

The NERC Cluster Grid

- Main aim: To make it easier for NERC institutions to share HPC cluster resources
- Key features:
 - Load and performance monitoring with Ganglia
 - <http://www.resc.reading.ac.uk/ganglia/>
 - Minimal data footprint on remote clusters
 - Applications deployed as *G-Rex services*

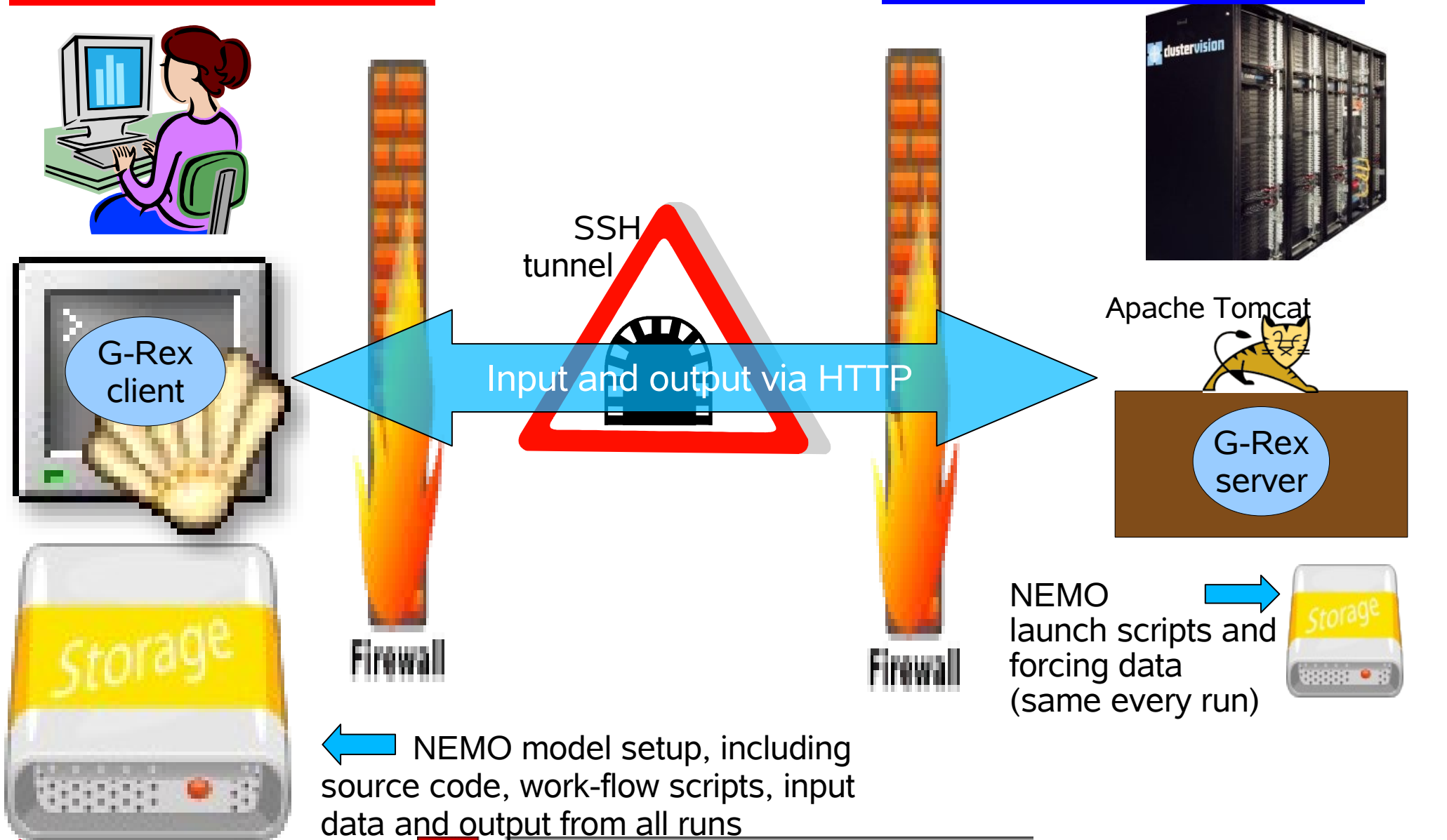
G-Rex (Grid Remote EXecution)

- **Light-weight** grid middleware implemented in Java
 - Built using Spring framework
- Originally developed by Jon Blower at ReSC
- Server component:
 - Web application that sits inside a servlet container
 - Applications deployed as remote **services**
- Client component: grexrun.sh (or grexrun.bat)
 - Behaves as if remote application is running locally
 - Output transferred during run; deleted from server
 - Easily incorporated into existing work-flow scripts

Deployment and use of NEMO G-Rex service at POL

Client: ESSC

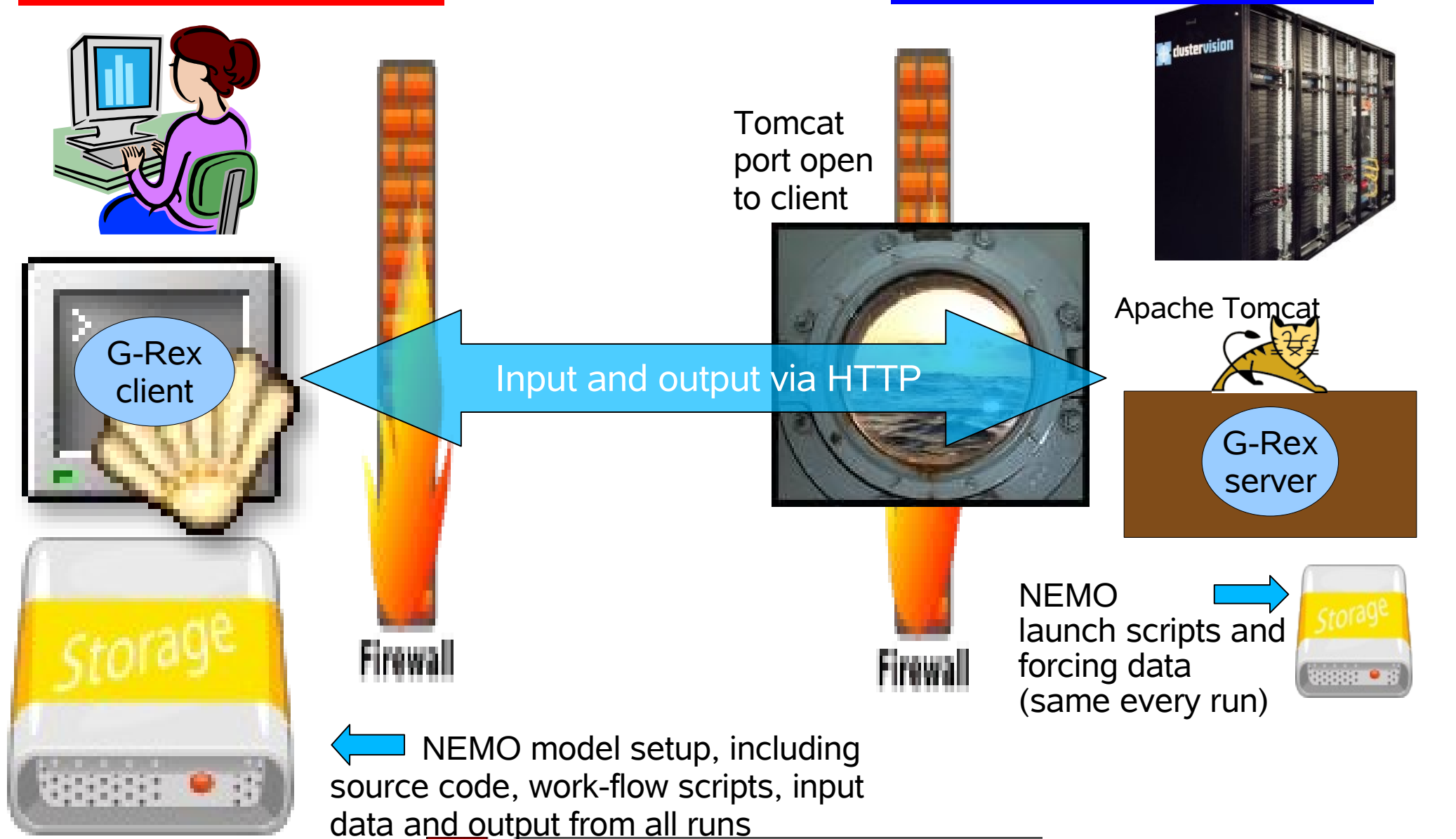
Server: POL



Deployment and use of NEMO G-Rex service at BAS

Client: ESSC

Server: BAS



NERC Cluster Grid services

- nemo
 - NEMO ocean model version 2 (current version)
 - Typical 40-processor run produces 700 500KB files every 2-3 minutes
 - 1-year run produces 25GB spread over 50000 files
- nemo1
 - NEMO ocean model version 1
- make-nemo
 - Compilation service
- qstat
- qdel

Security and accounting

- G-Rex user account details stored in
~dan/.grex/conf/GRexConfig.xml
- G-Rex uses Acegi security framework with HTTP digest authentication
 - Passwords are encrypted
- Grid Engine job IDs start with G-Rex user ID
- Activity logged in ~dan/.grex/log/server.log
 - Start and end of services with G-Rex user ID and command executed