

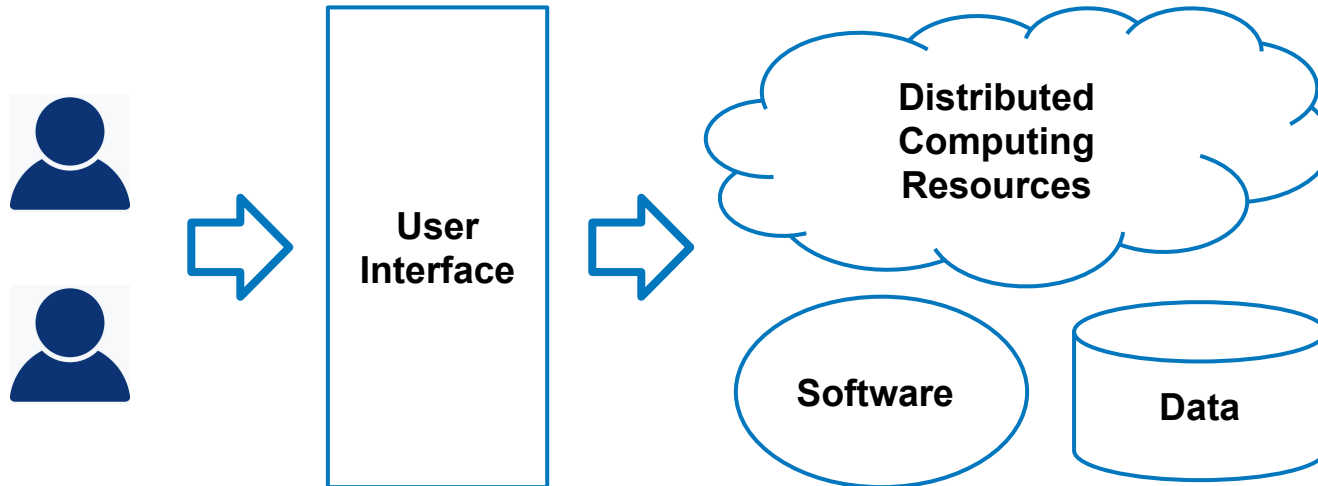
# Interactive Distributed Analysis of HEP Data

E. Tejedor, X. Valls for the **ROOT** Team and J. Cervantes



- ▶ Successful model of distributed analysis in HEP so far
  - Batch / Grid jobs
- ▶ What is not so great about it
  - **Wrapper code** / scripts necessary to perform submission
  - **Not interactive**: cannot view intermediate application results
  - **Hard to monitor** progress / estimate time to completion
  - **Hard to debug** and analyse performance
- ▶ As soon as they can, physicists work on ntuples that fit in their computer
  - Simpler, more user-friendly, interactive, full control of the resources
  - Will this still be possible in 10 years?

- ▶ A scalable *Interactive Distributed Analysis Environment*
  - **Easy entry** point to distributed execution
  - Pluggable computational **HPC/cloud** resources
  - Leverage on **big data** technologies from industry



- ▶ High-level programming model
  - Declarative
  - Little / no changes to go distributed
  - Interactive, easily modifiable
- ▶ Monitoring
  - View current status
  - Inspect intermediate results
- ▶ Debugging
  - Check locally whatever you can, automatically
  - Enable debugging and diagnosing (distributed) failures
- ▶ Web-based
  - No SSHing, no job submission to a queue, no local installation
  - Just a browser

- ▶ **Computing resources**
  - Pluggable, scalable, elastic
  - Selected via UI, feedback from resource manager
- ▶ **Software**
  - Same environment in client and execution nodes
  - Support custom software from the user
- ▶ **Data**
  - Data on EOS: ingestion is costly
  - Minimize cost of reading remotely (caching, prefetching)
  - Easy upload/download to/from the Analysis Environment
    - Automatic synchronization

# Backup slides



# What others are doing

---

- ▶ IBM Data Science Experience
  - <https://datascience.ibm.com/>
- ▶ Microsoft HDInsight
  - <https://azure.microsoft.com/en-us/services/hdinsight/>
- ▶ Google Colaboratory
  - <https://colab.research.google.com>