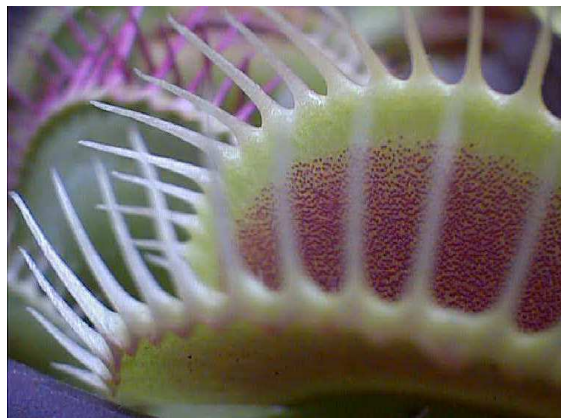


# *Greedy Pool @ EPFL*

## *Desktop Grid using Condor*



Dr. Michela Thiémard, EPFL-DIT-EX  
Pascal Jermini, EPFL-DIT-EX  
[grid-admins@groupe.epfl.ch](mailto:grid-admins@groupe.epfl.ch)

# Talk overview

- ▷ Project aim
- ▷ Characteristics needed for the grid software
- ▷ Grid software that has been considered
- ▷ Why Condor?
- ▷ Our Condor pool: Greedy

# EPFL computing equipment

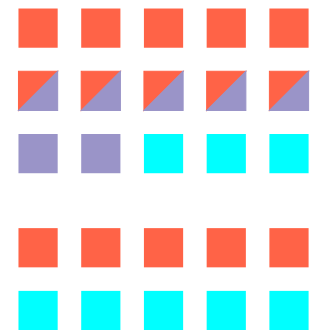
## Dedicated HPC machines

- ▷ about fifteen clusters of different types

## “Simple” machines

- ▷ classrooms

	Win	Win/Linux	Linux	Mac
CPU qty	626	317	30	32



- ▷ more than 3000 employee computers

# EPFL computing equipment

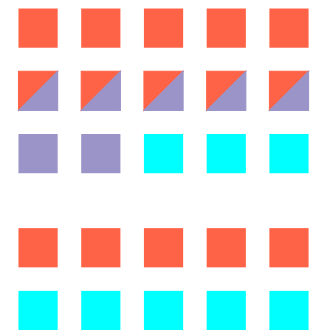
## Dedicated HPC machines

- ▷ about fifteen clusters of different types

## “Simple” machines

- ▷ classrooms

	Win	Win/Linux	Linux	Mac
CPU qty	626	317	30	32



- ▷ more than 3000 employee computers

⇒ a lot of CPU power to harvest

# EPFL computing equipment

## Dedicated HPC machines

- ▷ about fifteen clusters of different types

## “Simple” machines

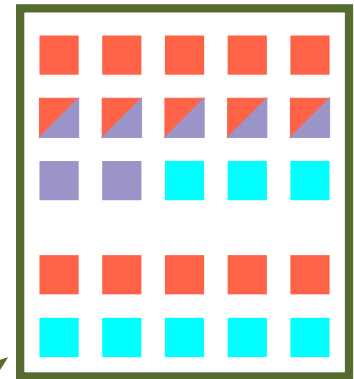
- ▷ classrooms

	Win	Win/Linux	Linux	Mac
CPU qty	626	317	30	32

- ▷ more than 3000 employee computers

⇒ a lot of CPU power to harvest

Desktop Grid



# *Desktop Grid: software characteristics*

## Various needed characteristics

- ▷ Light and simple to install on grid nodes

# Desktop Grid: software characteristics

## Various needed characteristics

- ▷ Light and simple to install on grid nodes
- ▷ Easy entry and exit from the desktop grid pool, such as:
  - ▷▷ time slots definition
  - ▷▷ activity detection on a grid node to avoid bothering its owner

# Desktop Grid: software characteristics

## Various needed characteristics

- ▷ Light and simple to install on grid nodes
- ▷ Easy entry and exit from the desktop grid pool, such as:
  - ▷▷ time slots definition
  - ▷▷ activity detection on a grid node to avoid bothering its owner
- ▷ **Security** of data on the grid nodes



# Desktop Grid: software characteristics

## Various needed characteristics

- ▷ Light and simple to install on grid nodes
- ▷ Easy entry and exit from the desktop grid pool, such as:
  - ▷▷ time slots definition
  - ▷▷ activity detection on a grid node to avoid bothering its owner
- ▷ **Security** of data on the grid nodes
- ▷ Multi-platform





# Desktop Grid: software characteristics

## Various needed characteristics

- ▷ Light and simple to install on grid nodes
- ▷ Easy entry and exit from the desktop grid pool, such as:
  - ▷▷ time slots definition
  - ▷▷ activity detection on a grid node to avoid bothering its owner
- ▷ **Security** of data on the grid nodes
- ▷ Multi-platform
- ▷ Compatible with standards middlewares, such as Globus and Unicore (**Swiss Grid Initiative**)

# Grid software considered

Several software considered and/or tested

- ▶  BOINC needs adaptation of the application
- ▶  Grid MP: commercial solution
- ▶  PBS Professional™ : commercial solution, known at EPFL, used for clusters management
- ▶  **Condor** High Throughput Computing : free solution, already used (in small scale) at Lausanne University and at EPFL

# Why Condor?

- 👍 Minimal intrusion on the grid nodes
- 👍 Time slot definition & activity detection

# Why Condor?

- 👍 Minimal intrusion on the grid nodes
- 👍 Time slot definition & activity detection
- 👍 Only **one submit server** is needed for all platforms

# Why Condor?

- 👍 Minimal intrusion on the grid nodes
- 👍 Time slot definition & activity detection
- 👍 Only **one submit server** is needed for all platforms
- 👍 **No user access** is needed on grid nodes
- 👉 Jobs run under a standard user whose account is **locked**  
grid nodes becomes multi-user machines!

# Why Condor?

- 👍 Minimal intrusion on the grid nodes
- 👍 Time slot definition & activity detection
- 👍 Only **one submit server** is needed for all platforms
- 👍 **No user access** is needed on grid nodes
- 👉 Jobs run under a standard user whose account is **locked**  
grid nodes becomes multi-user machines!
- 👍 Grid nodes are **authenticated** via SSL certificates
- 👍 **Data integrity and encryption** performed via SSL

# Why Condor?

- 👍 Able to run compiled jobs: C, C++, Fortran, . . .
- 👍 Able to run script jobs: Matlab, Maple, Octave, R, Mathematica, . . .



# Why Condor?

- 👍 Able to run compiled jobs: C, C++, Fortran, . . .
- 👍 Able to run script jobs: Matlab, Maple, Octave, R, Mathematica, . . .
- 👍 Promotes a **fair resources sharing**

# Why Condor?

- 👍 Able to run compiled jobs: C, C++, Fortran, . . .
- 👍 Able to run script jobs: Matlab, Maple, Octave, R, Mathematica, . . .
- 👍 Promotes a **fair resources sharing**
- 👍 In case of an unexpected loss of a grid node (node reclaimed by owner, reboot, . . . ), jobs are
  - ▷▷ either **suspended**, until node is released
  - ▷▷ or **interrupted** and rerun elsewhere from scratch

# Why Condor?

- 👍 Able to run compiled jobs: C, C++, Fortran, . . .
- 👍 Able to run script jobs: Matlab, Maple, Octave, R, Mathematica, . . .
- 👍 Promotes a **fair resources sharing**
- 👍 In case of an unexpected loss of a grid node (node reclaimed by owner, reboot, . . . ), jobs are
  - ▷▷ either **suspended**, until node is released
  - ▷▷ or **interrupted** and rerun elsewhere from scratch
- 👍 Quite robust in case of server crash

# Why Condor?

- 👍 Able to run compiled jobs: C, C++, Fortran, . . .
- 👍 Able to run script jobs: Matlab, Maple, Octave, R, Mathematica, . . .
- 👍 Promotes a **fair resources sharing**
- 👍 In case of an unexpected loss of a grid node (node reclaimed by owner, reboot, . . . ), jobs are
  - ▷▷ either **suspended**, until node is released
  - ▷▷ or **interrupted** and rerun elsewhere from scratch
- 👍 Quite robust in case of server crash
- 👍 Compatible with Globus & Unicore

# Condor history

- ▷ Developed at University of Wisconsin-Madison (UW-Madison)
- ▷ First implementation in 1986
- ▷ The «**Condor Project**» started in 1988, directed by Prof. Miron Livny
- ▷ Known pools worldwide: 1559, corresponding to **100418 hosts!**
- ▷ Latest stable version: 6.8.1 (19.09.2006)
- ▷ <http://www.condorproject.org>

# Compatibility tests

- ▷ Various combinations of architectures/Linux distributions/Windows versions are supported.
- ▷ Confirmed on following EPFL platforms:
  - ▷▷ Intel architecture (x86)
    - ✕ Fedora Core 3
    - ✕ Debian Linux 3.0 (Sarge)
    - ✕ Ubuntu 6.06
    - ✕ SuSE Linux 9
    - ✕ Windows XP Pro
  - ▷▷ PowerPC architecture
    - ✕ MacOS X
    - ✕ SuSE Linux 9 (PPC64)

# Our pool: Greedy

## Current pool

Active during **nights** and **week-ends**, formed by:

- ▷ 10 Linux machines,
- ▷ 160 Linux/Windows machines, and
- ▷ 1 MacOSX

## Ideal jobs

- ▷ **Monoprocessor** jobs
- ▷ Duration: **max 3-4 hours**
- ▷ Reserved for **production**, no debugging

# Our goal

## Our goal

Integrate as many machines as possible in the Greedy pool

## How to reach this goal?

Make the access of machines to the pool painless:

- ▷ Custom-made packages and installers (almost no human-intervention needed)
- ▷ Support large variety of platforms
- ▷ Make this grid effort known to the EPFL population



# *More informations*

- ▷ `http://greedy.epfl.ch`
- ▷ `grid-admins@groupe.epfl.ch`