Advance Reservations
✓ QoS guarantee
✓ Co-allocation
✓ Predictability

Draw backs
x Low Utilization
x Rigidity

A Naïf Approach

Efficient Algorithm

• Key features
  • Idle periods are stored in a binary search tree
  • Idle periods are stored in the leafs
  • Intermediate nodes store information, e.g., median values and help on navigating the tree
  • There is one tree for each single grid slot

Complexity
Due to the use of binary search trees the algorithm runs at $O(\log n)$ where $n$ is the number of servers

Simulation Results

Conclusions
• Advance reservations can be provided without hurting system performance.
• Our work provides new insight and allows for efficient organization of reservations.
• Scheduling algorithm that have good performance that can scale up to large systems thanks to the logarithmic bound of the algorithm.

Future work
• Co-Allocation of resources
• Uncertainty of execution and arrival times of jobs
• Negotiation mechanisms to maximize utilization