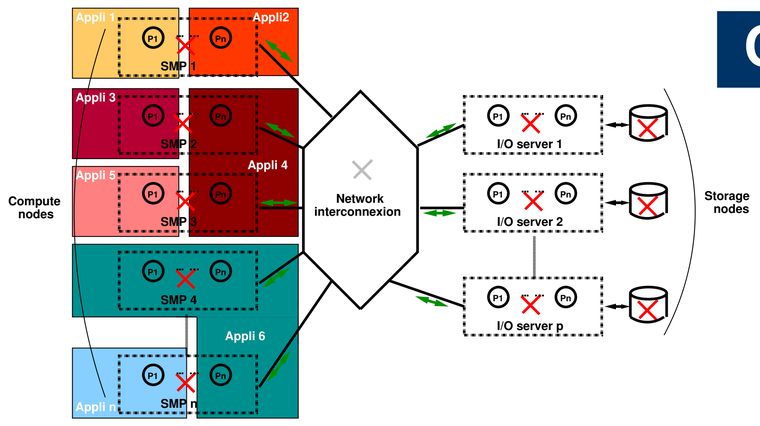
Laboratoire Informatique et Distribution

I/O Scheduler for High Performance Computing

Cluster–Wide Adaptive I/O Scheduling for Concurrent Parallel Applications

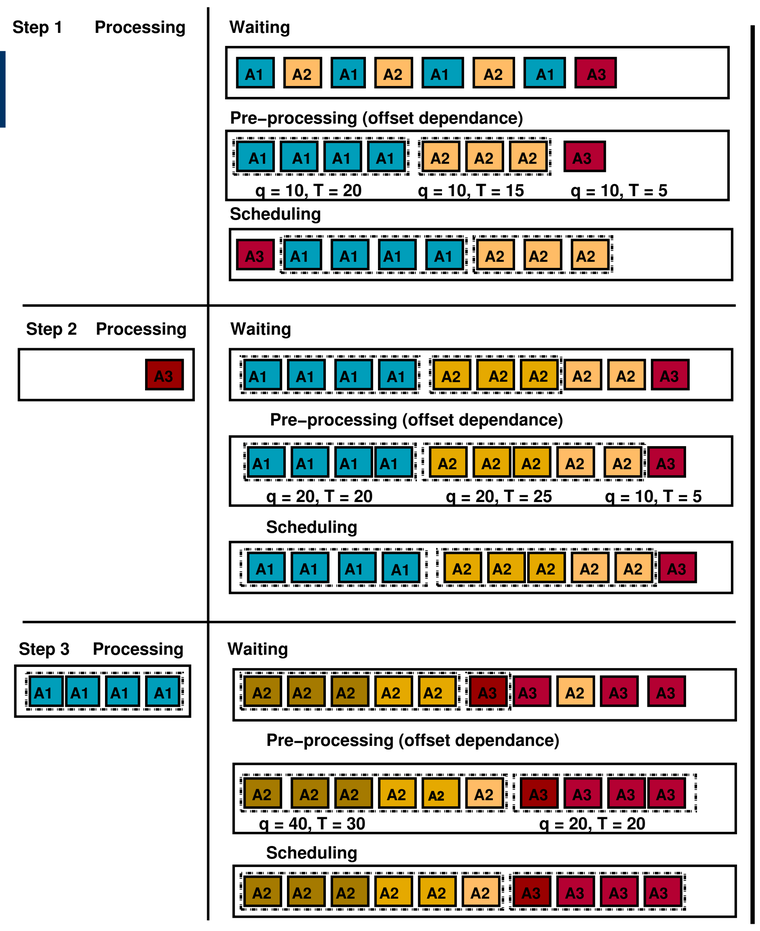


Context

- High Performance Computing
- Clusters are multi–applicative environments
- Scientific applications become more and more I/O demanding
- Parallel I/O libraries are focused on one application
- File systems are mainly based on low level schedulers

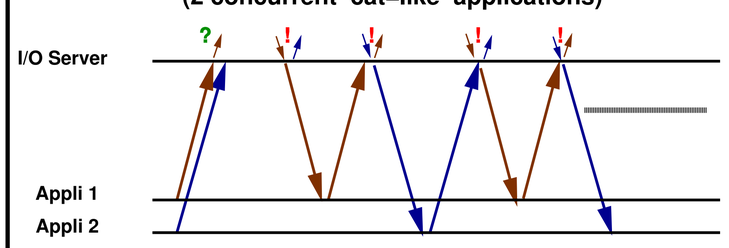
Multi–applicative scheduling

- Concurrent "parallel I/O" applications
  - Aggregate incoming requests from a same application
  - Balance storage access ("fairness") between the running programs
  - A quantum based algorithm (MLF), one quantum per application grows at each step →

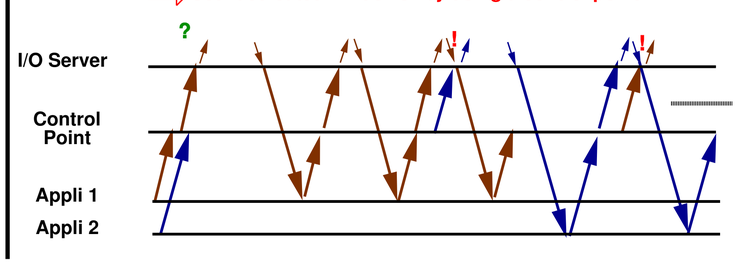


- Synchronous behavior
  - Benefit from "read–ahead" strategy (for all levels)
  - Reduce seeks on hardrive

Synchronous behavior
(2 concurrent 'cat–like' applications)

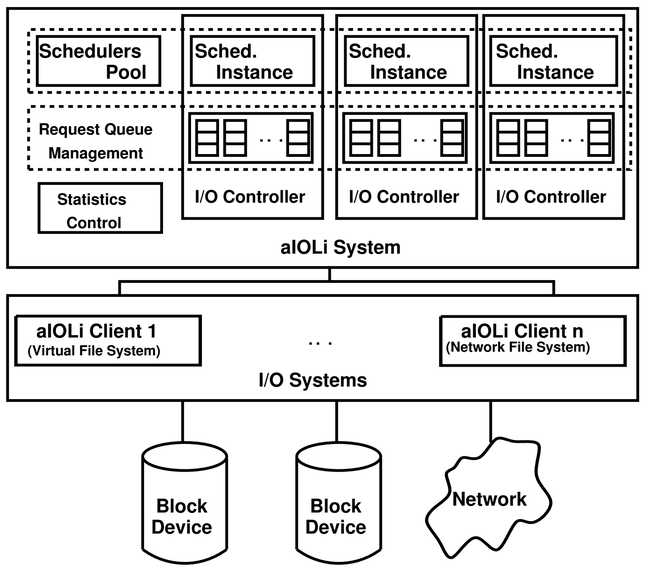


Non efficient usage of read–ahead strategy
Potential head seek on the hard–drive
⇒ Set "dedicated windows" by using a control point

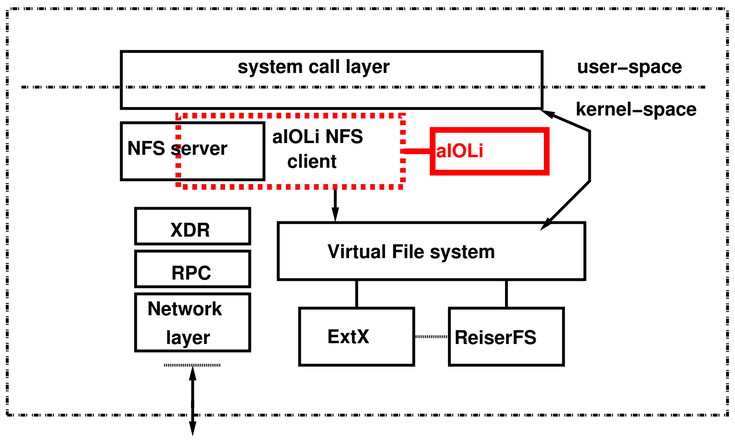


Implementation

- Independent of the applications
- A linux module pluggable into any I/O systems



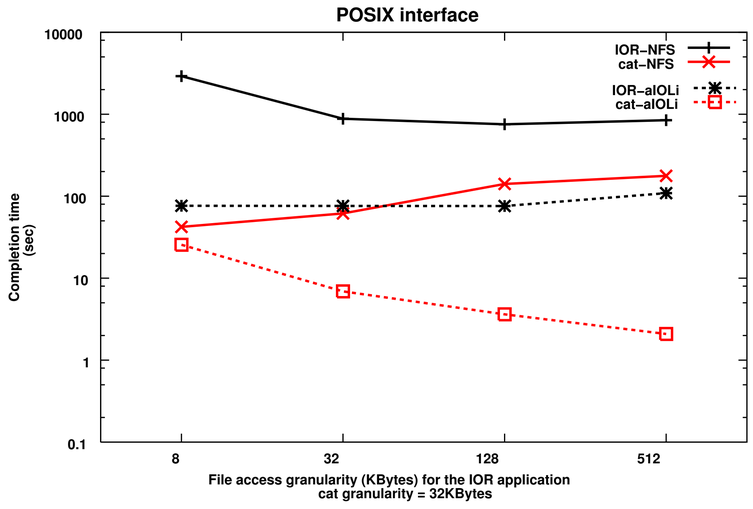
- File systems, acting as clients, use the aIOLi scheduling service: as an example, an NFS server



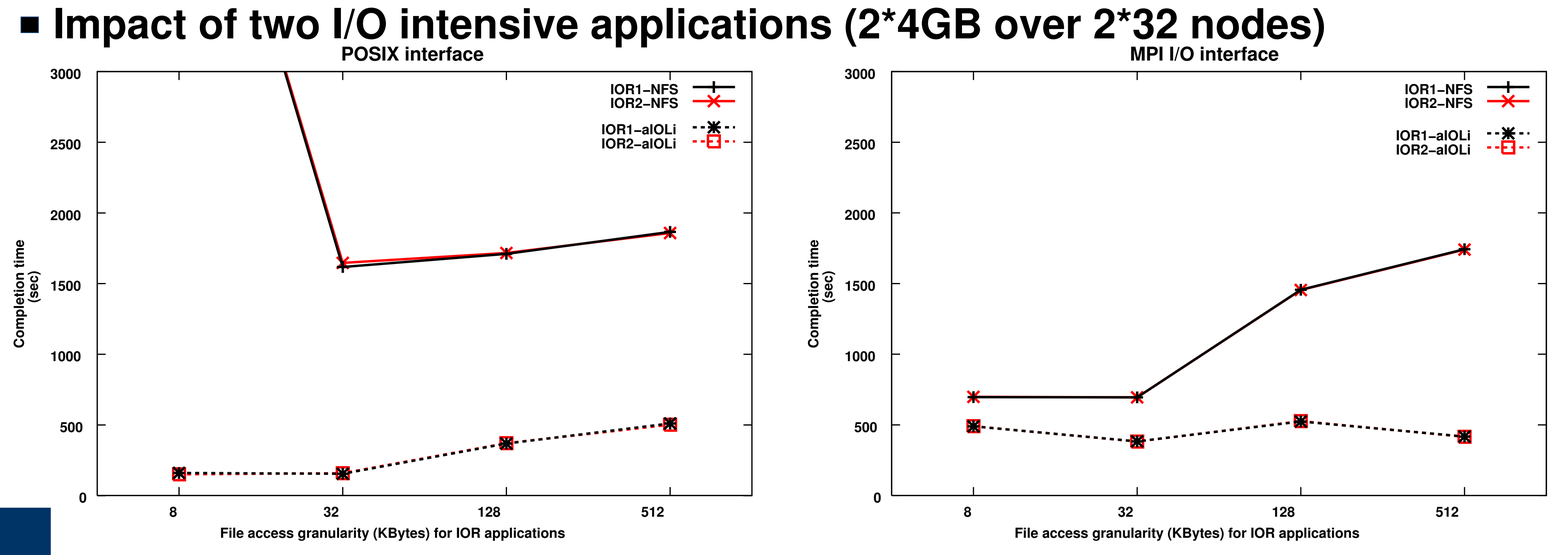
aIOLi plugged to an NFS server

Experiments

- Sophia–antipolis GRID5000 plateform, Linux 2.6.15, NFS v3, mpich 1.2.5 (ROMIO) and IOR benchmark (from LLNL)
- Impact of an I/O intensive application (4GB over 32 nodes) on a small "cat–like" program (16MB from one node)



- Impact of two I/O intensive applications (2*4GB over 2*32 nodes)



Conclusion

- Preliminary results are excellent ! The more applications, the better the results
- Add aIOLi in several points to reduce congestion issues
- Evaluate aIOLi in presence of parallel FS

Adrien Lebre
Yves Denneulin
Guillaume Huard
ID–IMAG Lab (FRANCE)

http://aioli.imag.fr

Sowa Przemyslaw
ICIS (POLAND)

June 2006