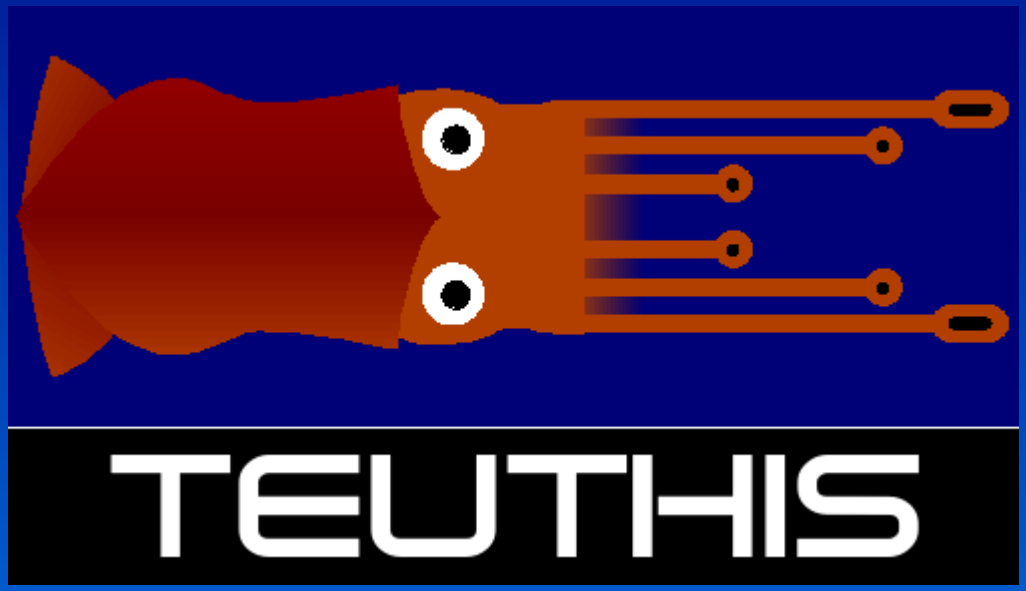
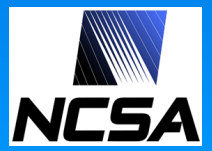
Managing Astrophysical Simulations with



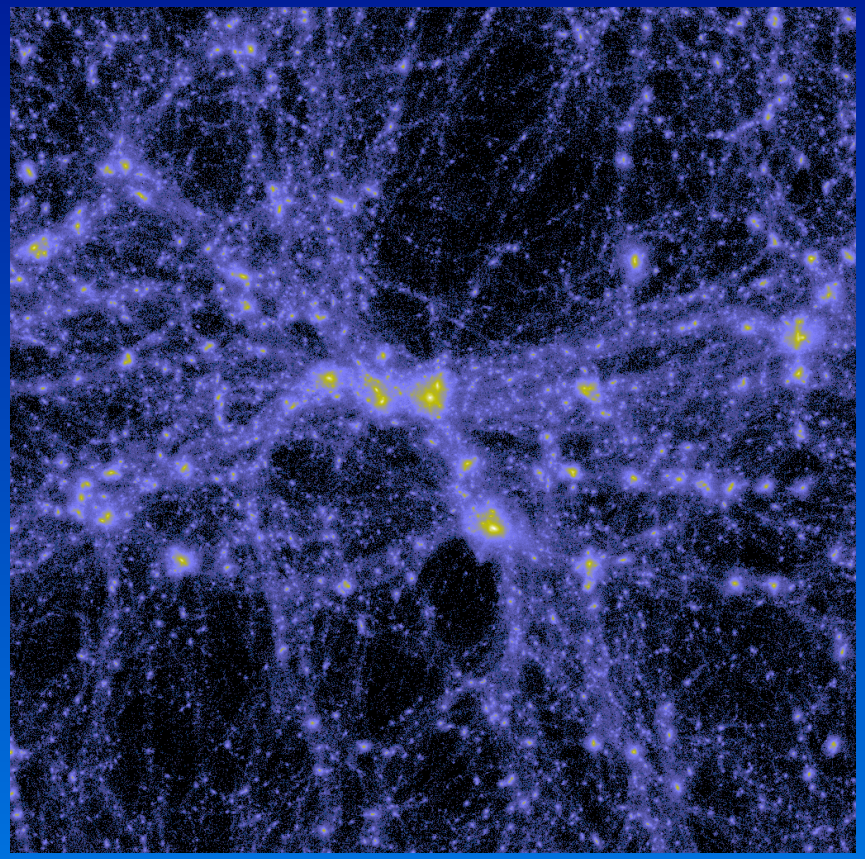
Paul Ricker
National Center for Supercomputing Applications
University of Illinois at Urbana-Champaign

SC|05, Seattle, WA
November 2005



What do astrophysical simulations require?

- Multiphysics
  - Gravitational *N*-body problem
  - Gasdynamics
  - Reactive source terms
  - Radiation transport
- Huge range of scales
  - Length: typically > 10^5
  - Mass: typically > 10^9
- HPC resources a must
  - 100s - 1000s of CPUs
  - Datasets several TB per run
  - 10 – 1000 runs/project to realize scientific goals



What is Teuthis?

- A control panel
  - Remotely configure and build applications
  - Submit and track remote jobs
  - Painlessly create parameter studies and restart jobs
- A data manager
  - Stage and archive data
  - Keep track of where datasets are stored
- A notebook
  - Organize job metadata by purpose and disposition
- An aid to collaboration
  - Share notebook files with collaborators

Running simulations: Teuthis approach

Laptop running Teuthis

Upload source and build executable

Upload parameters and submit job

Retrieve log and stdout

Production supercomputer

Synch source code with repository

Stage job data

Archive job data

Group server

Retrieve data for analysis

Mass storage

Resources manipulated by Teuthis

- Projects
  - Collections of related scientific questions
  - e.g. “Galaxy cluster scaling relations”
- Experiments
  - Vary one or more parameters with a specific code to answer a particular question
  - e.g. “How does the level of galaxy feedback affect the mass-temperature relation?”
- Runs
  - Realizations of a single set of parameters, carried to completion
  - e.g. “Run with 10x fiducial energy input”
- Jobs
  - Individual attempts to complete a run
  - e.g. “Job 123456 on 128 processors for 18 hours”

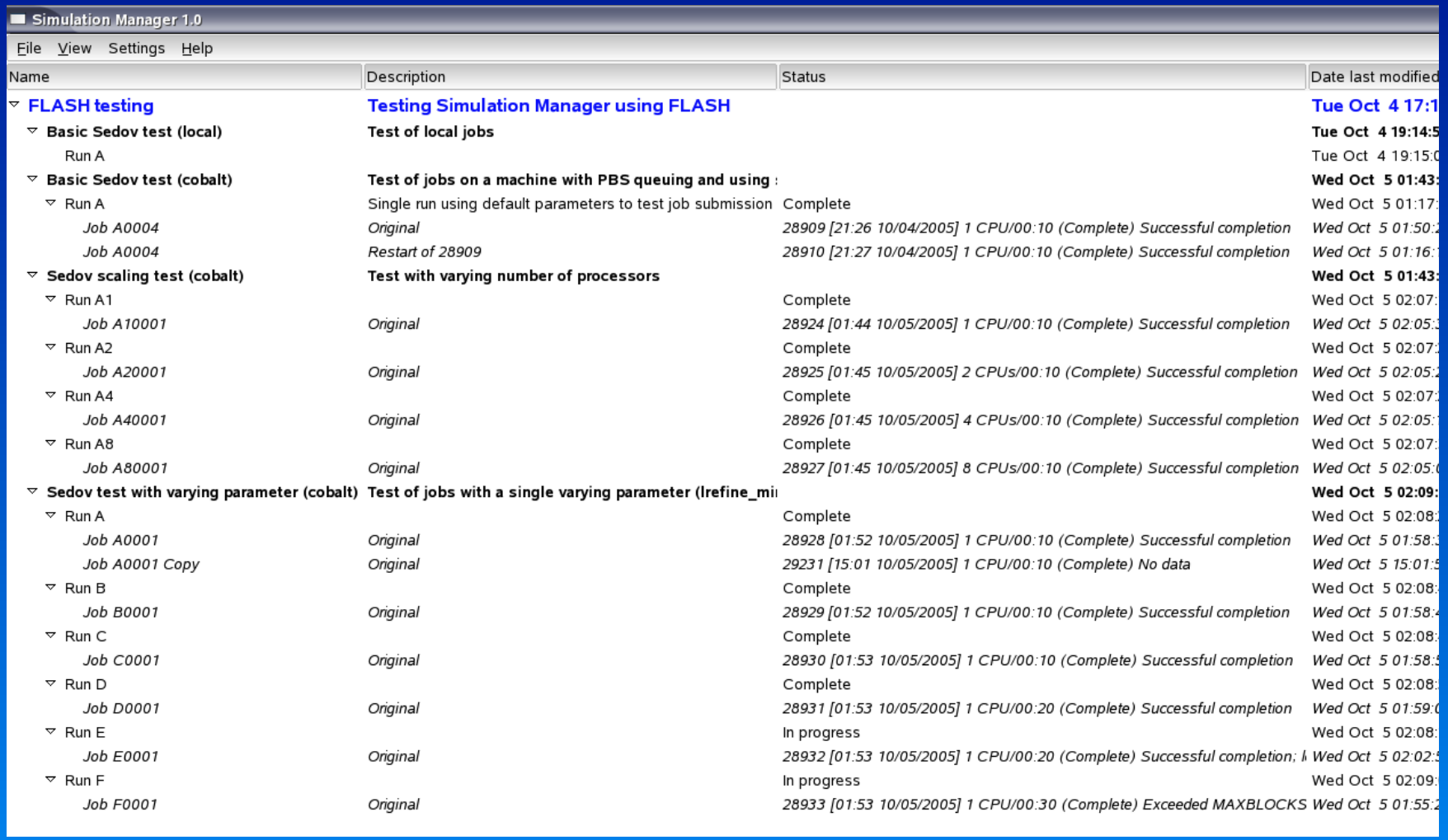
Resources manipulated by Teuthis

- Applications
  - Scientific codes that accept text-based parameter files on input
  - Execute noninteractively
  - Create a log file, messages to stdout, other data files
  - May or may not need to be recompiled for each experiment
  - e.g., simulation code, visualization frame generator, database processing
- Machines
  - Login host name
  - Access method
  - Queuing system
  - Paths (where to build, run, etc.)

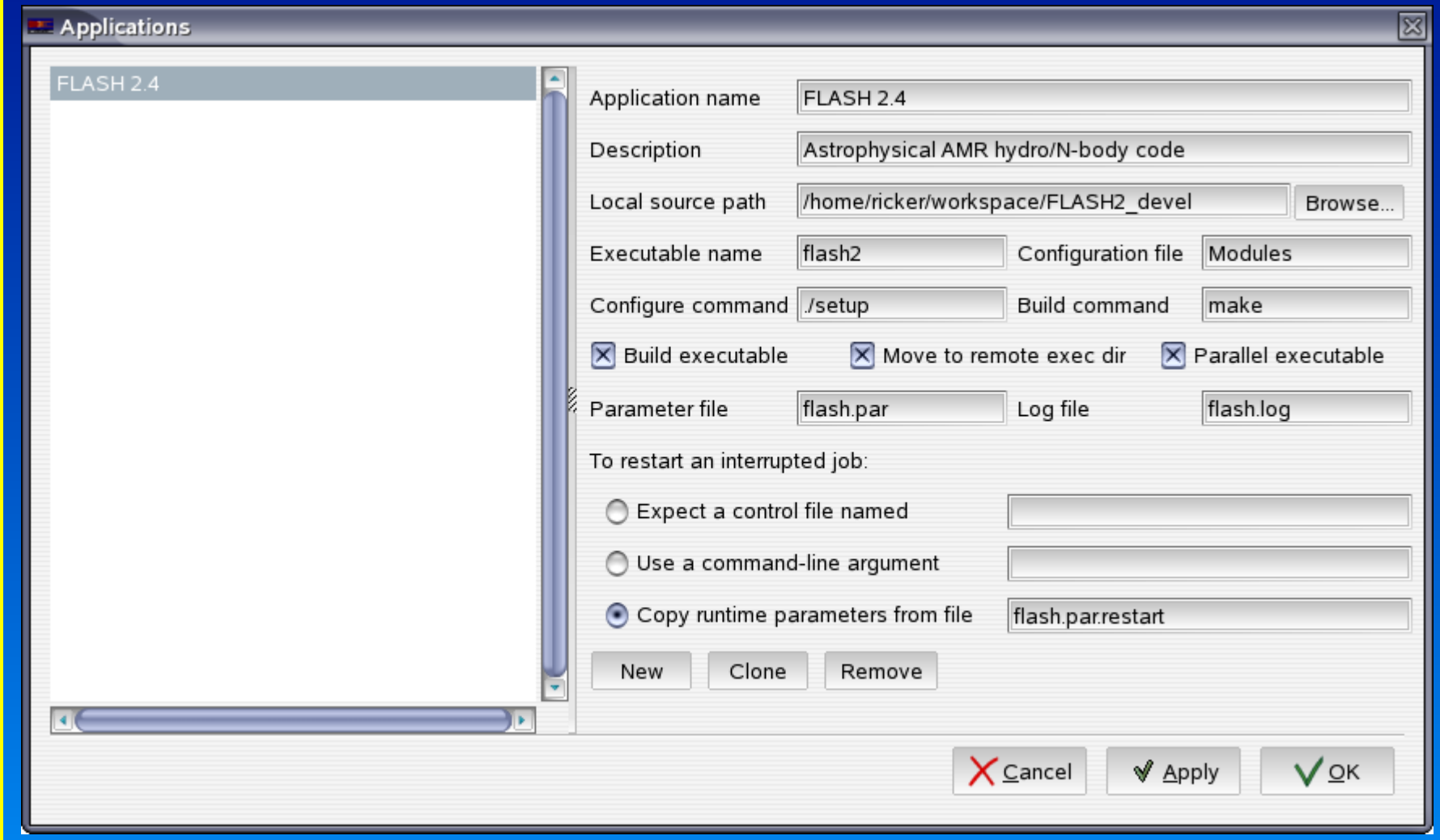
Under the hood

- Python/PyGTK
  - Cross-platform (including GUI)
  - Facilitates rapid prototyping
  - Modules for OS access, process control, Grid services
  - Widely used in physics/astrophysics community
- Access methods
  - Plain ssh – option to store password
  - ssh-agent
  - Kerberos
  - GSI authentication – local certificates or MyProxy
- File transfer methods
  - scp/gsiscp
  - uberFTP
  - globus-url-copy
- Job submission methods
  - PBS, LSF, LoadLeveler
  - Unix process control (no queue)
  - User-specified job template

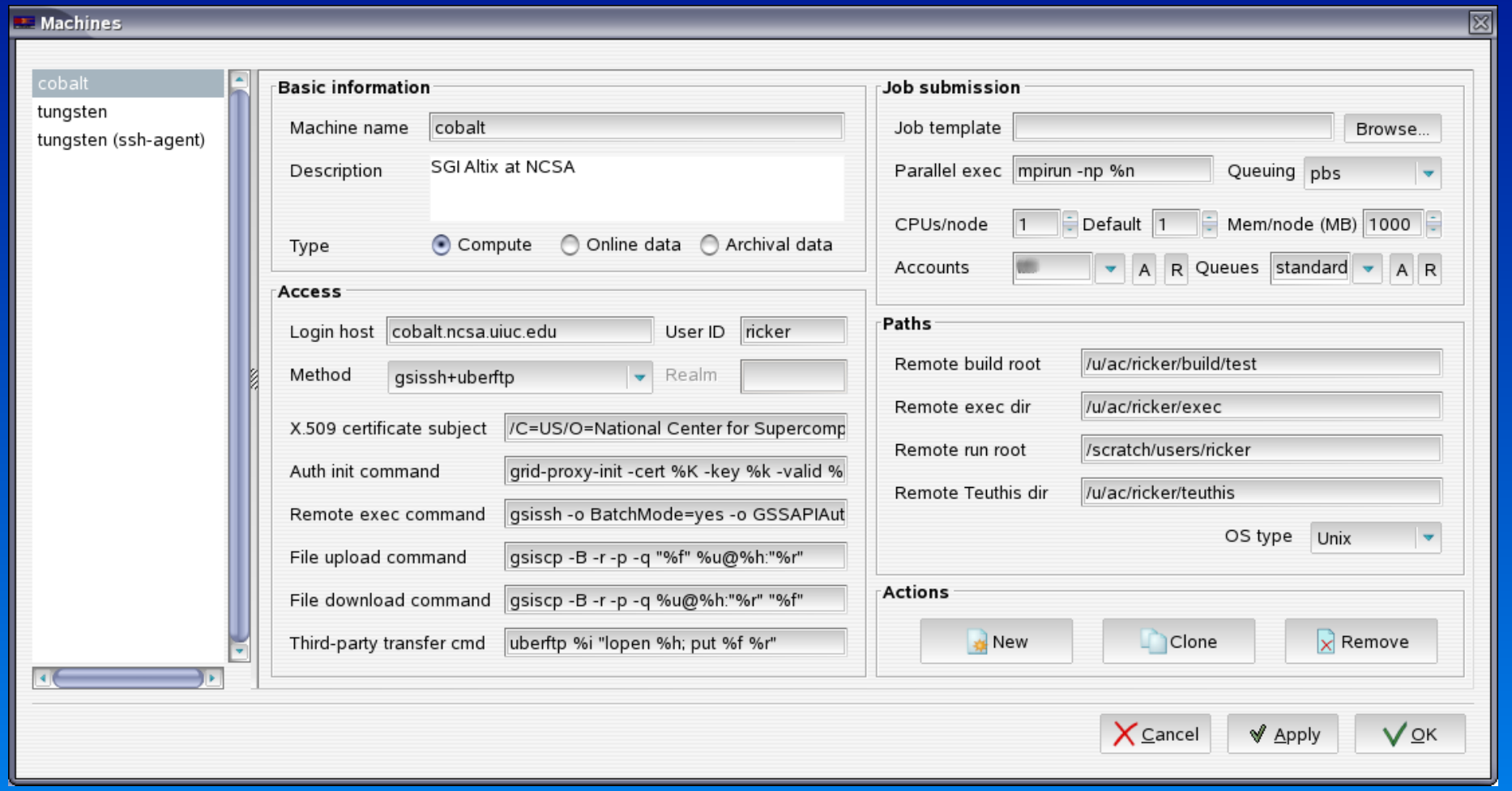
Project view



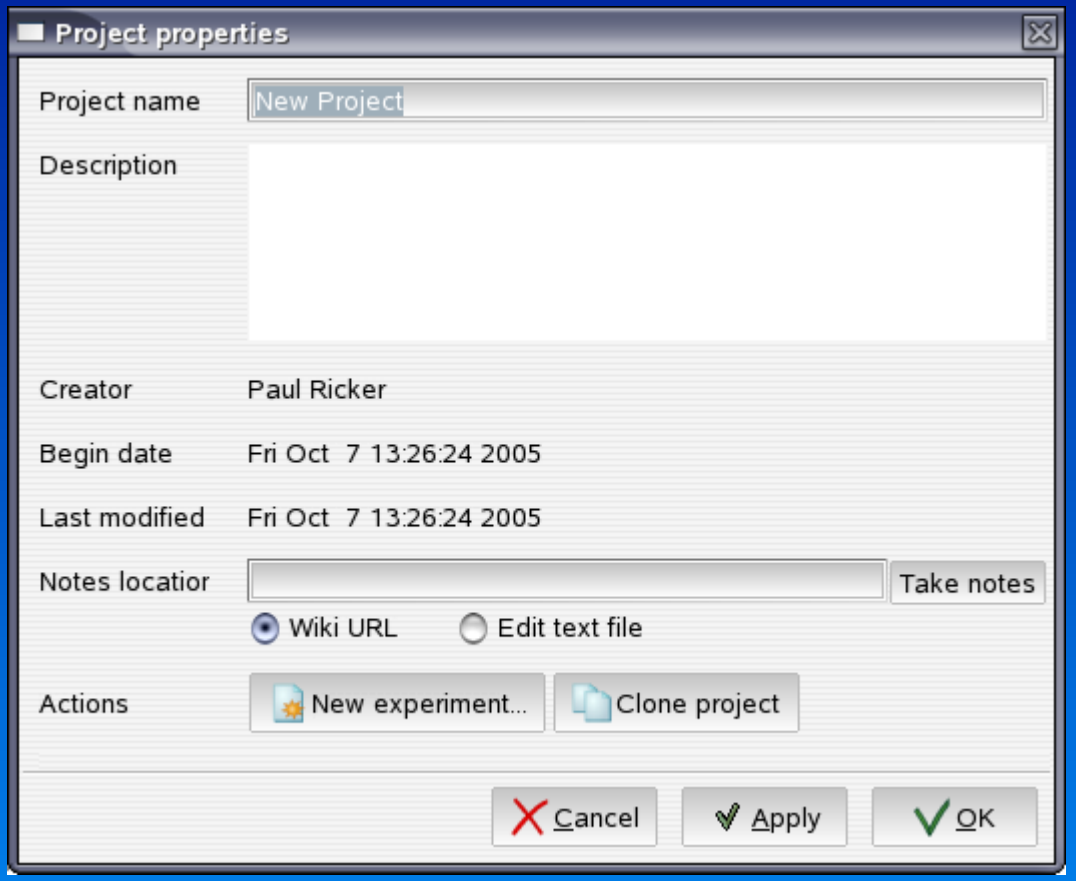
Configuring applications



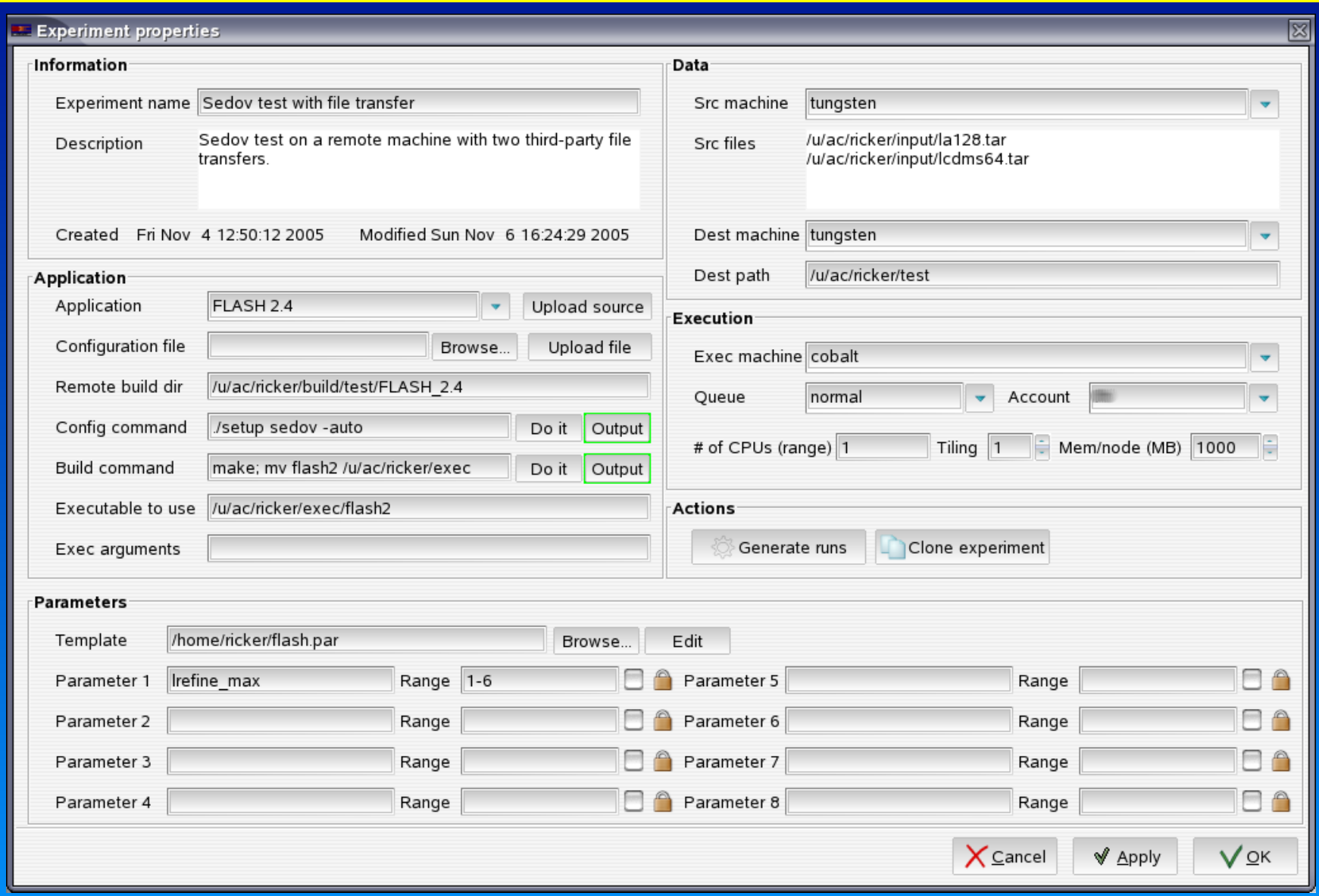
Configuring machines



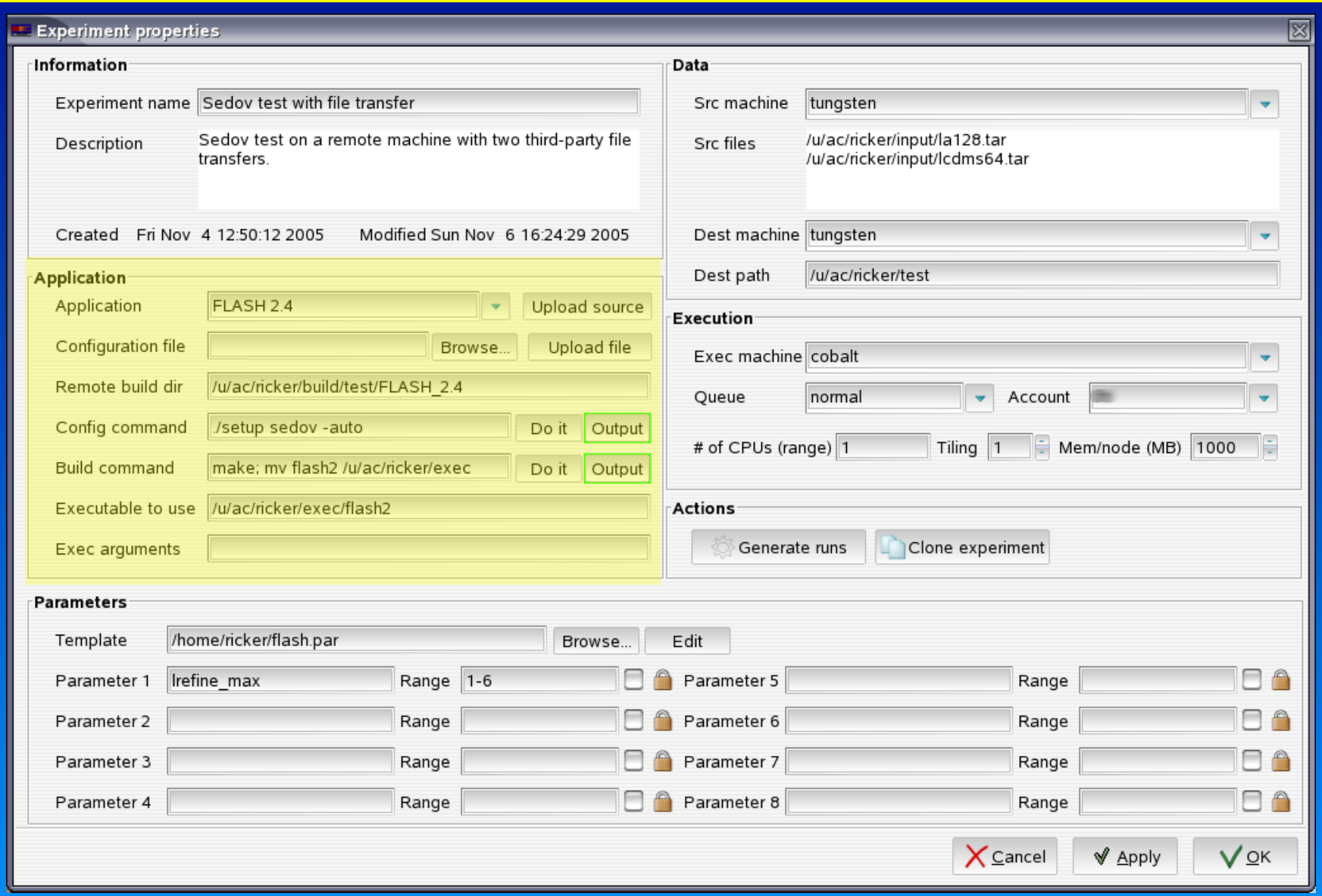
Project dialog



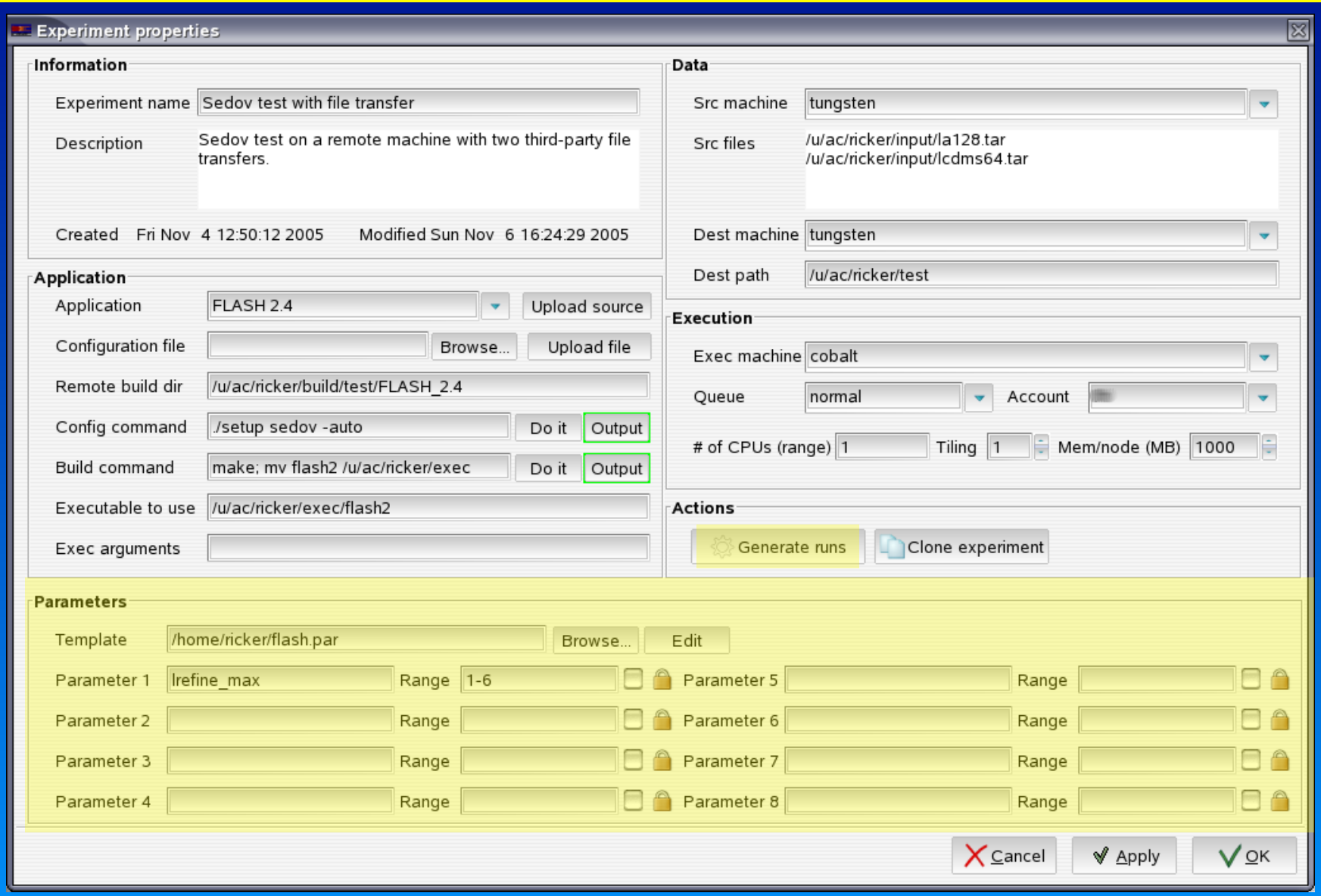
Experiment dialog



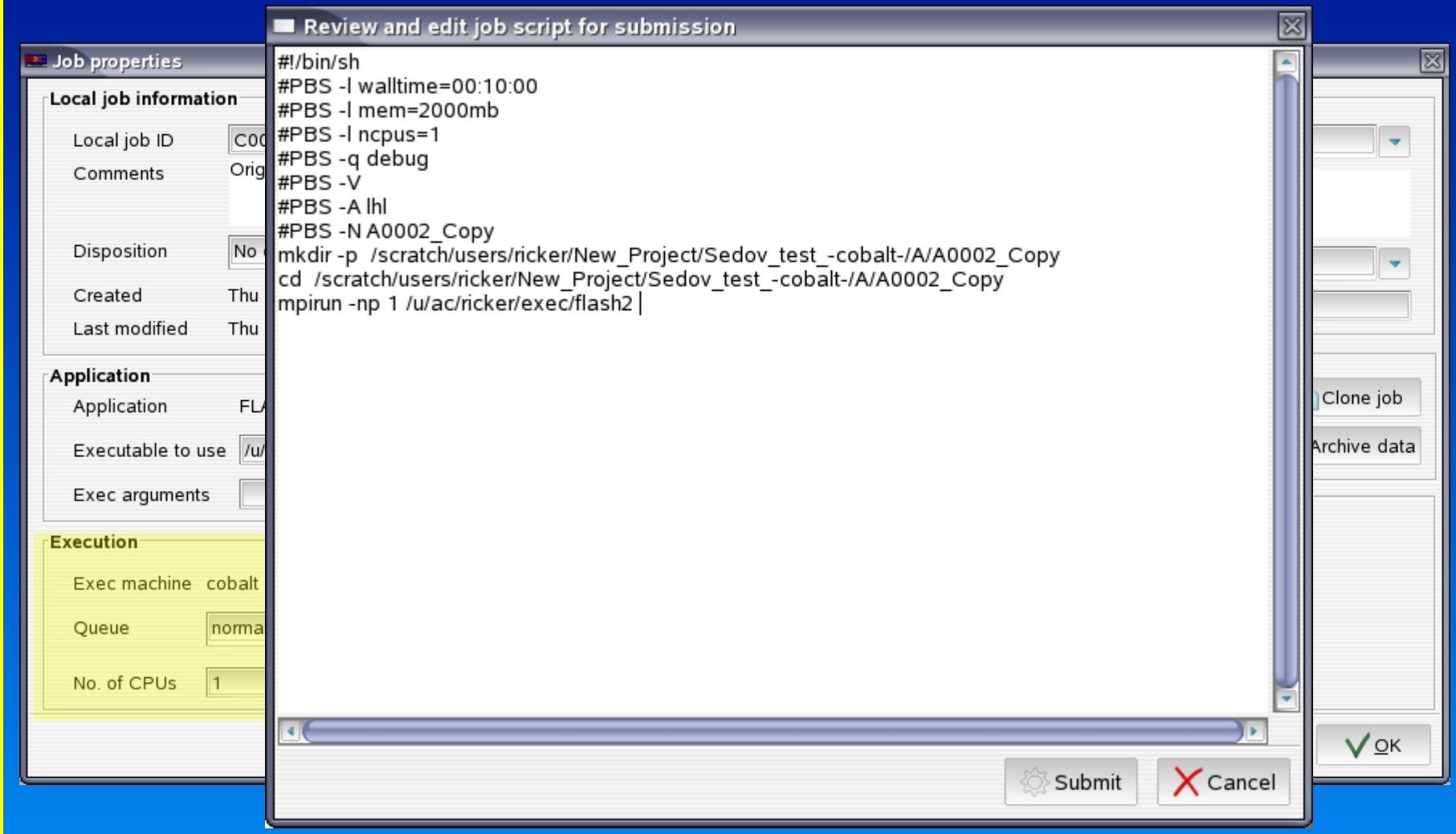
Configuring and building applications



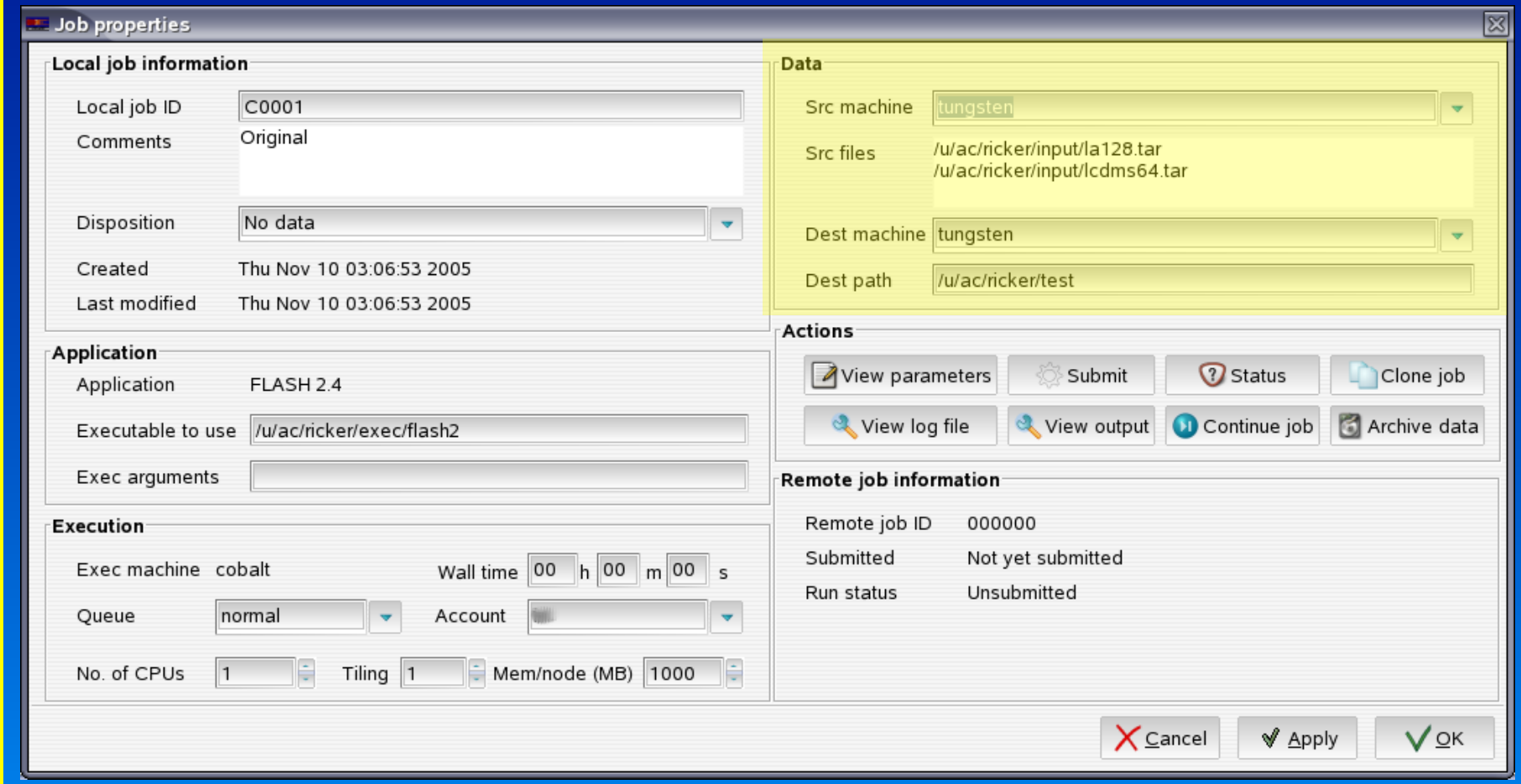
Generating runs



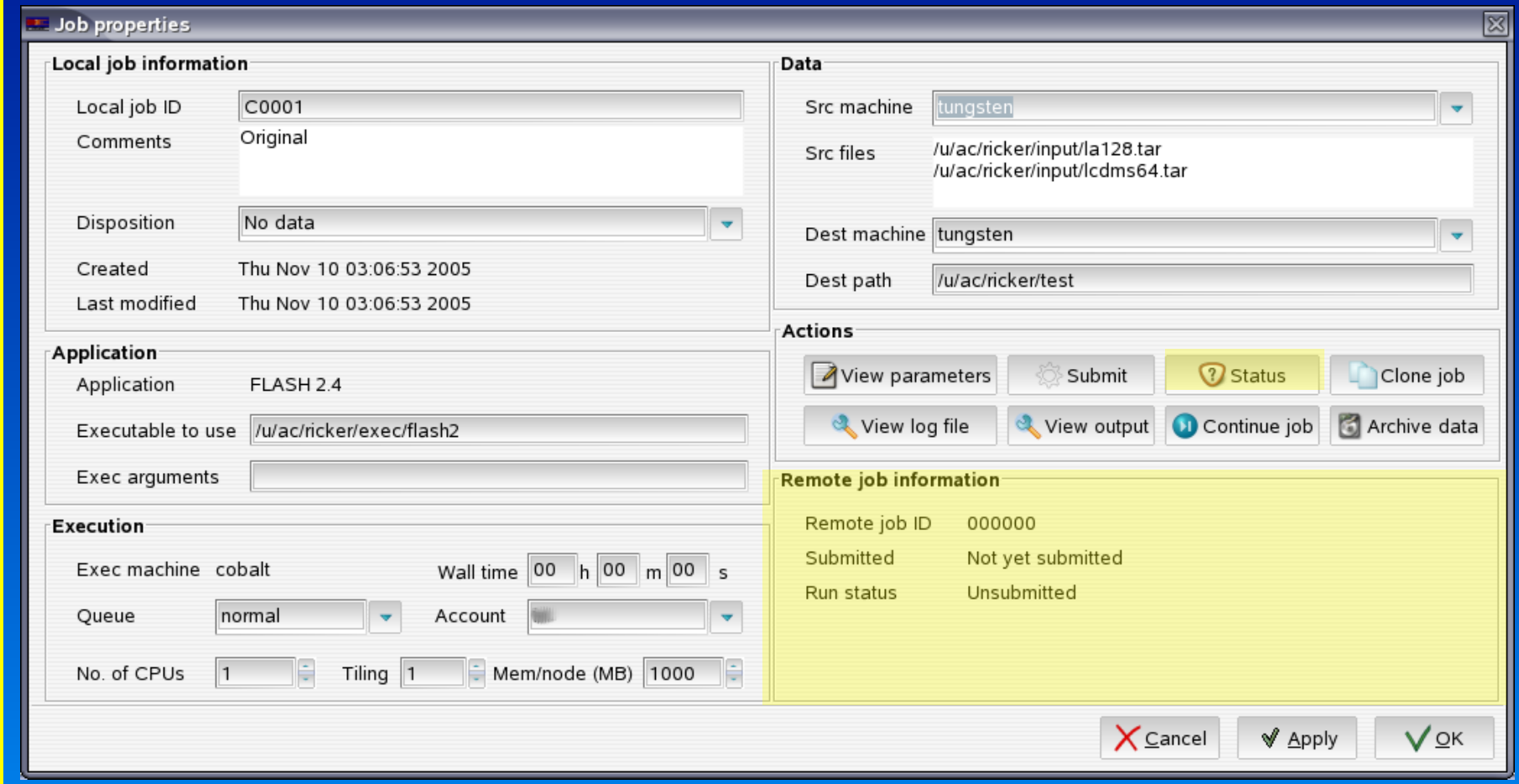
Submitting jobs



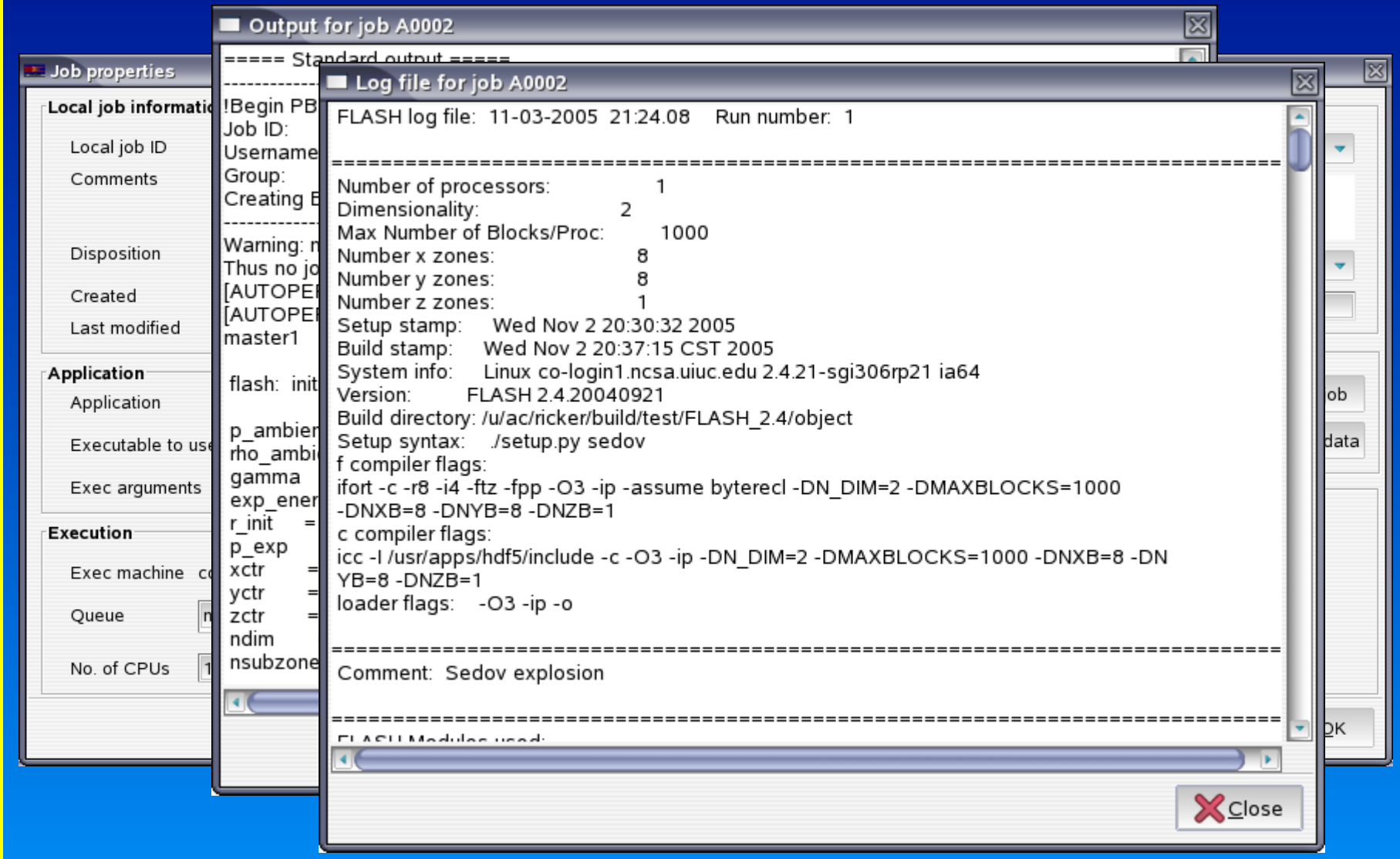
Transferring data



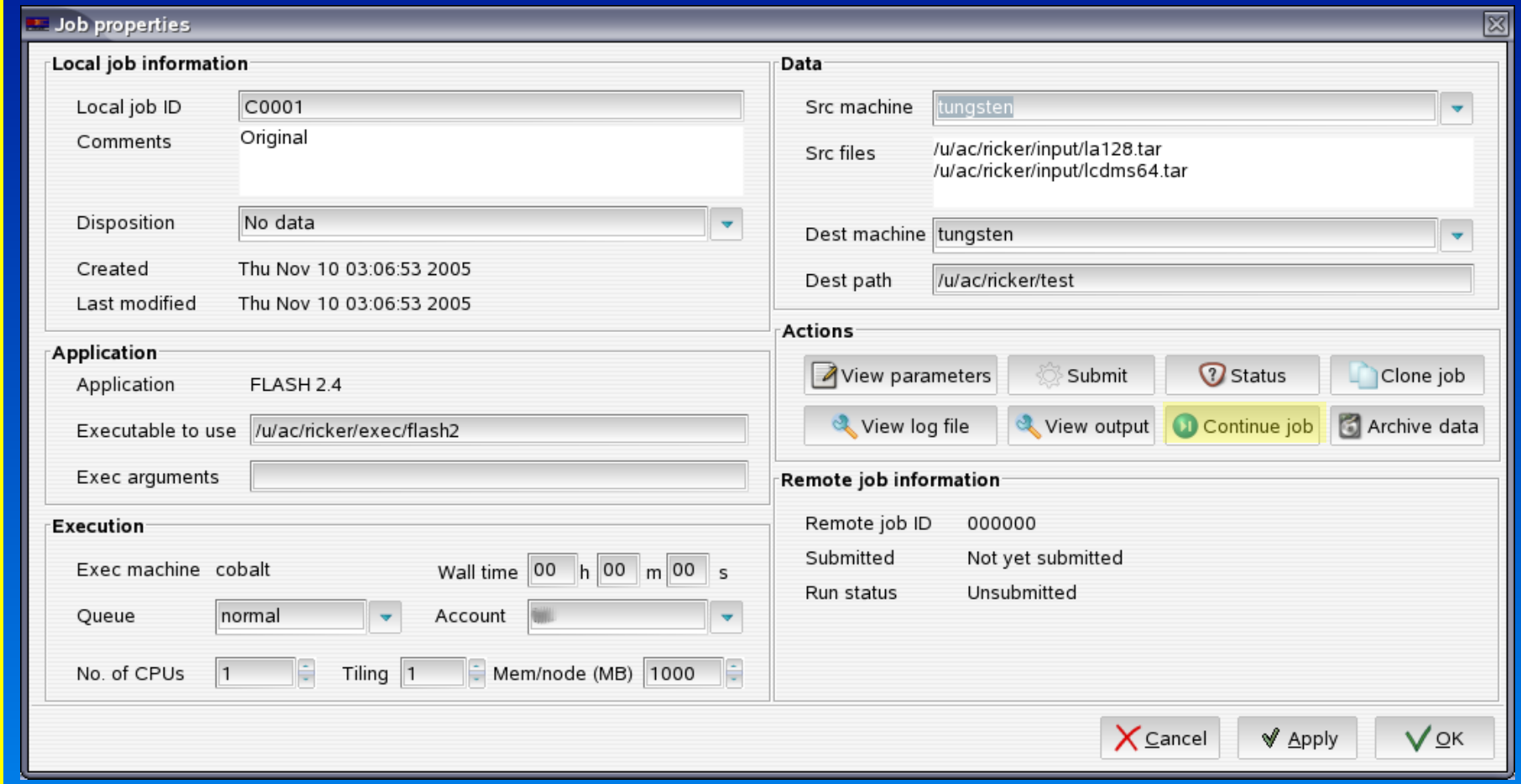
Checking status



Viewing log files and output files



Continuing a job



Future plans

- Data
  - Background file transfers/reliable file transfer (RFT)
  - Data replica management
- Job submission and monitoring
  - GRAM job submission
  - Use of OGRE to handle staging/run/archiving/analysis
- Metadata and analysis
  - Link initial conditions – simulation – analysis jobs
  - Link runs to multiple experiments
  - Plugins for external visualization/analysis tools
- Collaborative aspects
  - Grid portal version

Getting Teuthis

1.0 beta release now available at

http://mazama.ncsa.uiuc.edu/projects/teuthis