***Abstract*—Clusters make use of workload schedulers such as the Slurm Workload Manager to allocate computing jobs onto nodes. These schedulers usually aim at a good trade-off between increasing resource utilization and user satisfaction (decreasing job waiting time). However, these schedulers are typically unaware of jobs sharing large input files, which may happen in data intensive scenarios. The same input files may be loaded several times, leading to a waste of resources.**

We study how to design a data-aware job scheduler that is able to keep large input files on the computing nodes, without impacting other memory needs, and can use previously loaded files to limit data transfers in order to reduce the waiting times of jobs.

We present three schedulers capable of distributing the load between the computing nodes as well as re-using an input file already loaded in the memory of some node as much as possible.

We perform simulations using real cluster usage traces to compare them to classical job schedulers. The results show that keeping data in local memory between successive jobs and using data locality information to schedule jobs allows a reduction in job waiting time and a drastic decrease in the amount of data transfers.

***Index Terms*—Job input sharing, Data-aware, Job scheduling, High Performance Data Analytics**