Lecture 4: parallel computing, batch systems, cluster monitoring

https://sites.google.com/site/clustergateorg/

- Parallel performance
- Job — task to perform data conversion
- Batch processing

- Batch processing systems

Andrey.Shevel@pnpi.spb.ru

Parallel computing

- **The cluster (and even one server) consists of many independent components. It gives idea that a range of operations in computing (in general data conversion) might be divided on some independent steps which can be performed at the same time (in parallel).**
- **It seems parallel operation performance will decrease total time of computing. Quite often it takes place. But not each time.**
- **In reality parallel computing could decrease the total computing time in that degree in which we can divide our task in separate stages.**

Types of parallelism

- **Algorithm pallelism - parallel (almost at the same time) execution of different part of the program (program system).**
- **Data parallelism - parallel (almost at the same time) execution of one program with different data**
  - Statistics values (mean, sigma, etc) in different data sets.

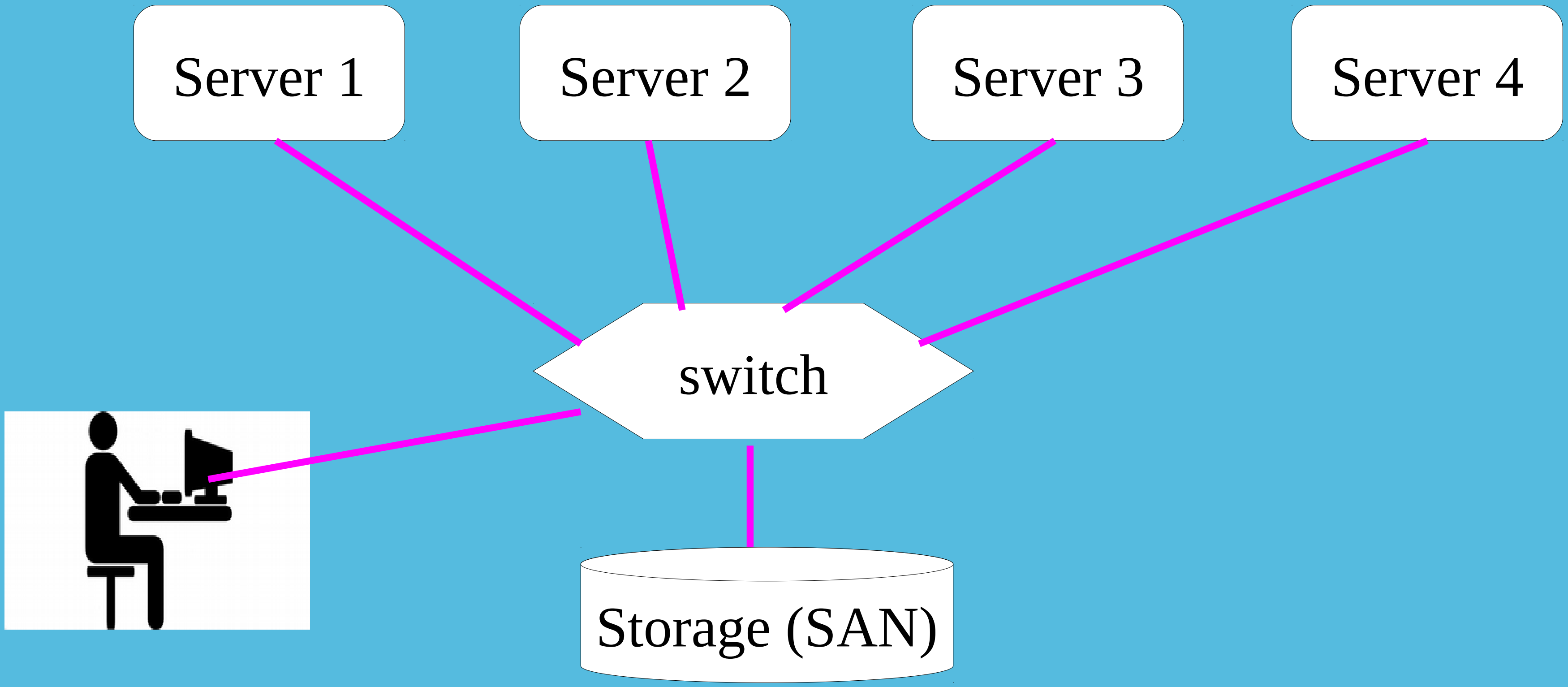
Amdal's law (~1960)

- If *alpha* is share of algorithm (or programs, or task), which can be performed in parallel on N CPUs, than maximum speedup can be achieved on level *1/alpha* even N becomes unlimited. (http://en.wikipedia.org/wiki/Parallel_computinghttp://en.wikipedia.org/wiki/Parallel_computing).

Which situation in computer infrastructure might prevent parallel performance?

What is possible speedup if we upgrade all servers to more powerful?

CPU load 100% for all servers



Storage bandwidth load is 100%

Parallel performance of computing jobs

- **Job** (request to perform something) – the description, which is interpreted by ***batch processing system***. Job might be interpreted and performed on one server or on the cluster. One job is accomplished in finite time. Often one job is part of large computing process.
- On the cluster there are many jobs from a number of users (customers) at the same time. So we can say about stream of jobs (or **batch**).
- To maximize the cluster usage (maximize the number of accomplished jobs per unit of time)   several ***batch processing systems*** are used.

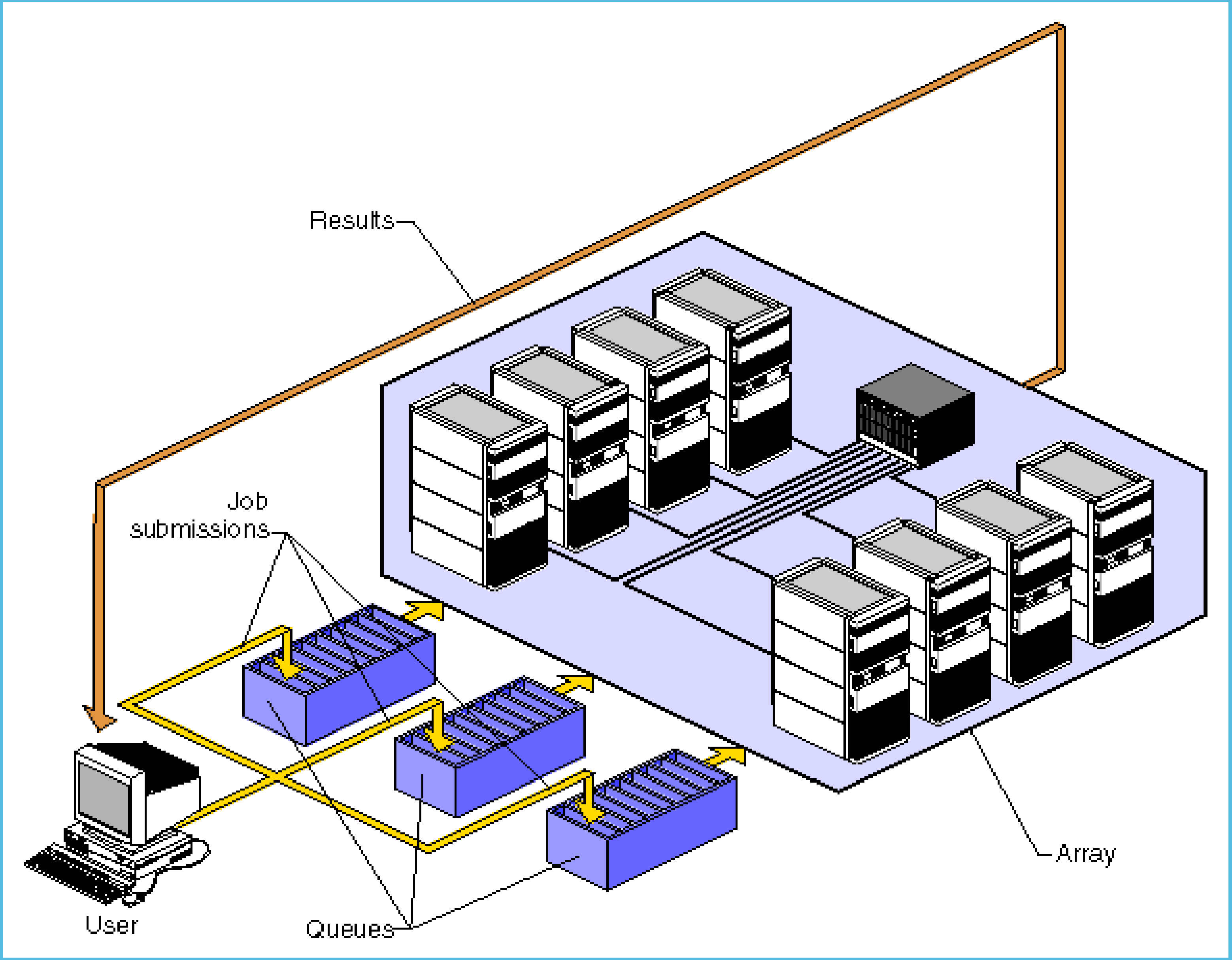
Queues

- If you have more jobs than it is possible to perform at same time you need to put them into ***queue*, *the batch system will get jobs from input* queue *in according another cluster node becomes free***.

Job queues

- Usually jobs enter into batch processing system in one of the possible input queues.
- It has to be rule how the concrete job is entered into concrete queue.
- Also it has to be the rule which input queue is more preferrable in concrete time.

Batch processing system functionality



Examples of batch processing systems

- **PBS/Torque** (http://en.wikipedia.org/wiki/Portable_Batch_System, http://en.wikipedia.org/wiki/TORQUE_Resource_Manager)
- **Condor** (http://en.wikipedia.org/wiki/Condor_High-Throughput_Computing_System)
- **LSF** (http://en.wikipedia.org/wiki/Platform_LSF)
- **SGE** (http://en.wikipedia.org/wiki/Oracle_Grid_Engine)
- **SLURM** - https://computing.llnl.gov/linux/slurm/
- https://en.wikipedia.org/wiki/Comparison_of_cluster_software

Several API for batch processing

- **Qsub – submit the job**
- **Qstat – get the jobs status**
- **Qdel – delete the job from any queue.**

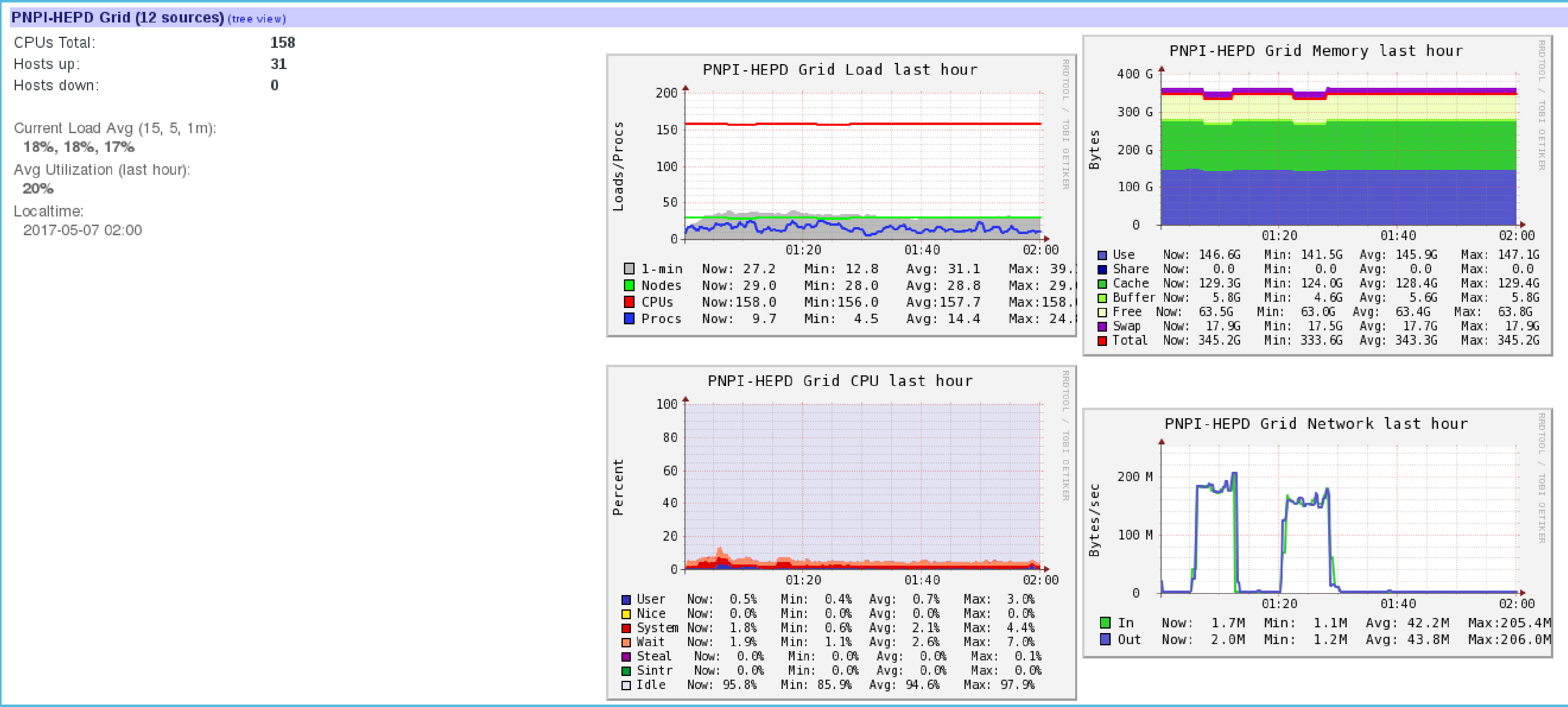
Message Passing Interface (MPI)

- **It is possible to use some library to create the program with parallel parts? YES, it is.**
  - http://www.open-mpi.org/
- **http://cluster.linux-ekb.info/mpi2b.php**

Cluster monitoring

- **Why the monitoring is required?**
  - Ganglia - http://ganglia.sourceforge.net/
  - Zabbix - http://www.zabbix.com/
  - Nagios - https://www.nagios.org/
  -

Ganglia view



PNPI-HEPD Grid (12 sources) (tree view)

CPUs Total:	**158**
Hosts up:	**31**
Hosts down:	**0**

Current Load Avg (15, 5, 1m):
18%, 18%, 17%

Avg Utilization (last hour):
20%

Localtime:
2017-05-07 02:00

PNPI-HEPD Grid Load last hour

```
1-min   Now: 27.2   Min: 12.8   Avg: 31.1   Max: 39.
Nodes   Now: 29.0   Min: 28.0   Avg: 28.8   Max: 29.
CPUs    Now:158.0   Min:156.0   Avg:157.7   Max:158.
Procs   Now:  9.7   Min:  4.5   Avg: 14.4   Max: 24.
```

PNPI-HEPD Grid Memory last hour

```
Use    Now: 146.6G   Min: 141.5G   Avg: 145.9G   Max: 147.1G
Share  Now:   0.0    Min:   0.0    Avg:   0.0    Max:   0.0
Cache  Now: 129.3G   Min: 124.0G   Avg: 128.4G   Max: 129.4G
Buffer Now:   5.8G   Min:   4.6G   Avg:   5.6G   Max:   5.8G
Free  Now:  63.5G   Min:  63.0G   Avg:  63.4G   Max:  63.8G
Swap   Now:  17.9G   Min:  17.5G   Avg:  17.7G   Max:  17.9G
Total  Now: 345.2G   Min: 333.6G   Avg: 343.3G   Max: 345.2G
```

PNPI-HEPD Grid CPU last hour

```
User   Now:  0.5%   Min:  0.4%   Avg:  0.7%   Max:  3.0%
Nice   Now:  0.0%   Min:  0.0%   Avg:  0.0%   Max:  0.0%
System Now:  1.8%   Min:  0.6%   Avg:  2.1%   Max:  4.4%
Wait   Now:  1.9%   Min:  1.1%   Avg:  2.6%   Max:  7.0%
Steal   Now:  0.0%   Min:  0.0%   Avg:  0.0%   Max:  0.1%
Sintr   Now:  0.0%   Min:  0.0%   Avg:  0.0%   Max:  0.0%
Idle   Now: 95.8%   Min: 85.9%   Avg: 94.6%   Max: 97.9%
```

PNPI-HEPD Grid Network last hour

```
In    Now:   1.7M  Min:   1.1M  Avg: 42.2M   Max:205.4M
Out   Now:   2.0M  Min:   1.2M  Avg: 43.8M   Max:206.0M
```

End of lecture