



---

# Enabling in-memory Workflow Execution For Large Scale HEP Data Analytics

Syed Ali Zahir Bukhari

Supervised by: Prof. Ashiq Anjum  
Prof. Nick Antonopoulos

# Aim and Objectives

- Aim is to transform and optimize workflows for improving **overall throughput** of HEP data analytics in large scale in-memory clusters.
- Objectives:
  - Setup an in-memory distributed system that can support data intensive analytics
  - Restructure and transform HEP workflows so that they can exploit the in-memory distributed system.
  - Design and implement novel in-memory scheduling and resource management algorithms that can increase the throughput of HEP data analytics.
  - Implement and optimize the proposed resource management and scheduling algorithms in the LCG.
  - Experiment and execute the HEP workflows for demonstrating the improvements in the overall throughput of the proposed system.

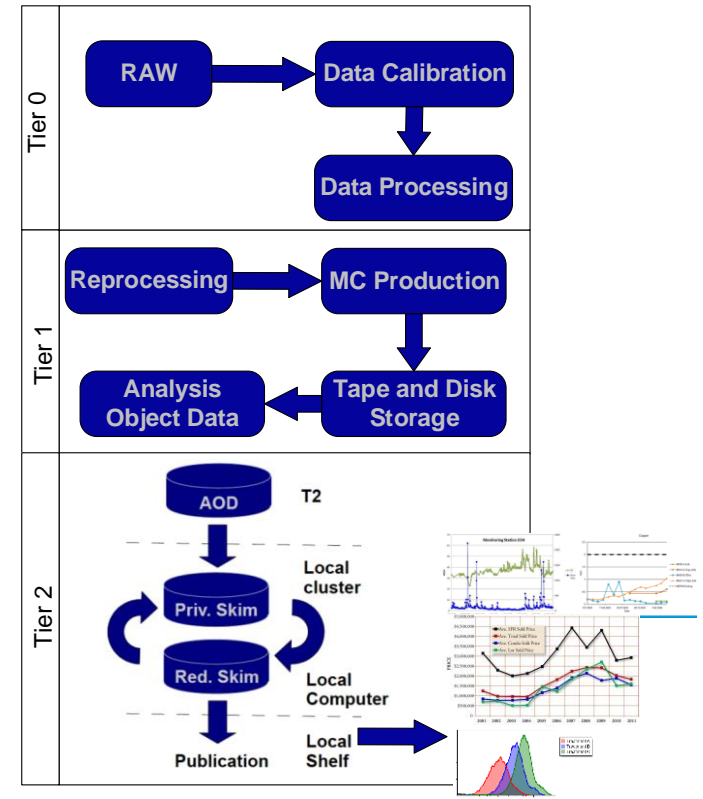
# HEP Data Analytics

- High-Energy Physics (HEP) is often cited as a typical Big Data use case
- The common stages are:
  - Data Filtering
  - Data Mining
  - Data Visualization



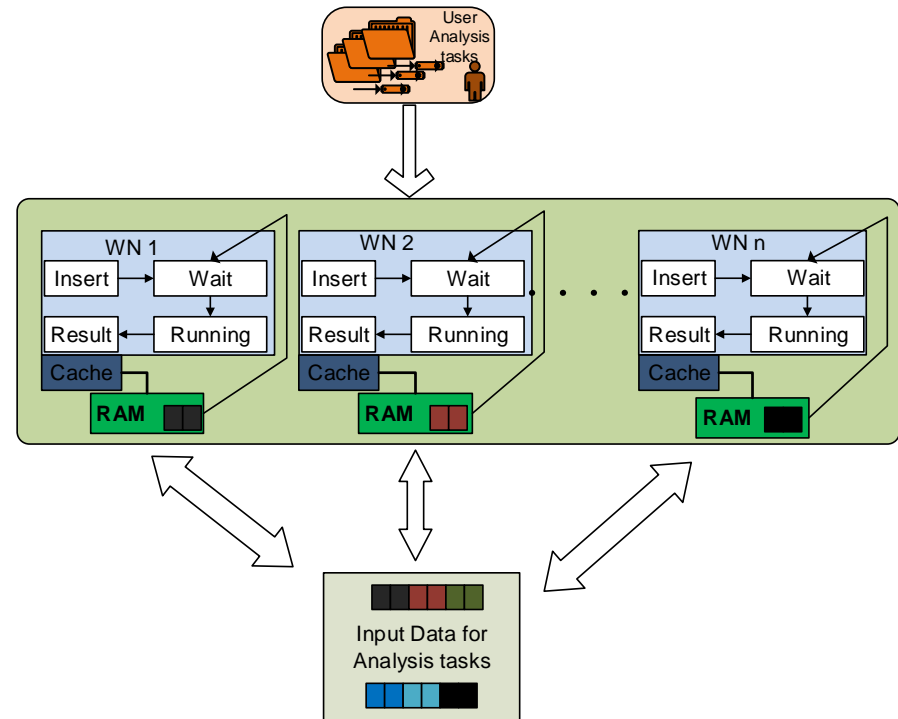
# HEP Data Analytics Workflows

- Data Calibration, Processing and Reconstruction of raw data
- Generation of AOD
- Data skimming
- Data export to local cluster -> local machine
- Visualization/results



# Workflow Execution Process

- Workflows produce and consume large amount of data during their execution.
- Workflow is a set of tasks.
- Each task take some input data
- Input data is divided into small chunks and then loaded into memory.
- Sequential processing



# Example

Analyzing 8GB of input data set  
100 analysis tasks  
80 cores  
160GB of total memory  
56Mbps bandwidth for each core  
**~640 secs**

“Analyze millions of tasks and  
TBs of data take hours”

# Research Challenges and Solutions

- During workflow execution, memory has to load/unload data from storage and vice versa.
- Large volume of generated data
- Shortage of memory due to the size of input data.
- Latency issues

Problem



How to

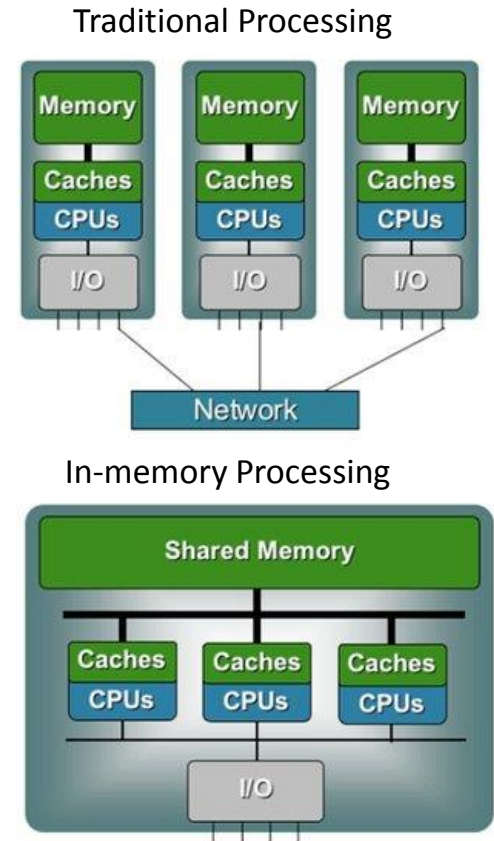


- In-memory distributed system
- Reconstruct HEP workflows
- Design in-memory RMS algorithms

solution

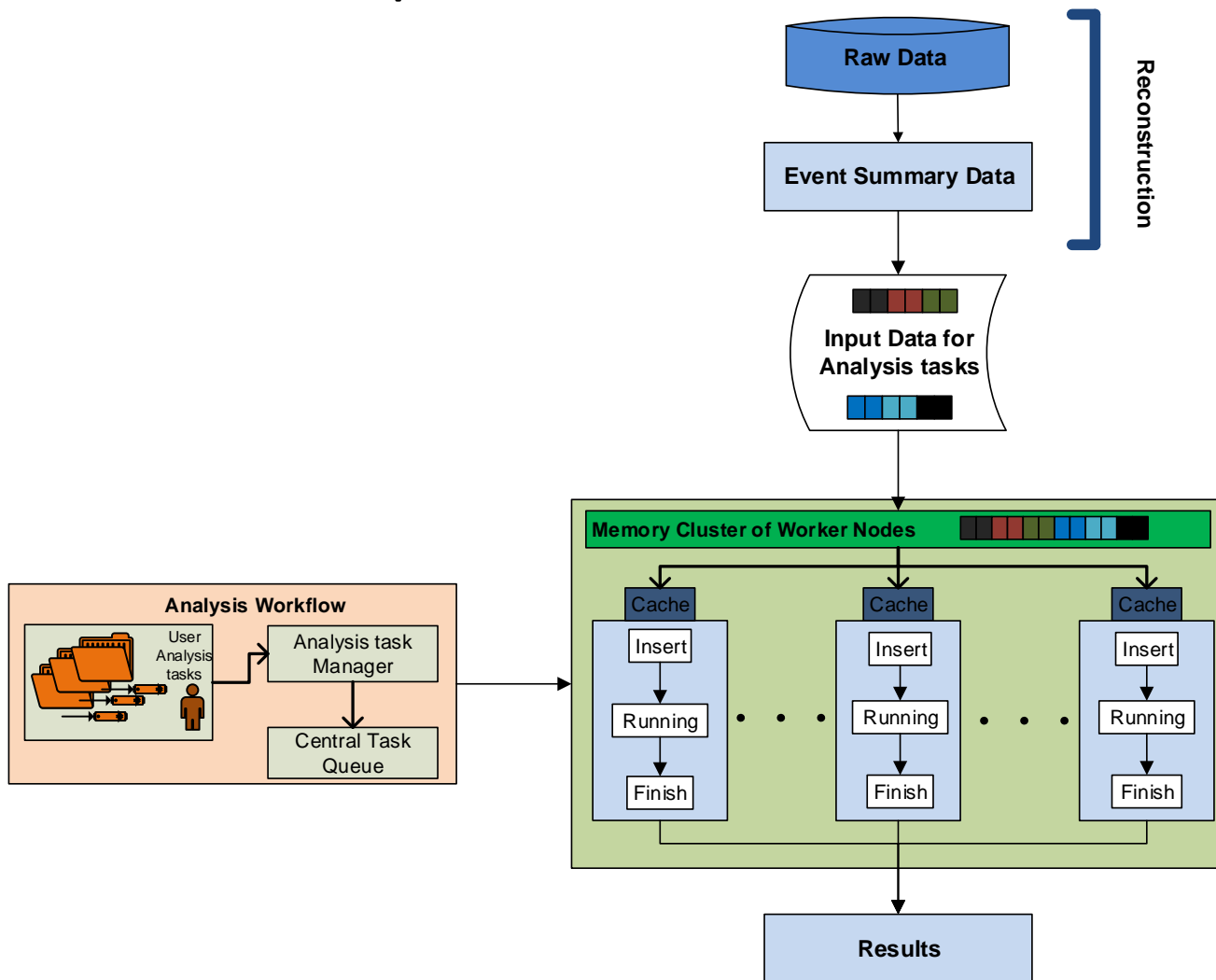
# In-memory Execution

- Memory is hundred thousand times faster than any storage devices.
- In-memory clustering
- In-memory storage systems stores generated data in DRAMs





# Proposed Solution



# Benefits

- Minimize the data relocation between Memory and storage.
- Significantly reduce the data access time
- Maximize the CPU utilization
- Increase in overall throughput of analytics

# Tools & Technologies

- Apache Flink
- Apache Spark
- Hadoop
- ROOT/AlIROOT

# Conclusion

- Our aim is to transform and execute workflows using large scale in-memory clusters.
- It will improve the overall throughput of HEP data analytics.
- Minimize data movement
- Optimize processing efficiency
- Using in-memory computing for HEP analytics workflows.

# References

- Aamodt, K., Quintana, A.A., Achenbach, R., Acounis, S., Adamová, D., Adler, C., Aggarwal, M., Agnese, F., Rinella, G.A., Ahammed, Z. and Ahmad, A., 2008. *The ALICE experiment at the CERN LHC*. Journal of Instrumentation, 3(08), p.S08002.
- ALICE Collaboration. *Upgrade of the ALICE Online - Offline computing system, Technical Design Report*. CERN-LHCC-2015-006/ ALICE-TDR-019, 2015
- Bhat, P.C., 2011. Advanced analysis methods in particle physics. *Annual Review of Nuclear and Particle Science, volume, 61*, pp.281-309.
- Russo, S.A., Pinamonti, M. and Cobal, M., 2014. *Running a typical ROOT HEP analysis on Hadoop MapReduce*. In Journal of Physics: Conference Series (Vol. 513, No. 3, p. 032080). IOP Publishing.
- A. Uta, A. Sandu, S. Costache, and T. Kielmann, “Scalable InMemory Computing,” in 2015 15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), 2015, pp. 805–810
- Russo, S.A., *Using the Hadoop/MapReduce approach for monitoring the CERN storage system and improving the ATLAS computing model* (Doctoral dissertation, Udine U.).

# Questions??