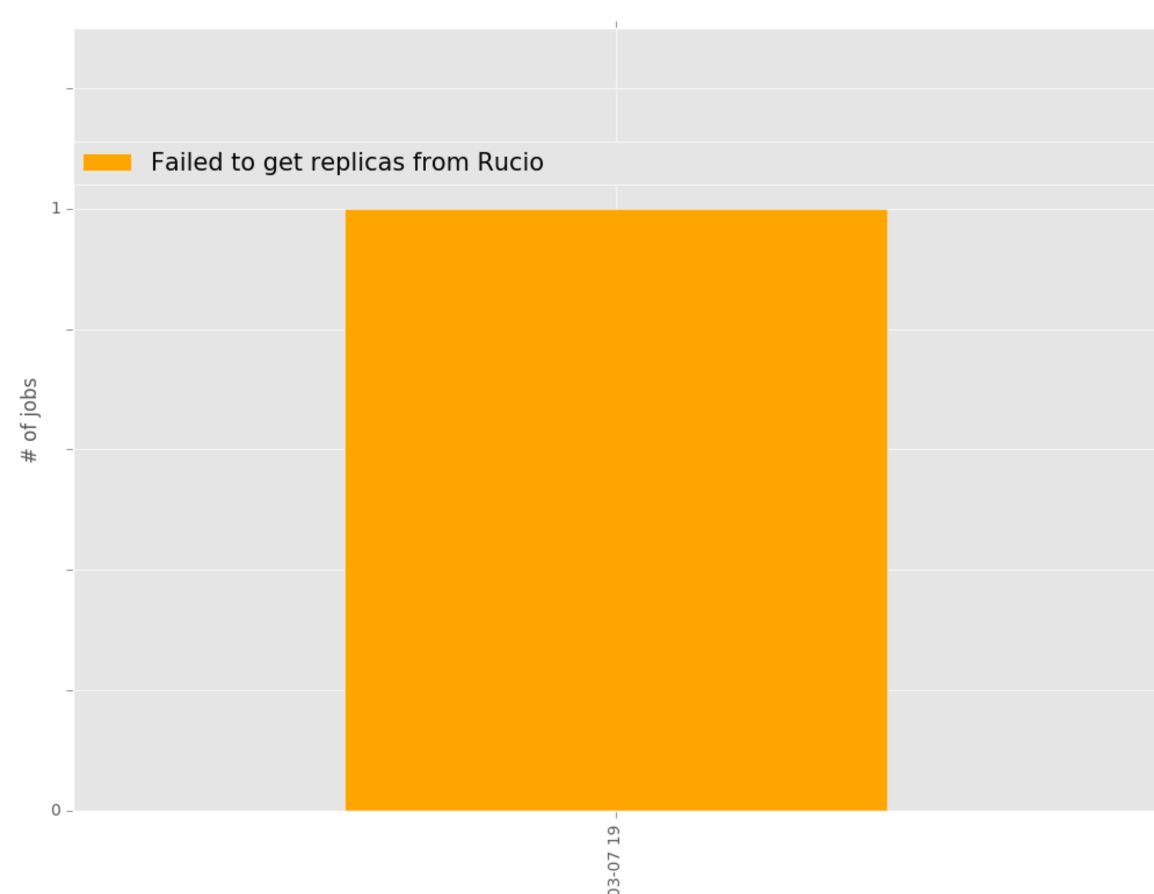
Failed jobs of T1_MultiCore from 20170307 to 20170307



Failed jobs of T2_SingleCore from 20170307 to 20170307

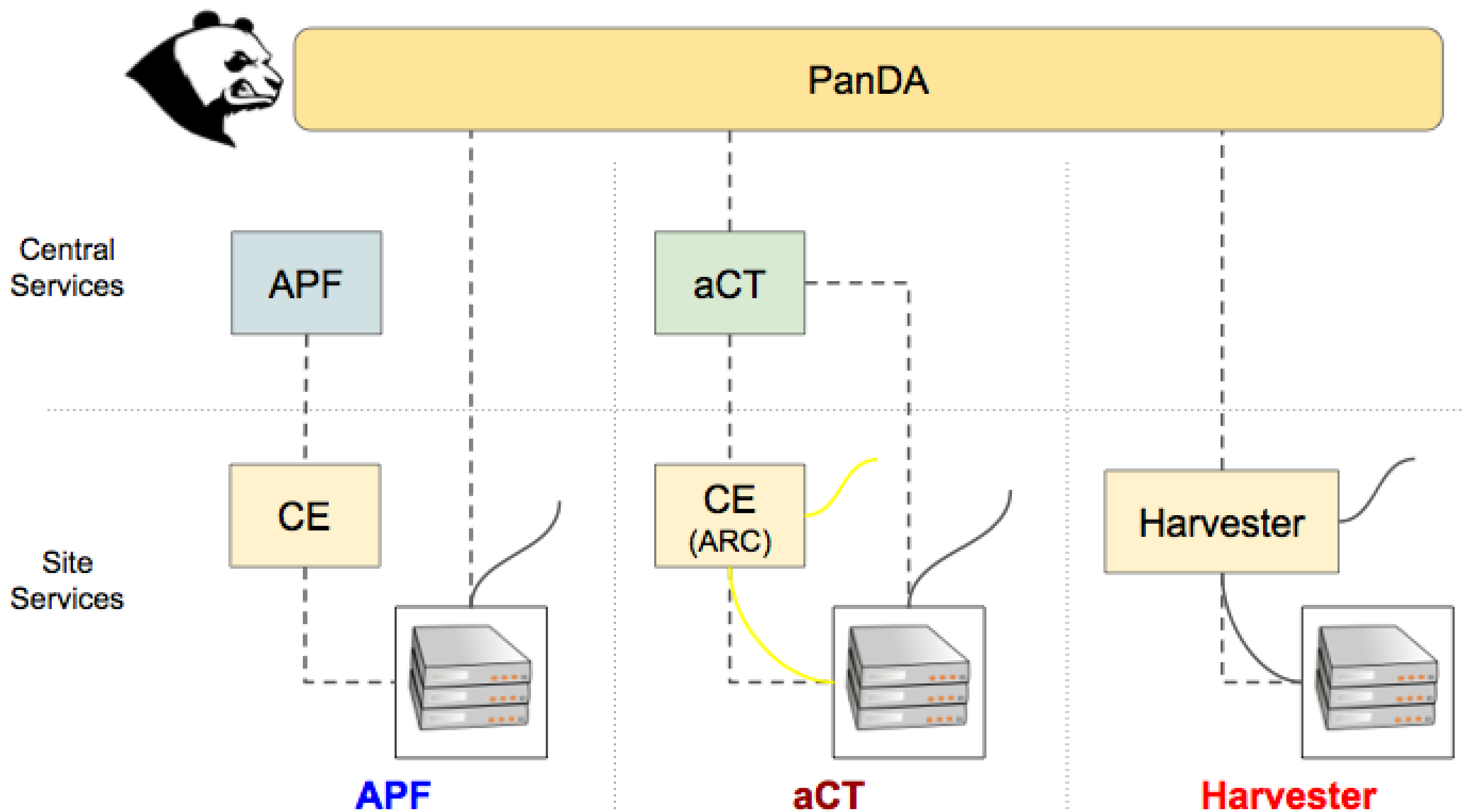


Failed jobs of T2_FTT_MultiCore from 20170307 to 20170307



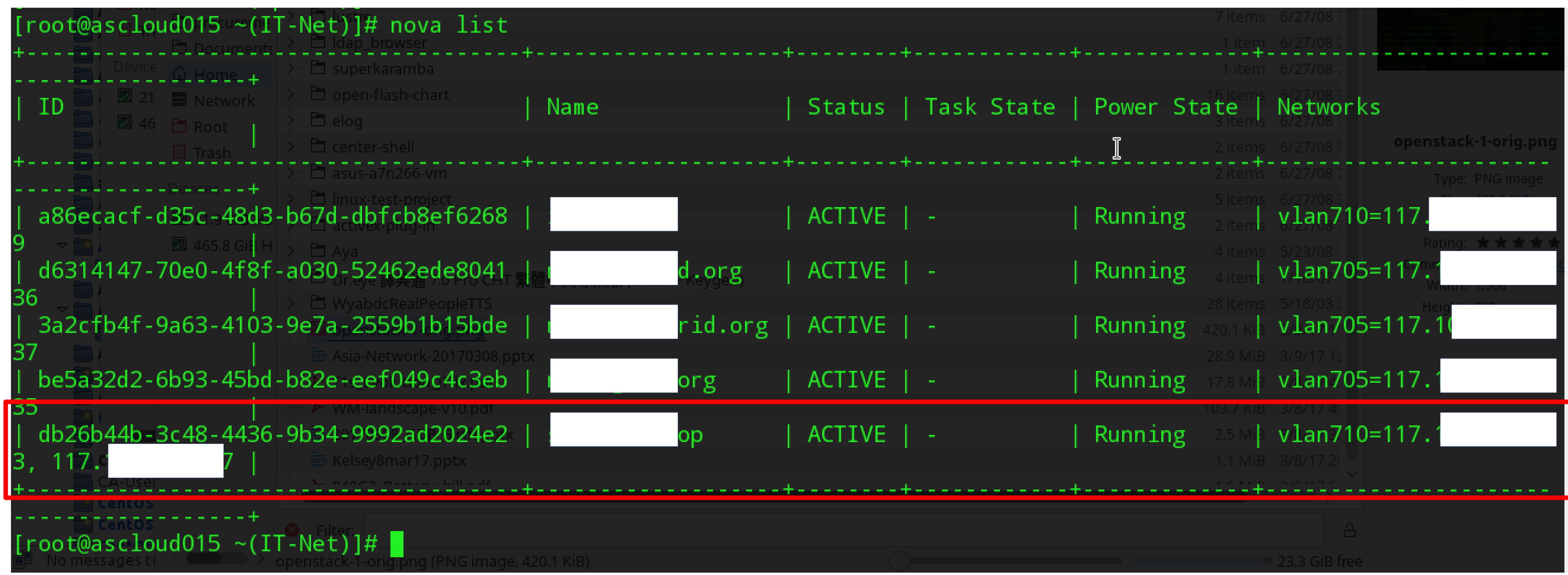
Joint Development of Harvester for PanDA

- On Stage-in/out modules to fit local workflows of HPC jobs in DiCOS



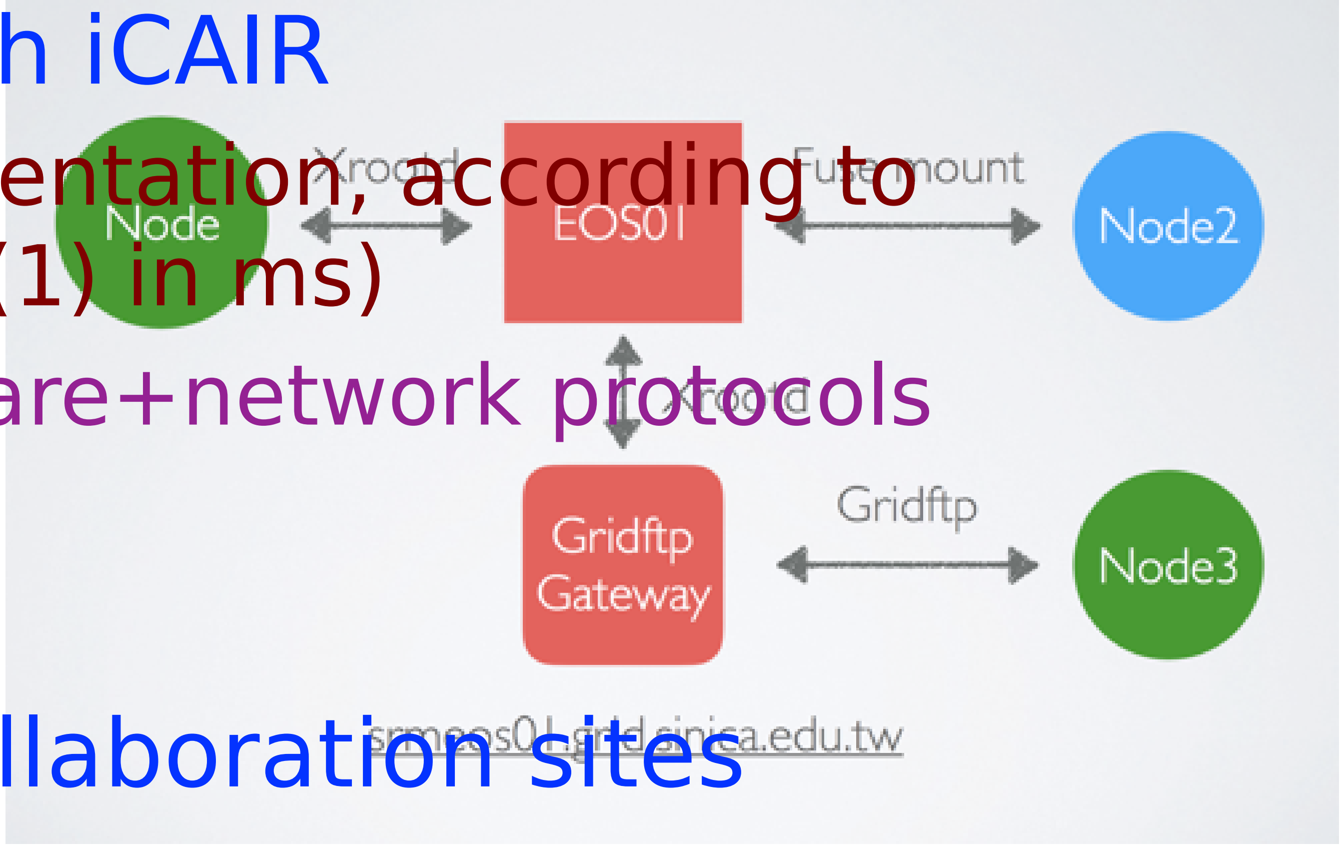
Efficiency for Cloud

- Openstack scheduler enhancement.
  - Identify problematic hypervisors and excluding them



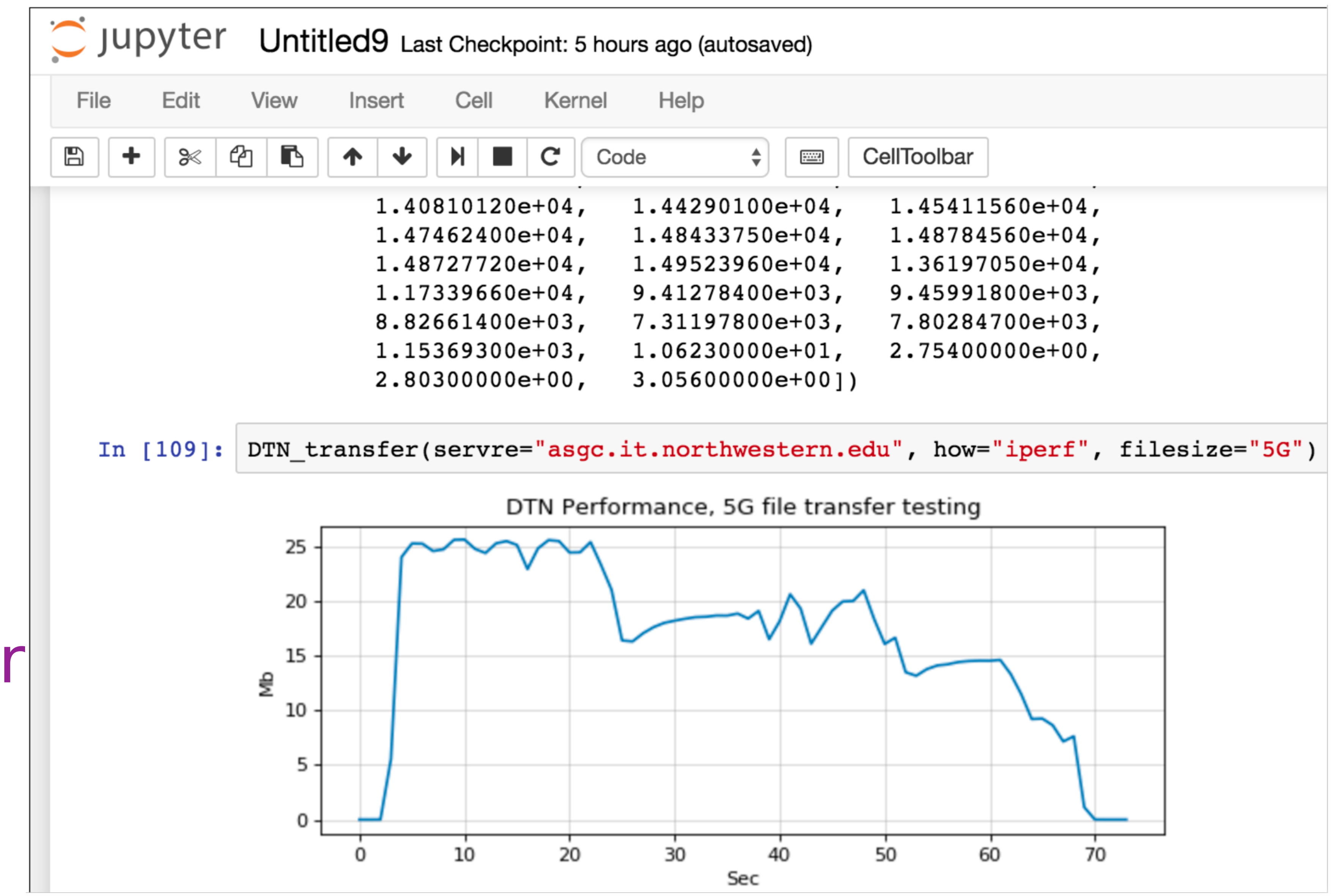
DTN for Data Transmission Efficiency

- Objectives
  - Provide stable and predictable data performance
- Supported by smart middleware to orchestrate data flows among DTNs
  - Integrate machine learning and monitoring mechanisms
- Experiments and Experiences: join WLCG testbed and the collaboration with iCAIR
  - Buildup reference implementation, according to latency (O(100), O(50), O(1) in ms)
  - Exercise hardware+software+network protocols measurements
- DTN for WLCG
- DTN between primary collaboration sites



Collaboration with iCAIR on DTN

- DTN with EOS (to be demonstrated at iCAIR on March 15)
  - Developing the scripts that simplify the file transfer process and visualize the Benchmark on the Jupyter. (In progress)
  - Implementation
    - matplotlib
    - psutil
- Integration
  - iperf
  - nuttcp
  - EOS data transfer

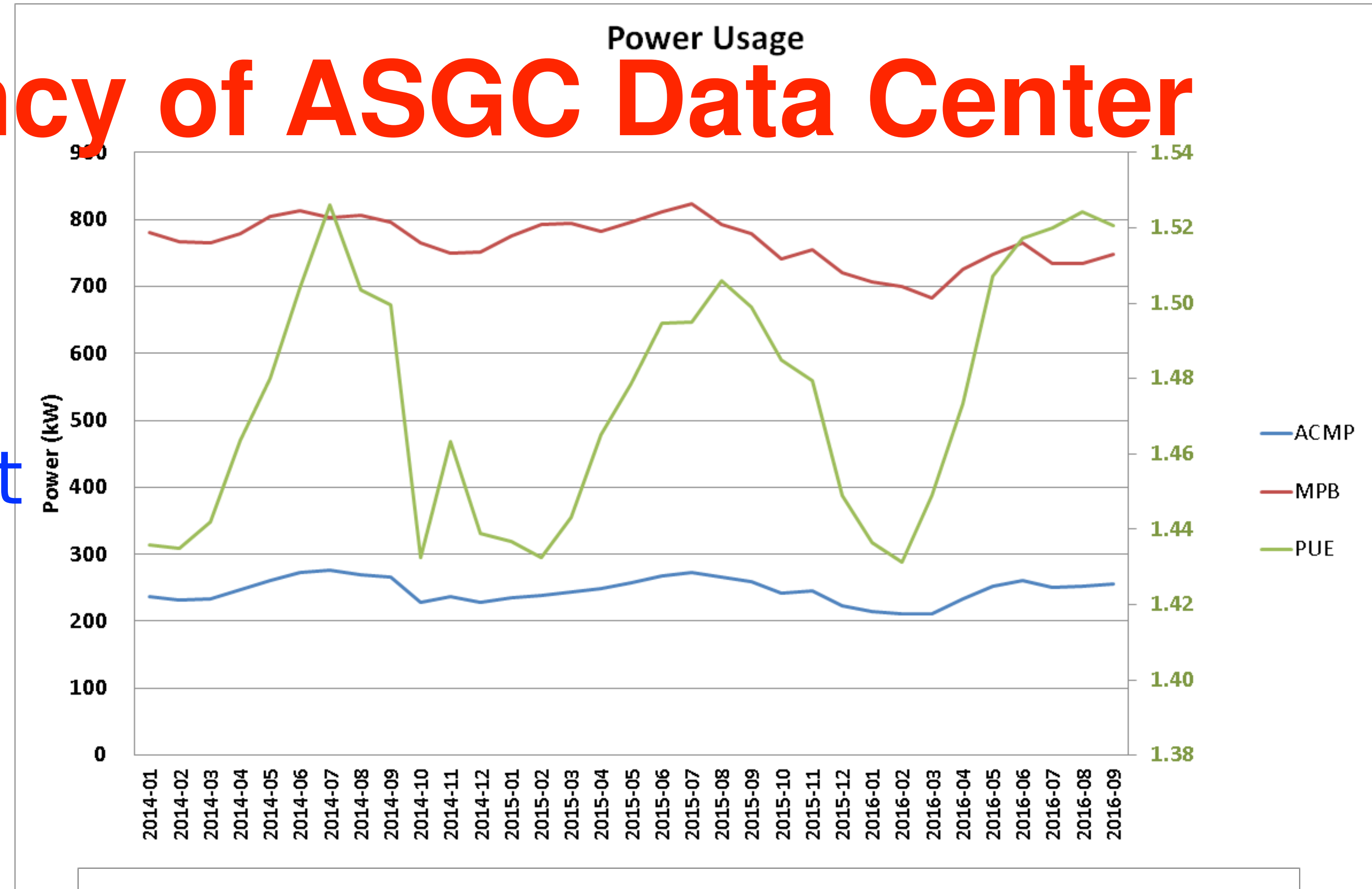


Security:

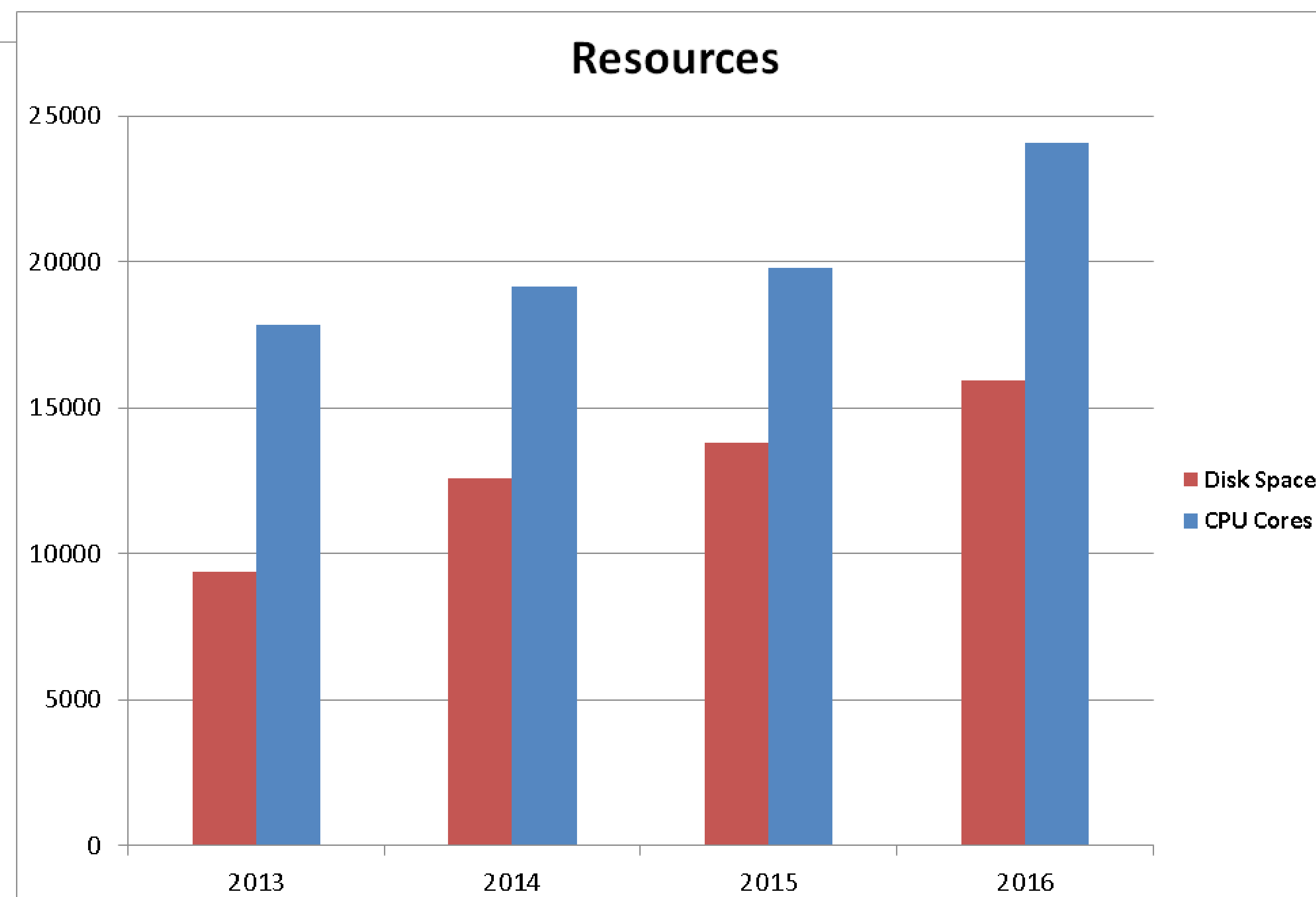
- Investigating Regular Behavior Patterns
- Anomaly Detection
- Automation of Analysis Processes
- Support by the blacklist database with MongoDB
  - Using MongoDB for scalability and read-oriented applications
  - MariaDB is superior in write performance

	1K	10K	100K	1000K
MariaDB	1.21s	15.87s	6m40s	924m45s
MongoDB	1.16s	9.65s	1m51s	65m37s

Power Efficiency of ASGC Data Center

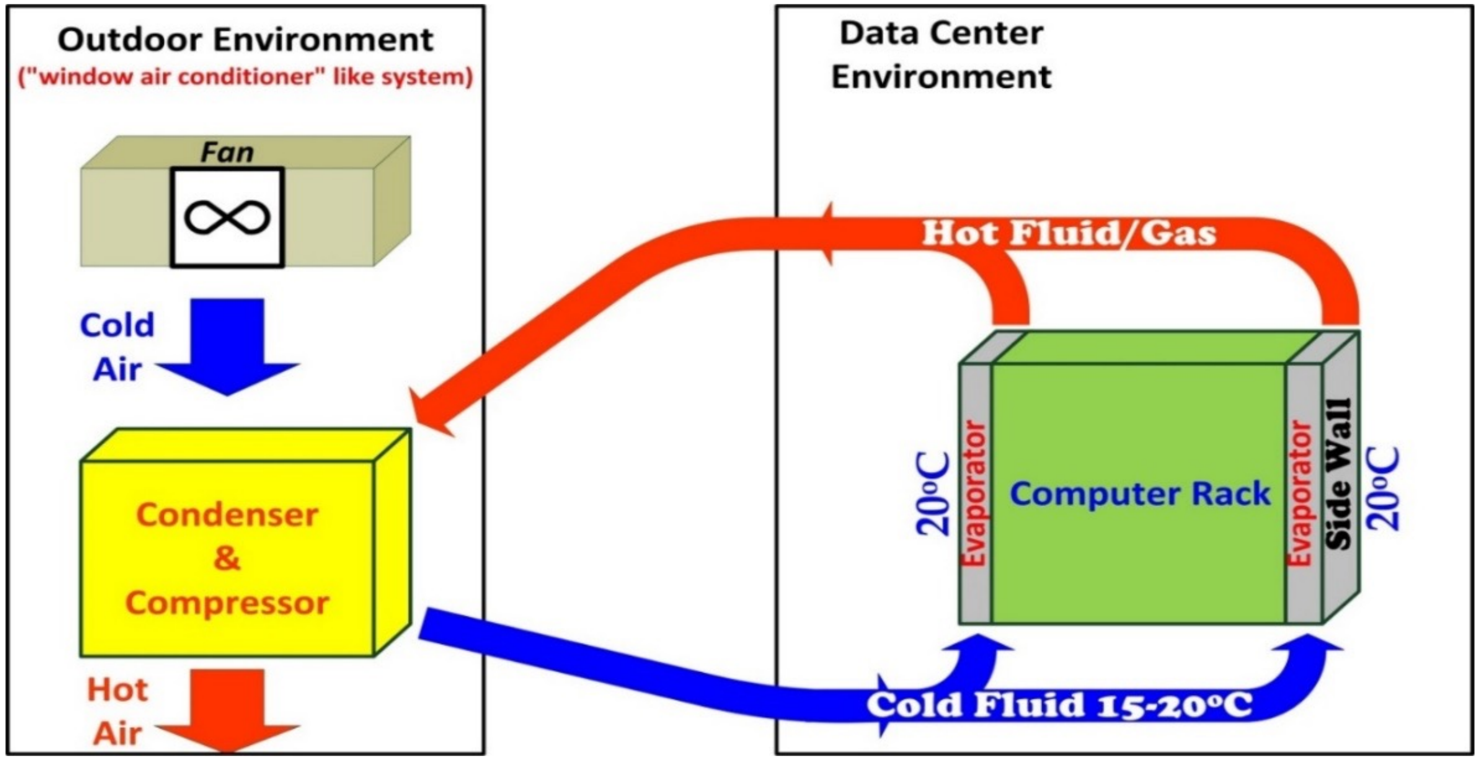


- Though power consumption does not grow while IT facility increasing annually, the PUE is not good enough.
- Single Rack Data Centre made by fan-less conditional cooling architecture had been prototyped
- Smart Power Control is also under experiment

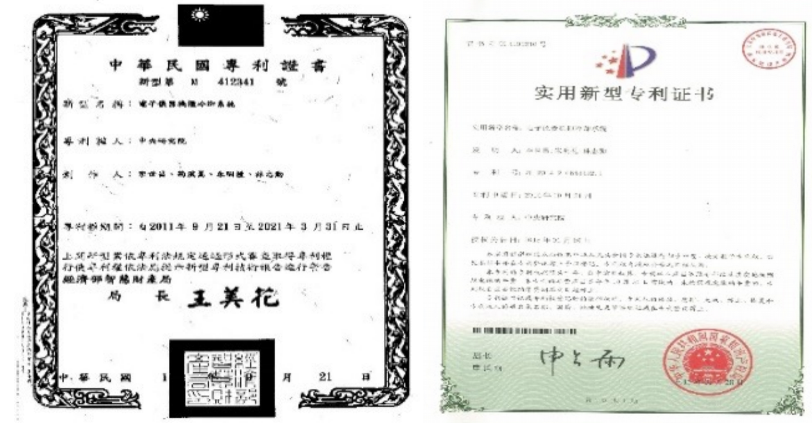


Fanless Single Rack Cloud Center

- *Novel Conductional Cooling Architecture based on space technology*



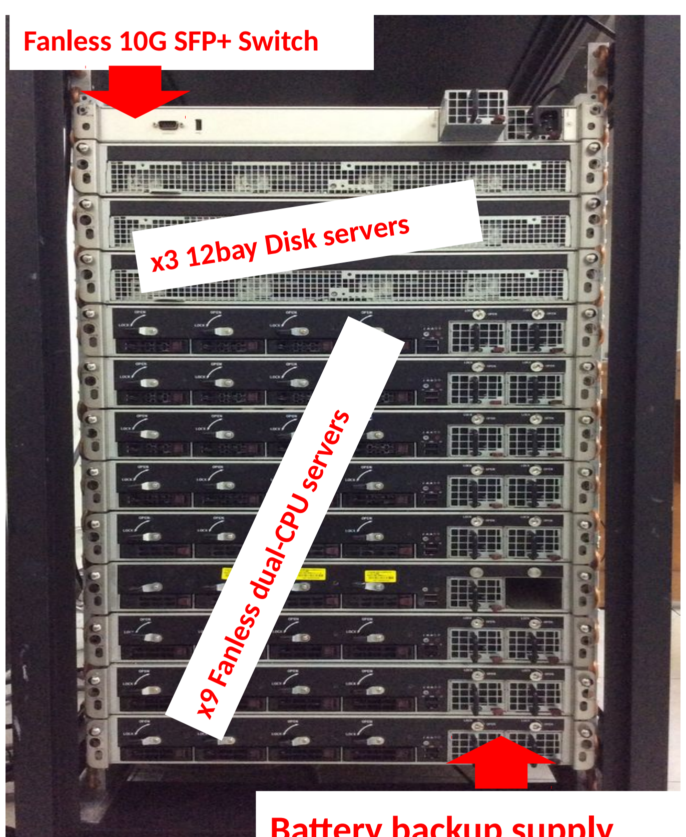
From PUE<1.2 —> PUE < 1.1

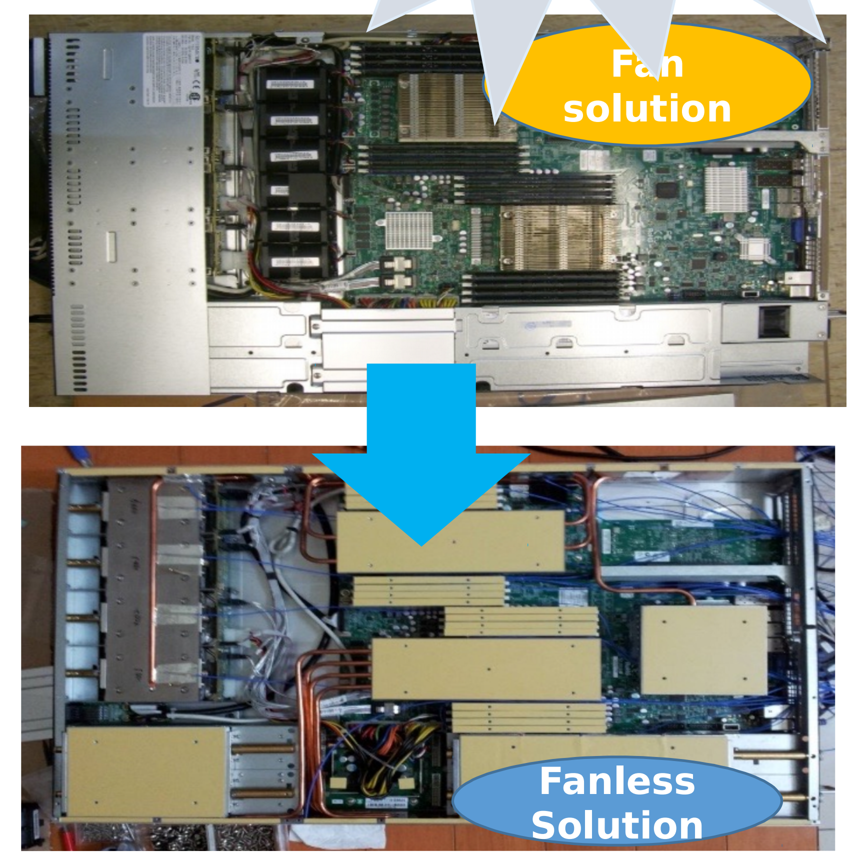


*US patent approved

No Noise
High PUE
Easy installation

- *First fanless rack prototype modified from SMC products*

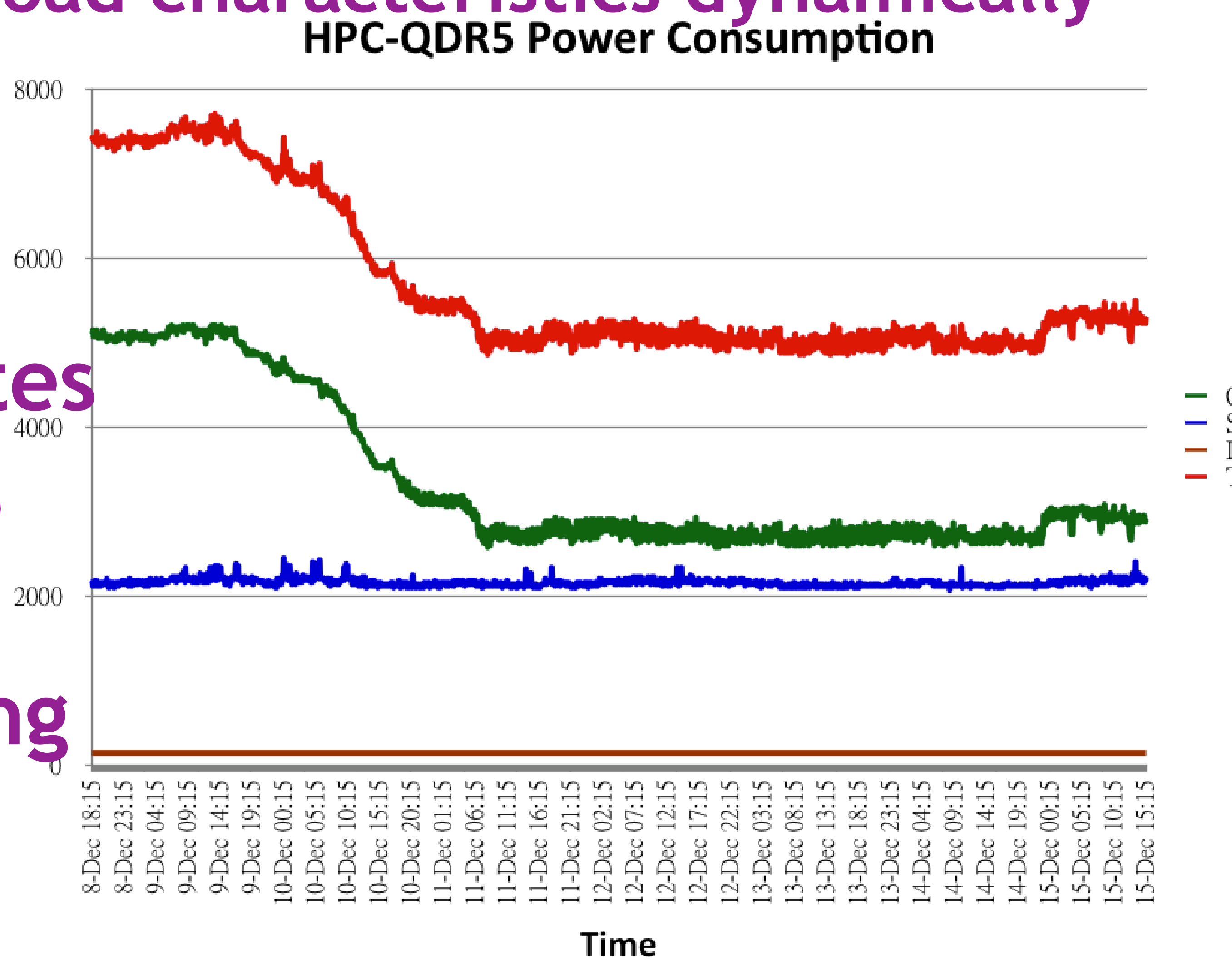




Smart Power Control for IT Equipment

- We have achieved good cooling efficiency. Most of energy is taken by IT equipment in a single rack cloud center.
  - Minimize power consumption of standby servers (might save up to 50% of its maximum power).
  - Smart power manager will optimize energy consumption and performance according to workload characteristics dynamically

- Smart Power Manager
  - Flexibility: Active, Hibernation, Shutdown States
  - Develop a model to arrange workloads with energy-efficiency without sacrificing performance



Summary

- In addition to Power, Thermal and Operation Efficiency, we are Extending to Distributed Cloud Sites and Applications Efficiency
- Efficiency Improvement is a Routine Task and should be Enhanced by Knowledge of Broader System Perspectives and Machine Learning Technologies
- Started from statistics of behavior patterns of finished and failed jobs by types of jobs (also the input), we like to differentiate the anomalies and to detect failure jobs as early as possible.
- Efficiency Improvement has to be automatic based on What we Learned from Daily Analysis.
- Deploying Machine Learning Technology is a must, but is still challenging.