



The LGI Pilot job portal

EGI Technical Forum  
20 September 2011

*Jan Just Keijser*

*Willem van Engen*

*Mark Somers*

**BiG Grid**

the dutch e-science grid



# Outline

- What?
- Why?
- How?
- Pro's and Cons
- What's next?
- Credits





# What is LGI?

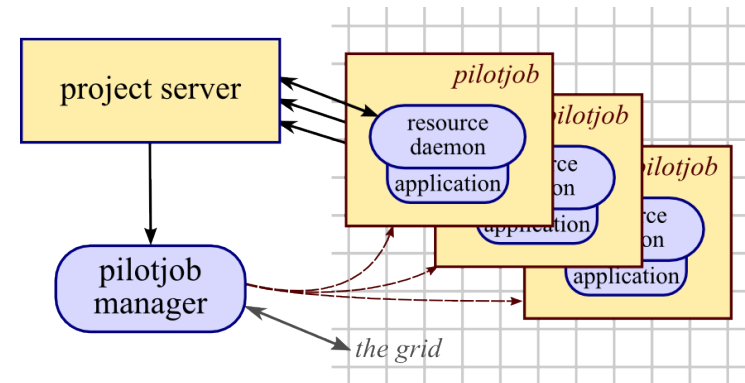
## LGI Project Server

- LGI = “Leiden Grid Infrastructure”
- LGI consists of
  - A *project server* which manages all users, jobs, applications and *resource daemons*
  - An *LGI resource daemon*, which is a process on a cluster or supercomputer which requests work from the project server
- Authentication and authorization is handled by the project server using LGI credentials. These are **not** grid credentials
- Available LGI client interfaces:
  - Web interface (HTTPS, RESTful)
  - Command-line (C++)
  - Python, PHP, Java, R

# What is the LGI Pilotjob framework?

## LGI Pilotjob Framework

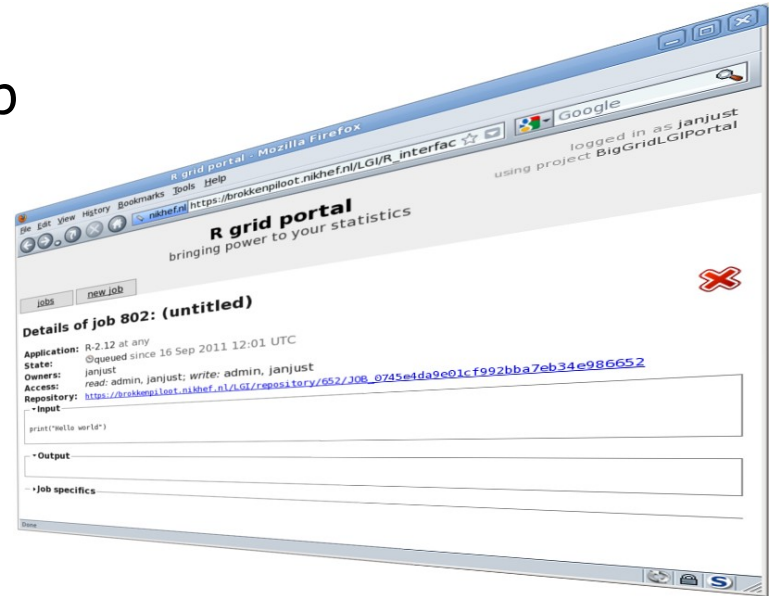
- The LGI Pilotjob framework adds the 'pilotjob manager', which is a new resource daemon
- The 'pilotjob manager' is capable of submitting grid jobs
- These grid jobs are LGI resource daemons themselves, which perform the actual work
- Pilot jobs can be added and stopped on demand
- Pilot jobs are submitted to the grid using "robot" grid credentials stored on a secure hardware device



# What is the LGI Pilotjob portal?

## LGI Pilotjob Portal

- Web portal based on the LGI Pilotjob framework
- Provides username/password authentication
- Provides easy method to download LGI credentials
- Provides a custom interface for an application and/or user community
- Custom R portal now available for bioinformatics users in Leiden





# Why was it built?

## To fulfill the need for a portal that

- Scales well when many jobs are submitted
  - Does not require a new client application
  - Does not require a user to have grid X509 credentials
  - Hides the underlying compute infrastructure (grid, cluster, supercomputing, BOINC, cloud) from the users
  - Can be used from within many applications:
    - Web browser
    - Command-line (C/C++)
    - Perl, Python, PHP
    - Java
    - R, Octave
- and many more ...*



# How does it work?

- Secure apache web server
- “Regular” X.509 certificates, no proxy certificates
- Project database based on MySQL
- Web interface fully PHP based
  
- LGI Pilotjob manager monitors the job queue
- If the number of queued jobs goes up, or stays too high, then extra grid pilot jobs are launched
- If the job queue is empty for some time then the grid pilot jobs are shut down





# Pro's and Cons

## Pro's

- Portal sysadmin is in full control of who runs what where
- Complete separation of grid credentials and portal credentials
- The client interface can be integrated very closely with the end-user application
- Hides the compute infrastructure complexity

## Cons

- All jobs run on the grid using the same grid credentials
- Storing of output files stored on the grid is not addressed
- Instead of competent grid operators you now (also) need a competent portal administrator



# What's next?

## Conclusions

- LGI Pilotjob portal has been deployed for R & Rgpu users
- Users do not need grid credentials, only a username and password on the LGI Pilotjob portal itself
- The system works extremely well as long as the input and output files are small

## Future work

- Interface the LGI portal with SAML-based logins
- Add new applications
- Improve statistics and logging
- Investigate how an LGIpilot job can submit new jobs itself





# Credits

## Developers

- Mark Somers, Theoretical Chemistry, Leiden University
  - Developer of the Leiden Grid Infrastructure (LGI)
- Willem van Engen, Nikhef / BiG Grid
  - Wrote the LGIpilot framework
  - Wrote the new python-lgi interface
  - Wrote the Rlgi interface

<http://wiki.biggrid.nl/wiki/index.php/LGI>



**BiG Grid**  
the dutch e-science grid

