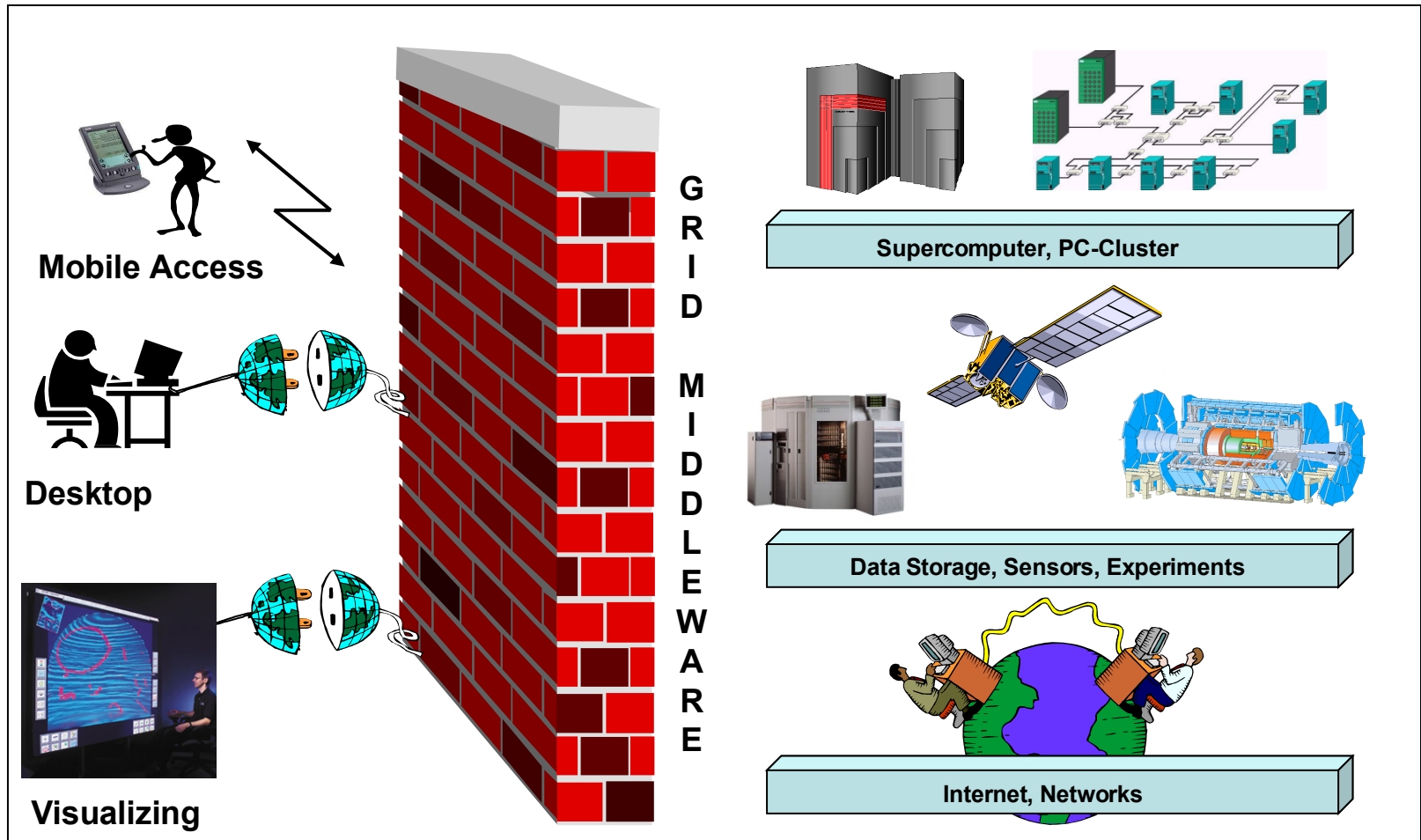


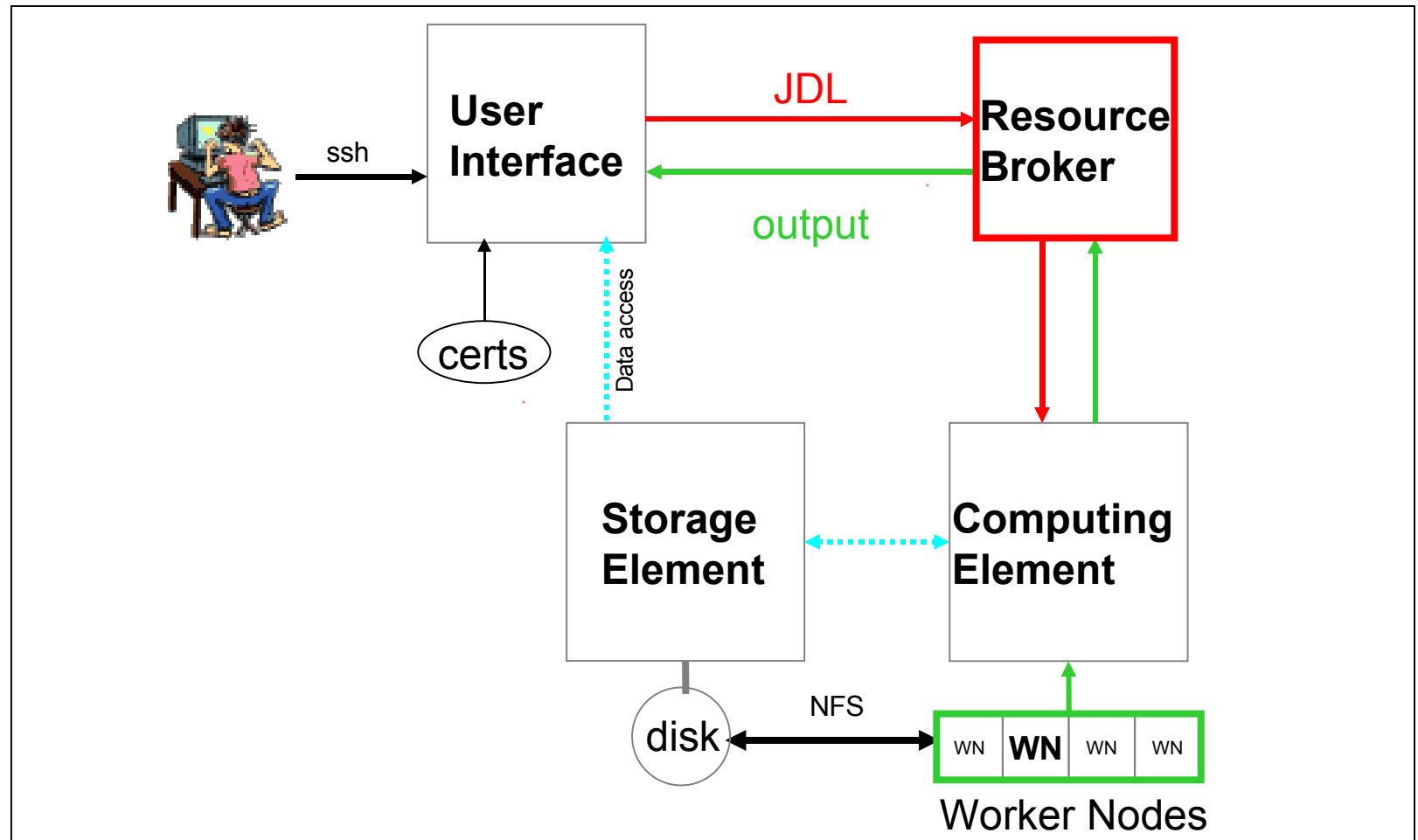


Grid Job submission

Schematics



Schematics (cont.)





Job-submission

- The user specifies the job (i.e. inputfile, #evts, filenames, steering, runscript,...)
 - The job itself knows nothing about Mokka or how to run it
 - It does not know what files are needed and what to do with the output
 - The user has to specify all this as a shellsript.
- Jobsubmission to CE
 - The GRID needs a Job-Definition-File (also written by the user)
 - Actually one commandline
- CE chooses an WN and starts the runscript on it
 - The WN executes the runscript, which was written by the user
- After the job ends the user can download his files from SE



User work

- Mokka needs a steering-file and a GEANT4-macro-file
 - These can be submitted along with the runscript
- All necessary files have to be somewhere on a SE
- The Job-Definition holds:
 - Files to be shipped with submission
 - Information about the VO
 - File to execute on the WN
- The runscript should:
 - Download all necessary files (input, libraries, GEANT4 datafiles...)
 - Check if they are ok
 - Set the environment
 - Execute Mokka (with steering & macro)
 - Check the output
 - Upload the output
 - Terminate



We have

- A “config script” in which the job is being specified
 - Inputfile
 - Number of events per job
 - Number of jobs
 - Simulation Parameters (Model, physics-list, DB to use, ...)
 - Generic steering- and G4macro-file
 - Outputfilename and location where to put it
 - Necessary files
 - Info to put in our MC-DataBase
- A “submission script” which submits jobs
 - Checks if an outputfile (same name) already exists
 - Generates steeringfiles and Job-Discription based on setting in config script
 - Splits the Inputfile in multiple jobs according to settings
 - Writes info in our MC-DB
 - Submits the job
- A “run script”
 - Full working script to execute Mokka on a WN
 - Does all needed tasks (download, environment, processing and upload)
- A Database which holds locations of all simulated files



Things to improve

- Poor documentation
- Configuration is very uncomfortable
 - Maybe a GUI could be created
- MC-DB-Entries have to be filled by user
 - If the user does not care about this, the DB will be full of thrash
- We depend on have a connection to the french Mokka-DB
 - Only CEs with outbound connection are working
 - Workaround exists but not implemented yet
- No access to the status of the jobs while running
 - If a job hangs we are not able to see this
 - Workaround: Selfchecking inside runscript or outbound connection to a central DB
- Joboutput:
 - Lcio-file and tarball of helperfiles (now: only one big tarball)
- Scripts for Marlin are existing too, but usage is more difficult



Further Information

- Grid Computing:
 - <http://www.globus.org/>
 - <http://www.eu-datagrid.org/>
 - <http://cern.ch/lcg/>
 - <http://www.eu-egee.org/>
 - <http://d-grid.de/>
- DESY Grid Web Site:
 - <http://grid.desy.de/>
 - <http://www.dcache.org/>