

# Getting Your Work Done in Grids, Politely

Ozan Sonmez, Hashim Mohamed, Bart Grundeken, Alexandru Iosup, and Dick Epema (TU Delft, NL)

 {O.O.Sonmez,H.H.Mohamed,A.Iosup,D.H.J.Epema}@tudelft.nl

## Rewarding Application

- **Eternity II: The puzzle with a \$2,000,000 prize**  
Place 256 pieces on a 16x16 grid board. Grey edges go along the border of the board, colors and symbols on adjacent pieces have to match. Best found solution for this board gets \$2,000,000.
- **Solving Eternity as a (typical) parameter sweep application**  
A wide range of solving methods, from brute force to artificial intelligence, can be applied. Automate the execution of thousands of jobs in the DAS as a Parameter Sweep Application. Get notified when the current sweep finishes.
- **Getting to the best solution**  
Best solution so far: over 90% of the pieces placed.

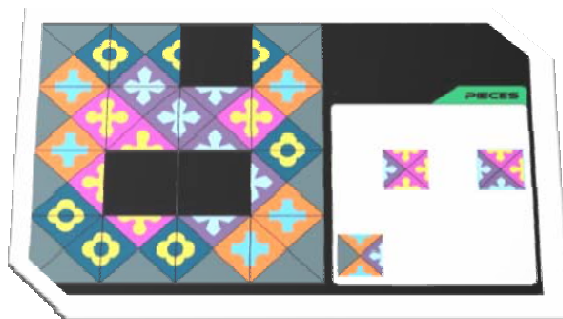


Figure 1: A sample Eternity Puzzle with 4x4 board. Placed so far 13 out of 16 pieces.



Figure 2: Fair-Sharing Policies for Cycle Scavenging.

## Politeness

- **Cycle scavenging (CS) for adaptive use of any spare capacity**  
Exploit otherwise wasted computational power, but **be polite** and preempt CS jobs whenever non-CS jobs demand for the resources without causing drastic delays.
- **Share it fairly**  
Also **be polite** and partition the idle resources among CS users evenly.
- **Customizable application level scheduling**  
Not all applications are the same. We optimize the general case. For the others, application-level scheduling can increase productivity.

## The Basis: Koala

- **Load-balancing and Co-allocating grid scheduler**  
Automatic selection of free resources simplifies job submission task. No more the tedious manual selection of clusters. Simultaneous allocation of resources from different clusters for a single application. Deployed on DAS3.
- **Cycle scavenging added to Koala**  
Including the fair-share cycle scavenging. Already implemented fair-sharing policies: Equipartition-All, Equipartition-PerSite.

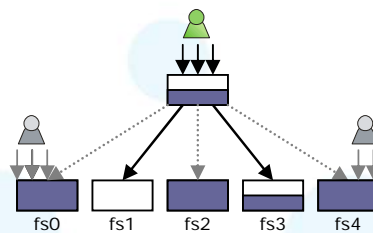


Figure 3: The Koala co-allocating grid scheduler.

## The Submission: GrenchMark

- **GrenchMark is an easy-to-use job submission tool**
- **...and it also analyzes your job execution, too.**  
How much time did the jobs take to finish? What is the resource consumption? Customize: When will Eternity II be solved?



## How about your work?

- **Koala-CS is an easy-to-use, reliable, and polite tool for grids**  
Koala was used so far to run over 300,000 jobs in DAS2 and DAS3. Koala-CS was used so far to run over 25,000 jobs in DAS3.
- **Solving Eternity to over 90% pieces placed**  
Koala-CS found over twenty free machines for the past two months to be used for our Eternity II solver without disrupting the non-CS users of DAS3.
- **You can use Koala-CS for your application**  
Use it out-of-the-box. Let us help you adapt your application to Koala-CS.

## References and Links

- H.H. Mohamed and D.H.J. Epema (2008). KOALA: A Co-Allocating Grid Scheduler. C&C:P&E, 20: 1851-1876.
- A. Iosup and O.O. Sonmez and S. Anep and D.H.J. Epema (2008). The Performance of Bags-of-Tasks in Large-Scale Distributed Systems. ACM/IEEE HPDC-17, pp. 97-108.
- H.H. Mohamed (2007). The Design and Implementation of the KOALA Co-Allocating Grid Scheduler. PhD Thesis.
- J. Buisson and O.O. Sonmez and H.H. Mohamed and D.H.J. Epema (2007). Scheduling Malleable Applications in Multicluster Systems. IEEE Cluster, pp. 372-381.
- Koala web site  
<http://www.st.ewi.tudelft.nl/koala/>
- GrenchMark web site  
<http://grenchmark.st.ewi.tudelft.nl/>
- PDS web site  
<http://www.pds.st.ewi.tudelft.nl/>