Data intensive ATLAS workflows in the Cloud

Gen Kawamura, Oliver Keeble, Arnulf Quadt, *Gerhard Rzehorz
on behalf of the ATLAS collaboration

Motivation

- Cloud Computing means “renting“ computing resources, different and complex pricing models give the opportunity to achieve a cost reduction
- The resulting flexible hardware gives the possibility to adapt the infrastructure to the workflows (not vice versa)
- Typically increased distance to storage (Cloud activity timeframe makes Cloud storage not worthwhile), this can lead to a significant workflow slowdown
- Slowdown can be mitigated by optimisation techniques (Overcommitting)
- For HEP: benefits of Cloud Computing and optimisation techniques difficult to quantify, impact on workflow performance not well understood
- The Model below solves this, by providing an output metric that quantifies these effects and makes the different scenarios (infrastructures, Clouds, optimisation techniques and workflows) easily comparable

The following sections show, how to evaluate Cloud sites, optimisation techniques and how to estimate workflow performance.

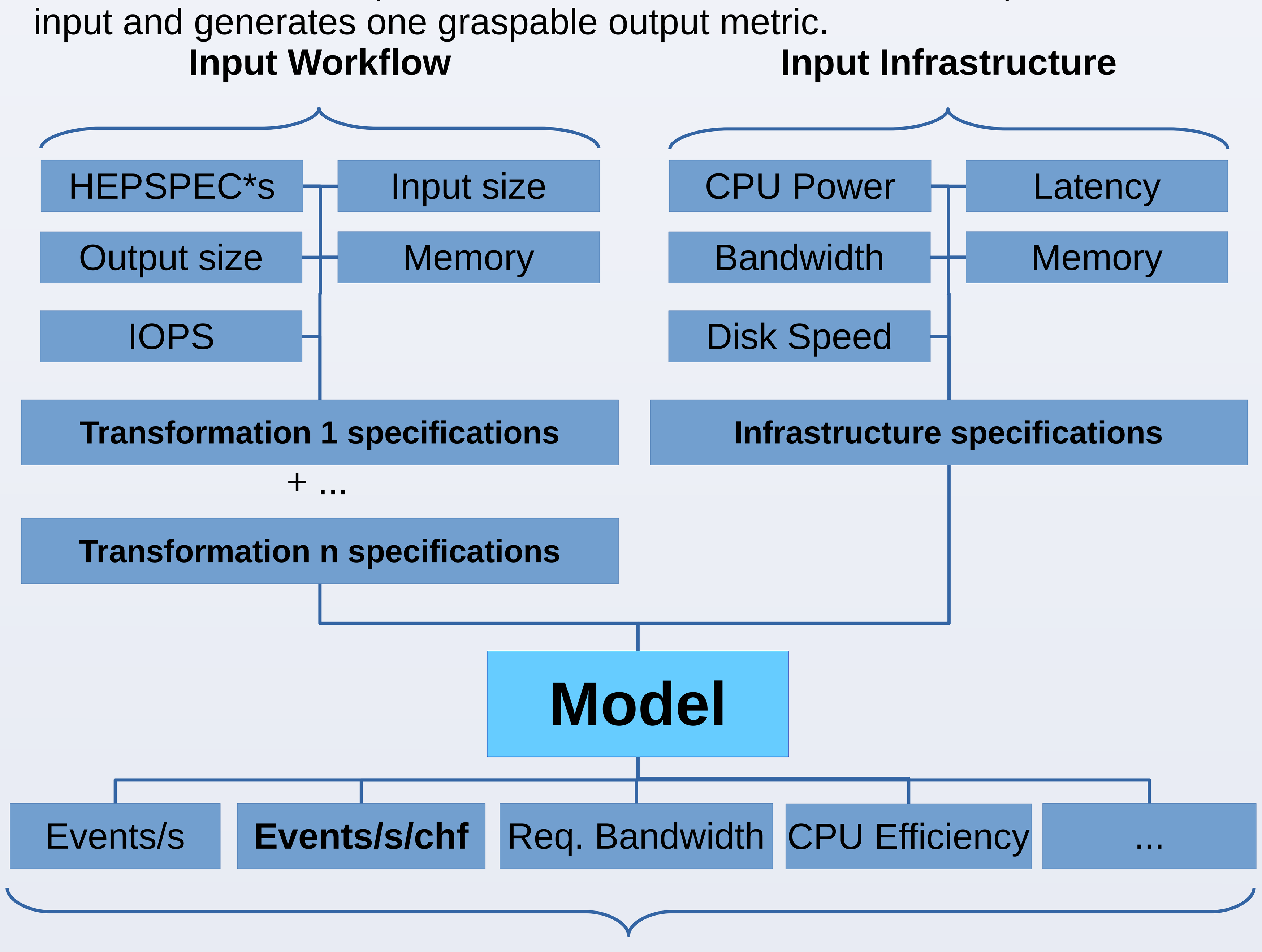
Workflow and Infrastructure Model

Not yet validated, no error estimations

The Workflow and Infrastructure Model solves the following scenarios:

- Evaluate workflow behaviour on a given infrastructure
- Compare different configurations with each other
- Find possible adaptations and optimisations
- Assess the requirements that Grid workflows pose to a Cloud site

The Model takes the plethora of workflow and infrastructure parameters as input and generates one graspable output metric.



Concept:

- Simple Model: Independent input values (HEPSPEC...)
- Modular: Combination of different (Sub-) Workflows
- Generic: all experiments, even outside physics
- Correlations: e.g. How does CPU-time impact bandwidth requirements

Result:

- Common metric: Events/s/chf (“physics“ per time and money)

ATLAS T0 Reconstruction Job Example

- 8 CPU cores
- 32 GB RAM
- Spinning Disk
- AthenaMP
- Remote input data (download beginning)

Modelling the Job characteristics:

- Different Job steps (Diagram: characterised by higher and lower CPU usage) are put into the Model: Start-up, Stage-in and out, CPU Processing, Swapping, I/O activity, Merging, Validation and Cleanup time
- Sum over all the above functions for every step gives the result
- Decreasing RAM leads to swapping activity and slowdown: The Model has a cut there, as the workflow becomes too inefficient
- Remote data streaming slower than download at beginning: either add download time (limited by speed/bandwidth) or add streaming time (event access + download time, limited by latency and bandwidth)
- Significant time spent merging outputs: The Model adds the time

CPU Overcommittment

What it is: Executing more parallel processes than there are CPU cores

Benefits of Overcommitting (OC):

- Reduction of CPU idle time
- Hiding of latency when using remote storage
- Optimisation of cost towards RAM
- Adaptable infrastructure to workflow optimisation

Example of ATLAS T0 Reconstruction run on a dedicated machine on the CERN Openstack cluster, 8 core VM, LHCONE connection:

ATLAS Real Data Reconstruction

Number of processes	RAM [GB]	Data location	Overall node throughput [s/event]	Overcommit improvement [%]	Duration improvement to standard [%]
8	32	BNL	4,19 ± 0,05		-32
2x8	32	BNL	2,55 ± 0,01	39	19
8	16	BNL	4,31 ± 0,08		-36
2x8	16	BNL	3,51 ± 0,08	19	-11
8	32	local	3,07 ± 0,04		3
2x8	32	local	2,24 ± 0,01	27	29
8	16	local	3,17 ± 0,09		0
2x8	16	local	3,33 ± 0,01	-5	-5

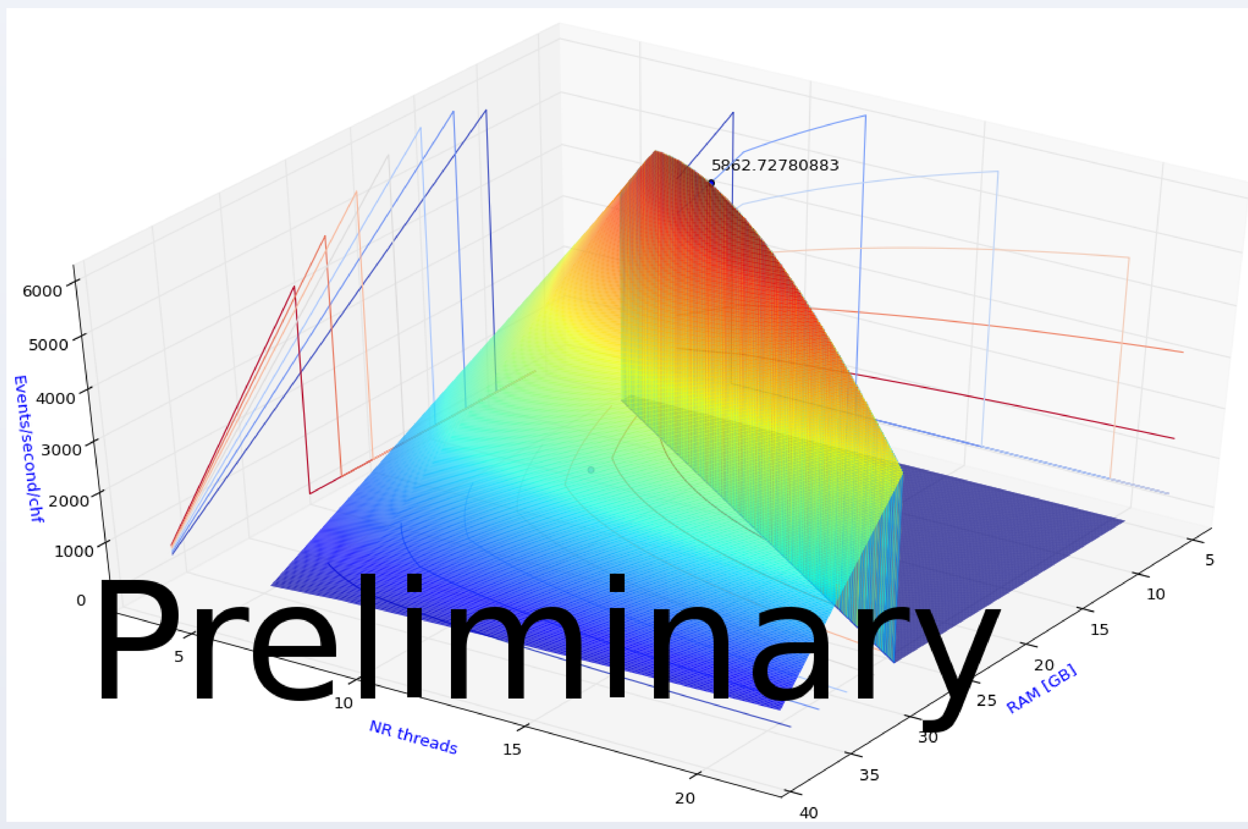
The OC improvement column shows the percentual time saved between the OC and non-OC scenarios. The last column compares the durations to the standard node configuration.

Results show:

- Overcommitting reduces remote data streaming overhead
- Overcommitting is RAM dependent
- Even local data scenario can benefit, given enough RAM
- Cost/Benefit not clear, since RAM also costs - the Model is applied

Model application

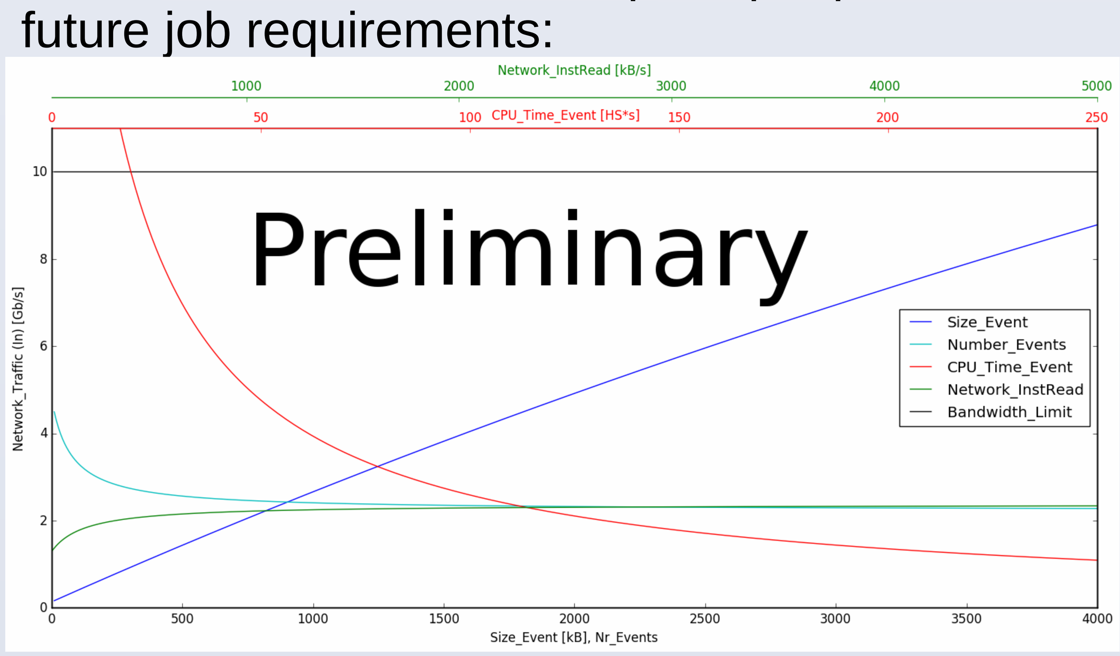
- Overcommittment improvements encouraging, depending on the infrastructure configuration: Additional RAM costs
- Understanding and quantifying the improvement is difficult when also considering the different hardware configurations: apply Model
- With a fixed budget on a particular cost model, what infrastructure configuration gives the best value for money:



Preliminary Result:

- Additional RAM here means less CPUs, example rate from CloudSigma
- Current 2-to-1 ratio of RAM [GB] to CPU [core] is not the optimum, even with the standard 8 parallel threads/machine (4405 Events/s/chf)
- The maximum of 5862 Events/s/chf lies at 14 GB RAM per machine with an Overcommittment of 11 threads/machine

Another application is to predict the overall **bandwidth requirement** of a Cloud site. In addition multiple input parameters are varied to accommodate future job requirements:



- 1000 4-core machines, 116 HS*s/evt CPU time, 850 kB/evt input, 2700 Evt/Job
- 10 Gbit/s are enough
- High impact: CPU time per event, unlikely to change dramatically
- Other parameters variation < 10 Gb/s
- To do: validate with T-Systems Cloud

Further applications:

- Investigate scheduling and caching optimisations
- Find “best“ Cloud provider
- Indicate high impact parameters

Conclusion

- The Model aims at describing workflows on different infrastructures
- It can help choosing the optimal Cloud provider: by choosing the correct infrastructure and also considering the pricing models
- The Model will be able to find and validate new optimisations (e.g. Overcommitting)