

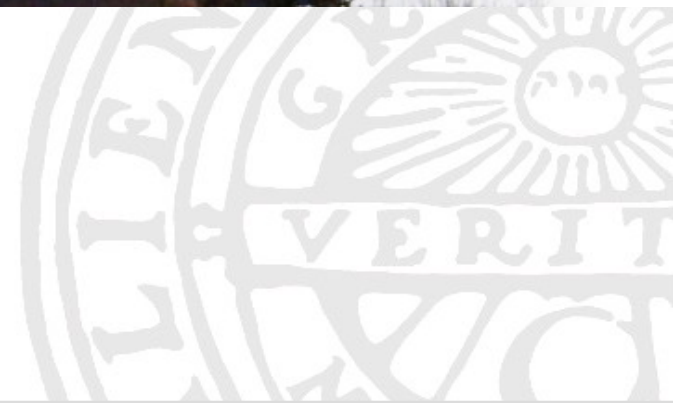


UPPSALA
UNIVERSITET

UPPMAX and UPPMAX systems



Jukka Komminaho
UPPMAX

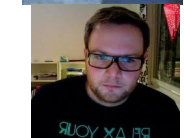
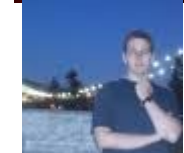




UPPSALA
UNIVERSITET

6 system experts

- Jukka Komminaho
- Tore Sundqvist
- Lennart Karlsson
- Martin Agback
- Peter Ankerstål
- Pontus Freyhult



Uppsala Multidisciplinary Center
Advanced **UPPMAX** Computational Science

jukka.komminaho@it.uu.se



UPPSALA
UNIVERSITET

UPPMAX computer systems

- Kalkyl, Intel E5520, 2572 cores
Delivered in October 2009
- Halvan, Intel X6550, 64 cores
Delivered in April 2011, 2 TB memory
- Tintin, AMD 6220, 2560 cores
Delivered in Nov 2011

Uppsala Multidisciplinary Center
Advanced Computational Science
UPPMAX

jukka.komminaho@it.uu.se



Storage

- What we have
 - Available storage today
 - PanFS, 1 PB
 - Home
 - Global scratch
 - Project storage
 - Lustre 1 PB
 - Global scratch (soon)
 - Node local scratch



Environment

- What we have, (cont)
 - Software environment
 - OS: SL6.3
 - Module environment system
 - Compilers: Intel, PGI, GCC
 - MPI: OpenMPI
 - Queuing system: Slurm
 - A lot of other SW via 'module'



Environment

- Things to think about
 - Try to use the installed software (via module)
 - (Use the cluster most suitable to your problem/application)
 - Try to estimate your applications memory and time requirement



Environment

- Don't do
 - Data base searches in files on remote file system
 - Compiling own version of MPI (usually leads to suboptimal performance)
 - Filling up disk quota
 - Running I/O-intensive jobs from your \$HOME directory



Environment

- Better to...
 - Write large files to node local disk (if possible). At end of job move to network disk (panFS)
 - Use installed MPI, we have tried to install it optimized for our environment
 - Keep an eye on your disk space usage
 - Run I/O-intensive jobs from your /glob, saving temporary large files to your \$SNIC_TMP



Queuing system

- Queuing system
 - We use Slurm (Simple Utility for Linux Resource Management)
 - Use sbatch to submit jobs
 - Important options
 - mandatory: -A <projectid> to run the job in project <projectid>
 - mandatory: -t X:0:0 to request a running time of X hours for the job
 - Mandatory: -p <partition>, to request which partition to use, devel, core or node



Queuing system

- Queuing system, (cont)
 - Parallel jobs:
 - -p node <N>, used for distributed memory programming (shared up to 8 core)
 - Interactive jobs
 - interactive



Queuing system

- Queuing system, (cont)
 - Useful commands:
 - See your jobs
 - 'jobinfo'
 - Summary over a project
 - 'projsummary <projid>'
 - Monitor time used in your project
 - 'projinfo <projid>'
 - Who is a member in your project
 - 'projmembers <projid>'



Storage

- Storage
 - Parallel file system from Panasas
 - Home, default 32 GB
 - Global scratch, default 256 GB
 - Project storage, need to apply for
 - Node local scratch
 - On Kalkyl, 220 GB (a few with >1TB)
 - On Tintin, 1.7 TB



Storage

- Storage
 - Parallel file system from Panasas
 - Fast, >3GB/s aggregated
 - Available on all UPPMAX resources
 - Not good on metadata-intensive operations, avoid making many thousand temporary files on Panasas, opening and closing files or writing to the same file from many nodes unnecessarily, use node local scratch



UPPMAX

- Overall goal:
To serve contented users
- Ways to reach the goal:
 - **support@uppmax.uu.se**
 - Reliable and efficient systems
 - Interesting and relevant workshops, seminars, *et cetera*
 - Strong team
 - Collaborations with the other SNIC centra