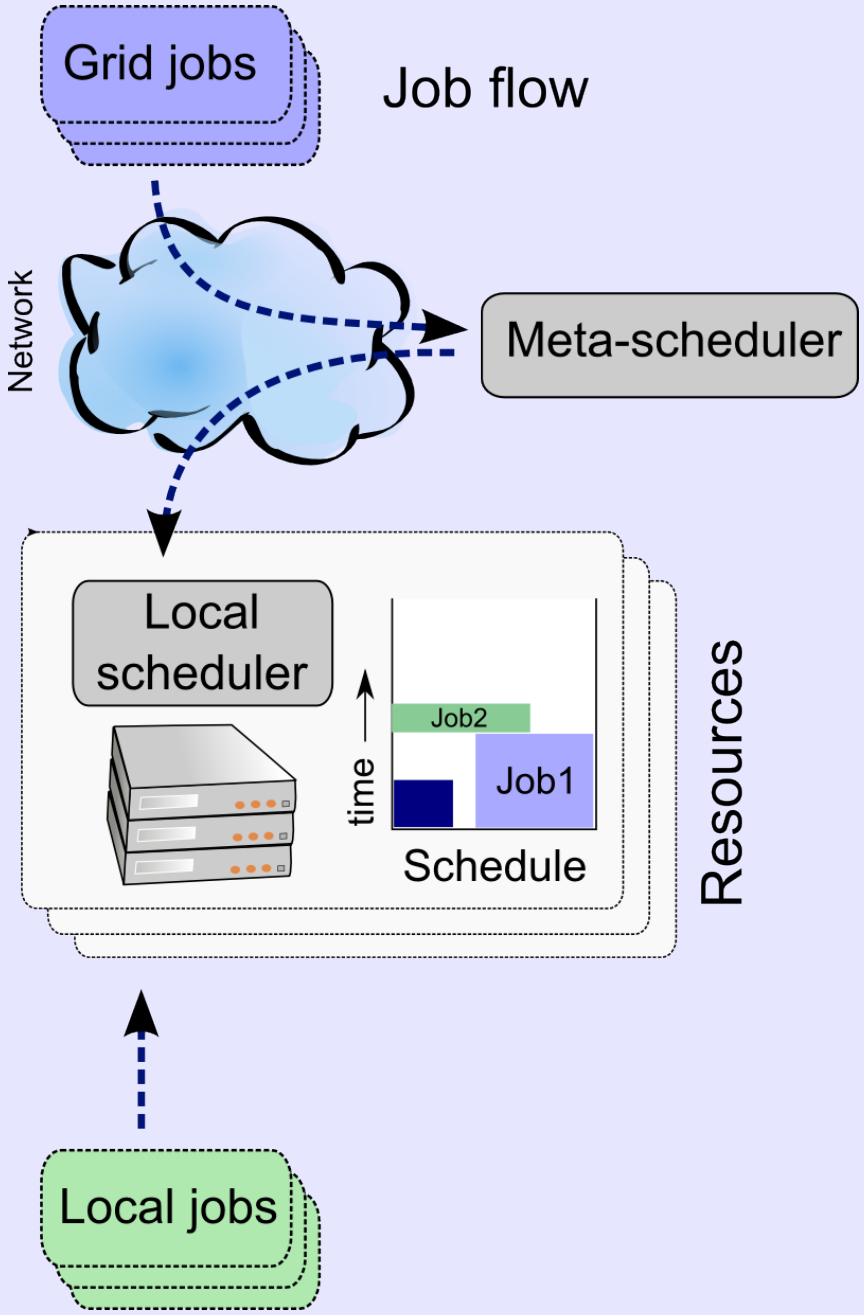
MODELING GRID BEHAVIOR USING WORKLOAD DATA

Grushin D., Kuzjurin N., Pospelov A., Shokurov A.
{grushin,nnkuz,ap,shok}@ispras.ru

Institute for System Programming, RAS

Scheduling computational jobs in Grid



- Each job is represented as a rectangle:
  - Width – number of requested processors
  - Length – requested execution time.
- Meta-scheduler has no direct control over resources

Multi objective optimization problem

- Minimize Average Wait Time
- Maximize throughput – total number of completed jobs
- Minimize maximum total processing time - makespan
- Minimize cost of execution (economic-based resource management)
- etc.

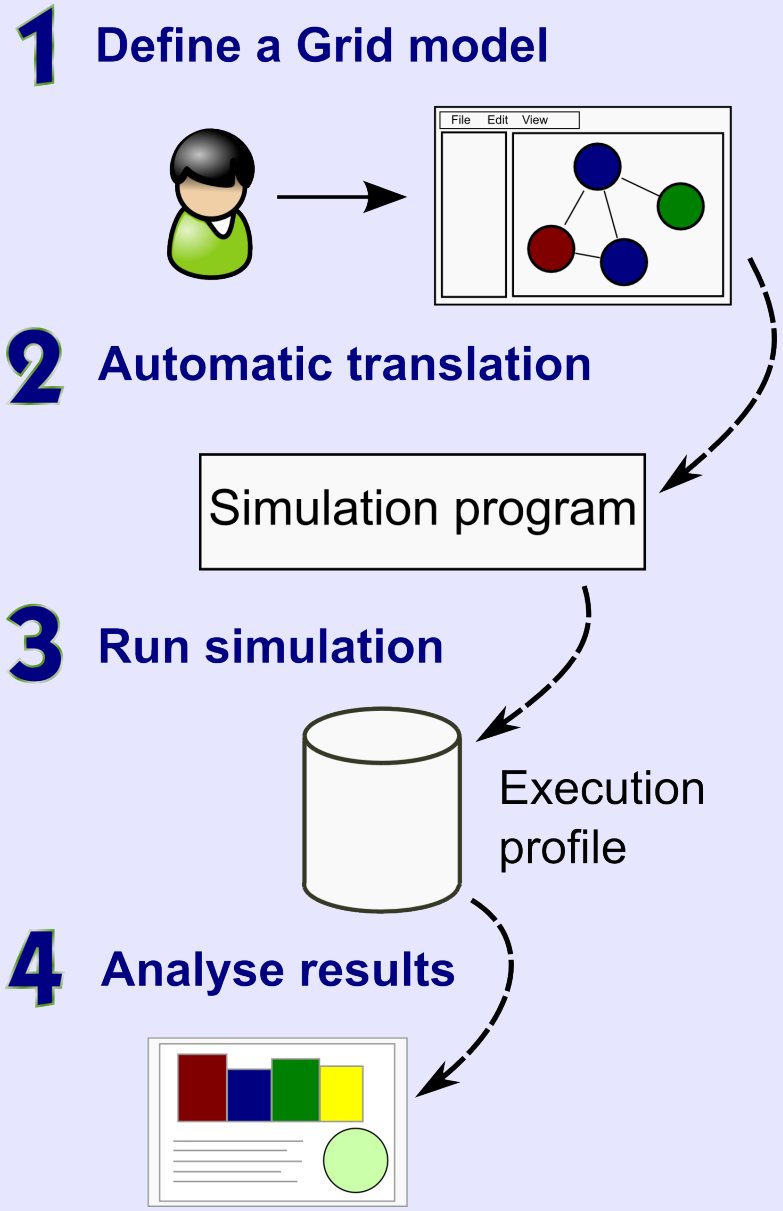
Known Grid simulators

- Bricks (1999)
- OptorSim (2002, last version 2006)
- SimGrid (1999, last version 2007)
- GridSim (2001, last version 2007)
- GSSIM (2007)
- Grid Value at Work (2003)
- Delft Grid Simulator – DGSim (2007)
- GangSim, ChicSim, etc.

Grid modeling environment developed in ISP RAS

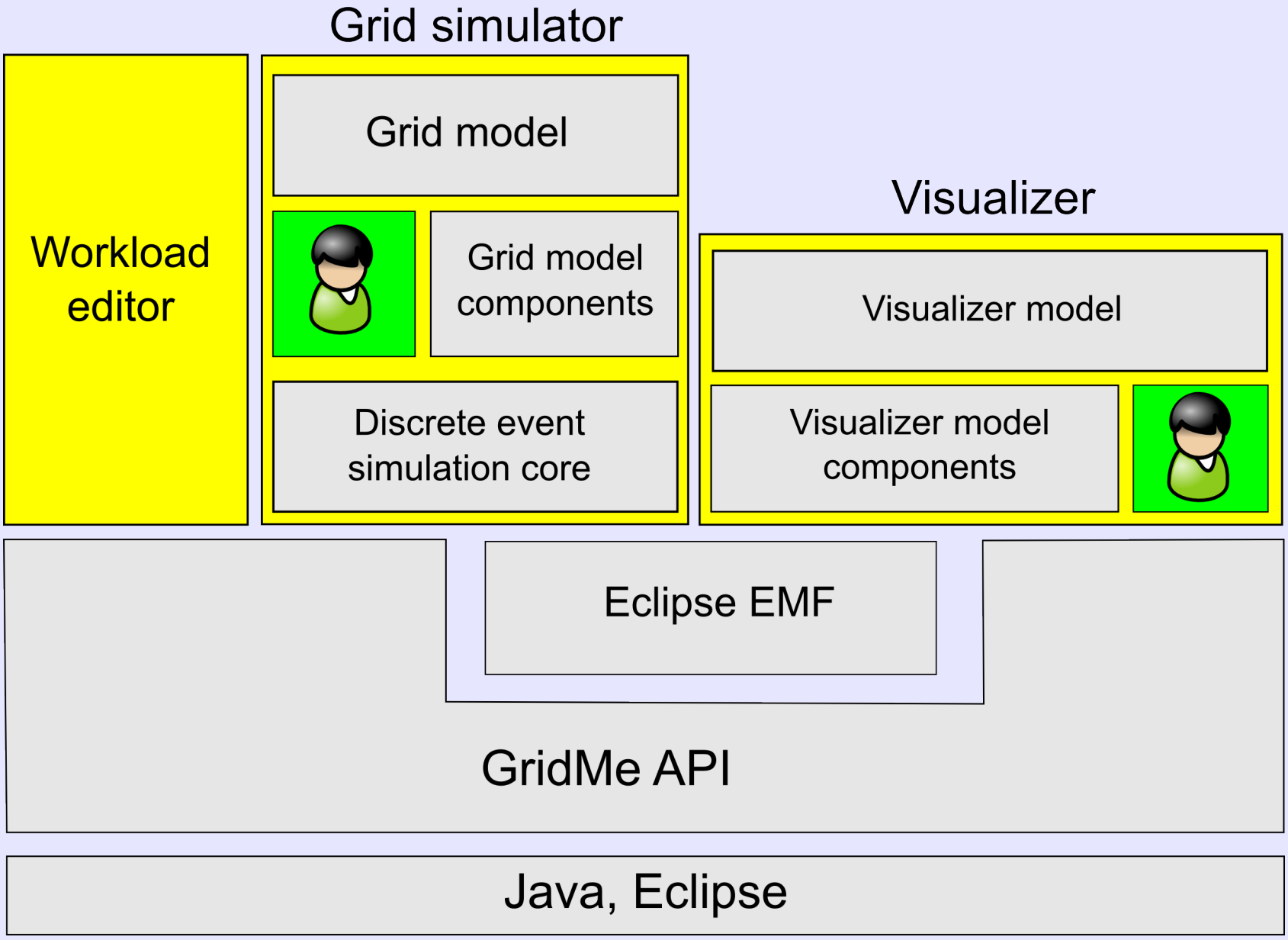
- Scalable discrete event simulator
  - ~10^3 processors, ~10^6 tasks
- Implemented in pure Java and based on Eclipse platform
- Configurable
  - Can simulate different Grid resource management architectures
  - Custom pluggable elements
- Provides simulation results processing tools
- Includes workload analyzer and editor

Common use case scenario



- No manual programming tasks required
- Use model editor to define:
  - Topology
  - Grid element properties
- Check errors on the fly

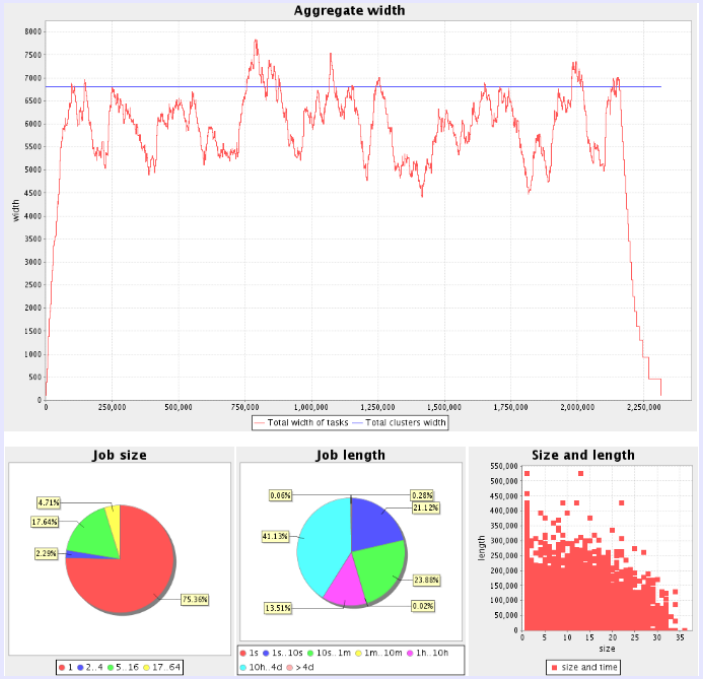
System architecture

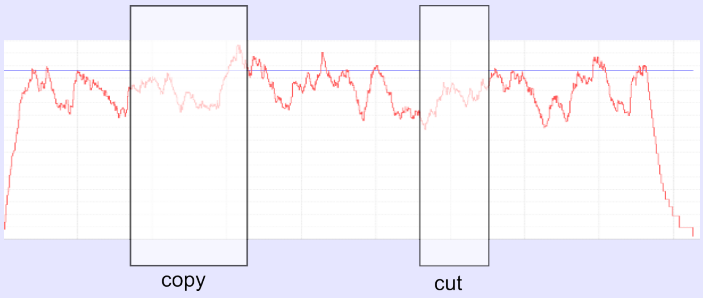


Implemented pluggable components

- Cluster
- Broker
- Network connection
- Local scheduler
  - FCFS, BestFit, Backfill
- Meta scheduler
  - RandomFit, BestFit, MCT, AverageLoad, etc.

Workload analyzer and editor





- Visualization of workload characteristics
  - Job size and length distribution
  - Aggregate width over time
  - Etc.
- Generate synthetic workload
- Edit several workload files
  - Cut, copy
  - Paste: over, into
  - Filter
  - etc.

Typical problems that can be solved with the Grid simulator

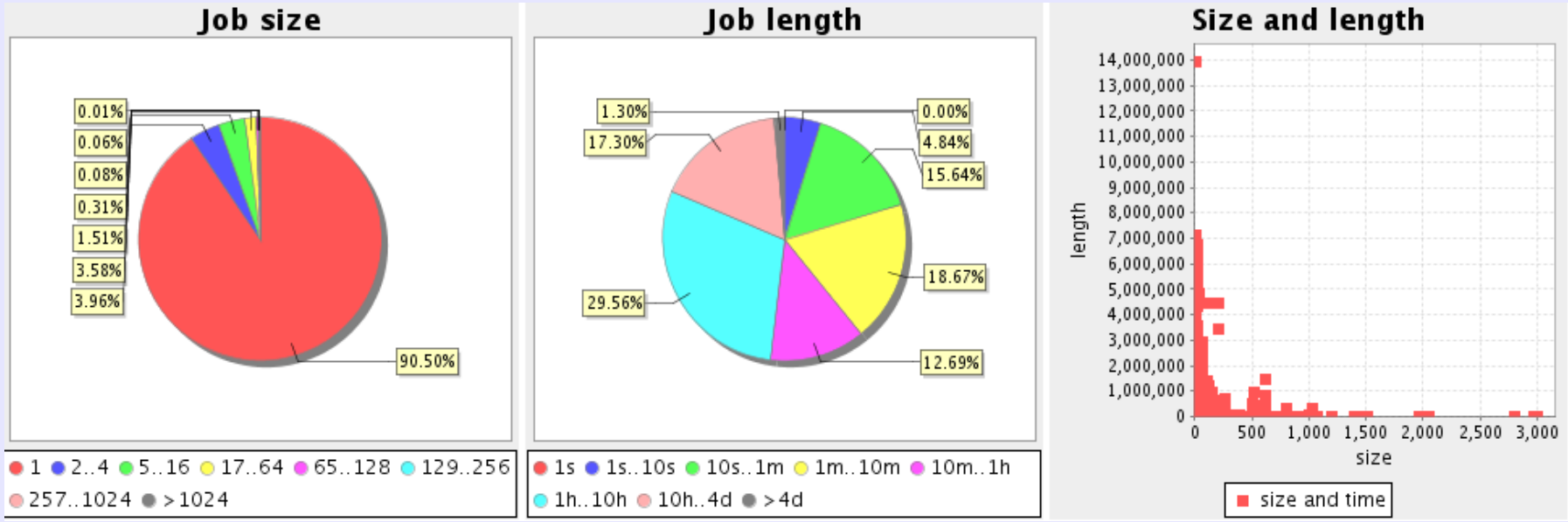
- Predict Grid system performance under various changes:
  - Different workloads
  - System configuration
  - Different scheduling heuristics
- Tune scheduling parameters and find better scheduling solution

Example of Modeling: Sharcnet
(Shared Hierarchical Academic Research Computing Network)

- Total 6828 Opteron processors
- Workload used
  - 1,195,242 jobs
  - December 2005 - January 2007
- No global scheduler – users manually choose clusters for their tasks

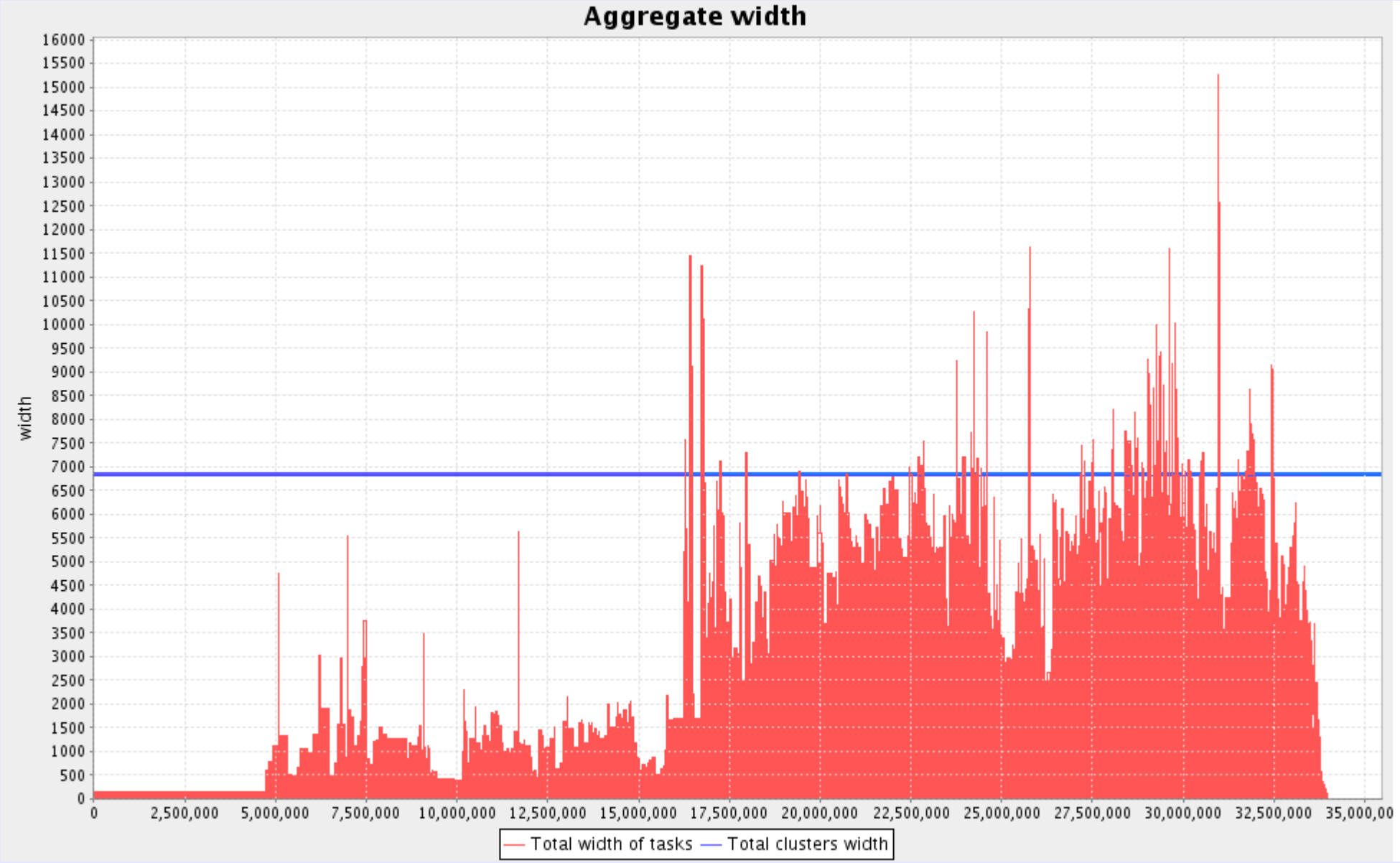
Cluster	Size
bruce	128
narwhal	1068
tiger	128
bull	384
megaladon	128
dolphin	128
requin	1536
whale	3072
zebra	128
bala	128

Sharcnet workload analysis

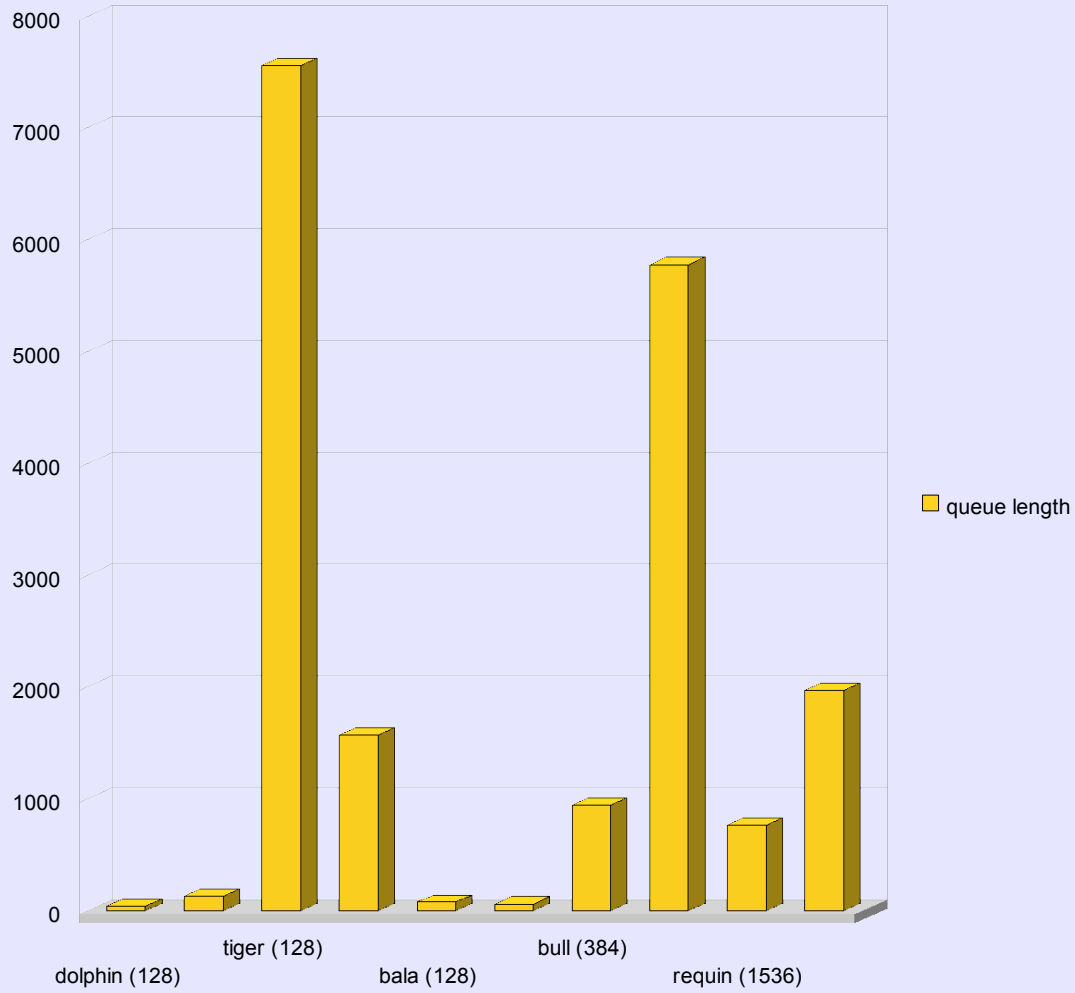


- 90% of 1-CPU tasks
- Big parallel tasks are not very long

Sharcnet workload analysis



Queue imbalance in Sharcnet



- Significant load imbalance exists
- Wait time for clusters differs significantly (up to 20 times)
- Can global scheduling help?

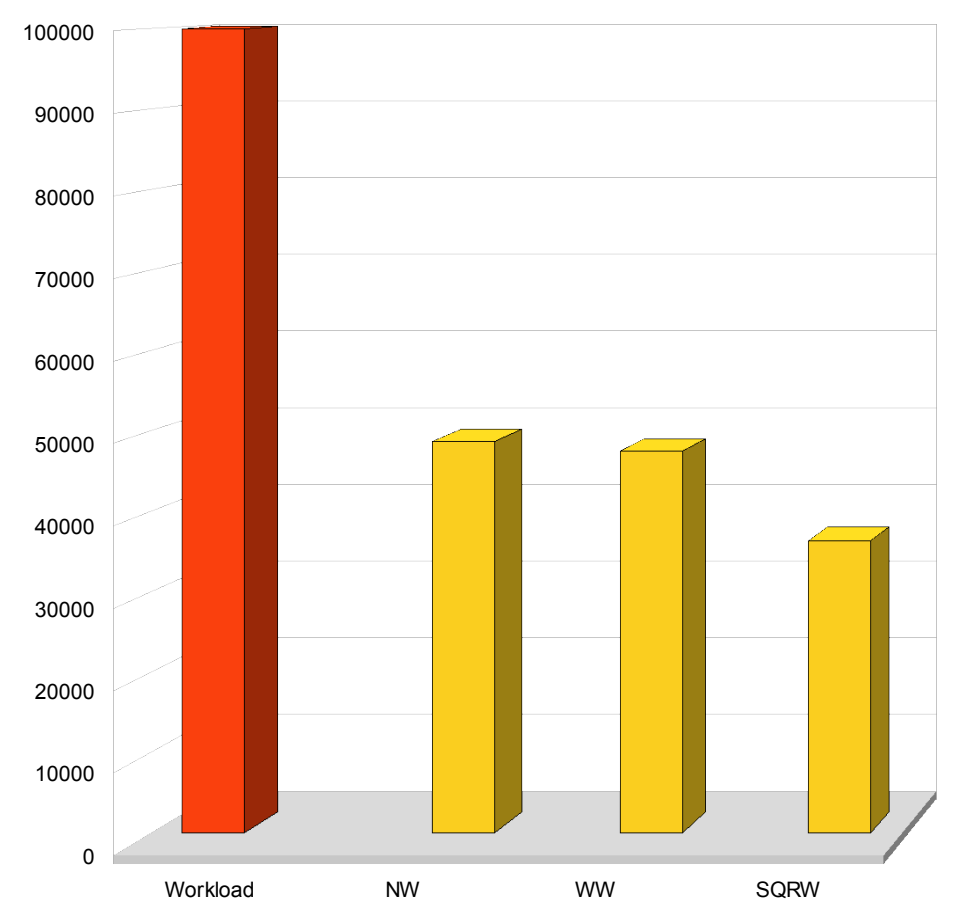
Experiments

- No broker, use workload resource assignment
  - Local scheduler uses backfill
- With broker
  - Heuristics: N/W, W/W, Sqr/W
  - Schedule all tasks
  - Schedule only 1-CPU tasks
- Workloads
  - Original
  - Synthetic
- No network overhead

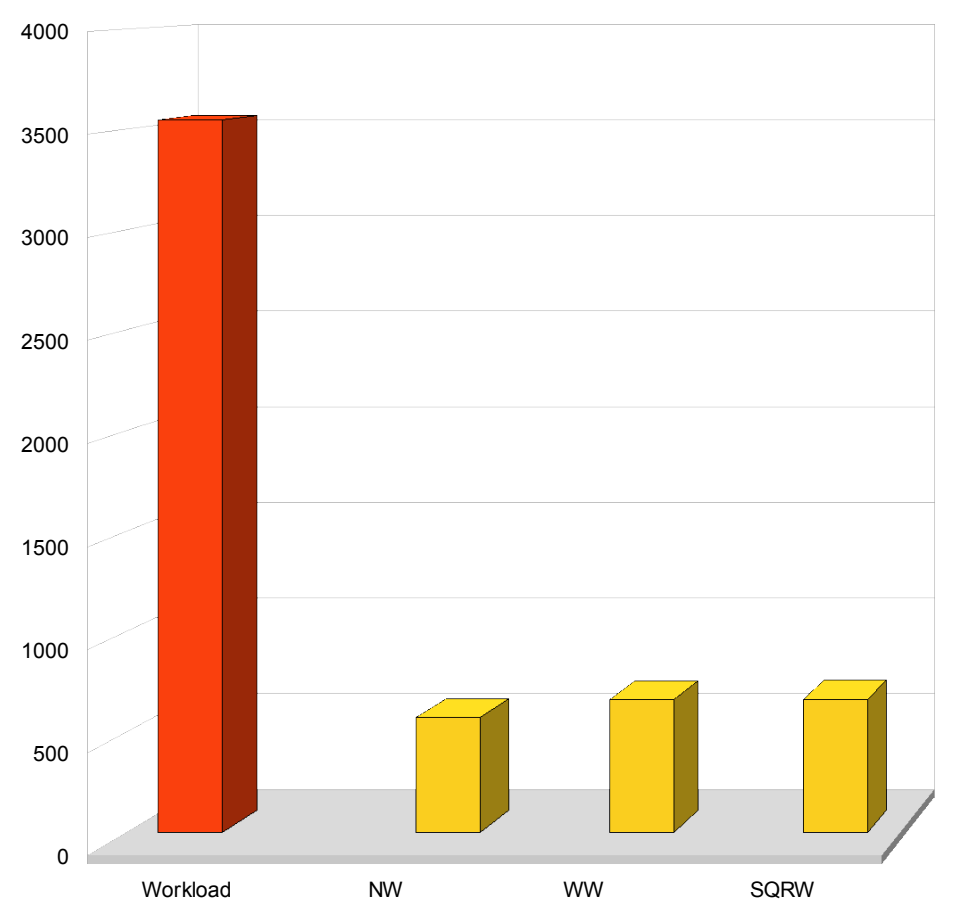
Evaluation criteria

- Average wait time in queue
- Queue – average and on each cluster:
  - length
  - square

Results



Average wait time (seconds)



Average queue length

Conclusion

- Experiments with Sharcnet show significant performance increase for the system with broker – about 130%
  - Similar results for Grid5000 and DAS2: A. Iosup et al. Inter-Operating Grids through Delegated MatchMaking, 2007
- Heuristics behavior depends on the workload characteristics and differs significantly
- Modeling should be used to evaluate the behavior of real computing system

Future work

- Full-featured performance evaluation tool
  - Possible integration with monitoring tools
  - Scheduling heuristics library
  - Usability enhancements
- More experiments with data transfer
- Workload editor enhancement
  - Synthetic workloads

Questions?

- Our system is available at http://gridme.googlecode.com/