Analisi Tier2 e Tier3 Esperienze ai Tier-2

Giacinto Donvito
INFN-BARI

OUTLOOK

- Alice Examples
- Atlas Examples
- CMS Examples

Alice Examples

- ALICE Tier-2s at the moment do not support interactive analysis
  - not foreseen by ALICE computing model for Tier-2s
  - No purely local (non-grid) access to Tier-2 batch queues (CEs) and SEs
    - The idle-time of the INFN-Bari resources can be used for non-grid jobs submitted by local community for testing and debugging the code developing
- Interactive analysis is done with PROOF at Analysis Facilities (--> D.Berzano)
  - CAF (mainly), SKAF, LAF, GSIAF
  - Very efficient for small datasets (< 1 M events)
- ALICE INFN Tier-2 federation is considering the possibility to provide AF services
  - Prototype of virtual AF at Torino Tier-2

Alice Examples

- ALICE grid/analysis solutions offer good degree of interactive functionality:
- Access to data via xrootd
  - Allows to read data on any SE from ROOT prompt
    - TGrid::Connect("alien://");
    - TFile::Open("alien:///alice/cern.ch/….");
  - Very useful to test new code
- Automatization of grid job submission and output retrieval
  - Alien plugin (--> M.Masera)
  - Do everything from ROOT prompt: data selection, software version, job submission and splitting (data/SE driven), output merging and local copy

Atlas Requirements

- Grid access (Ganga, pathena)
- Interactive facilities (coding and debugging)
- Local (fast) job submission
- Efficient I/O from analysis jobs
  - This often means a better CPU efficiency resulting in a better usage of the available resources
- Fast and reliable SRM interface for WAN transfers
  - DDM Tools (DQ2, efficient FTS channel)

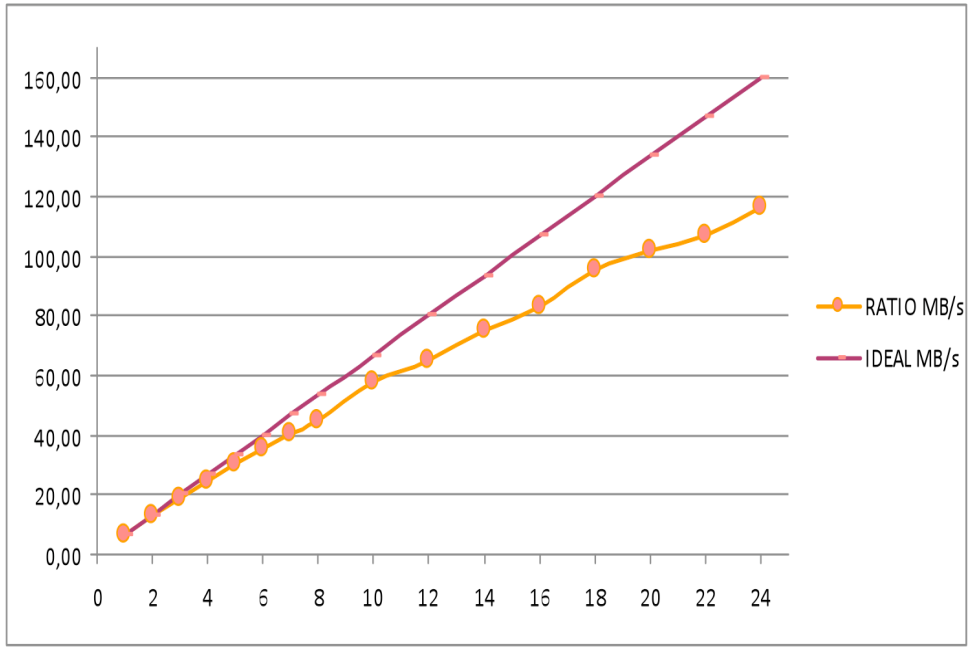
Overview of Milan Farm

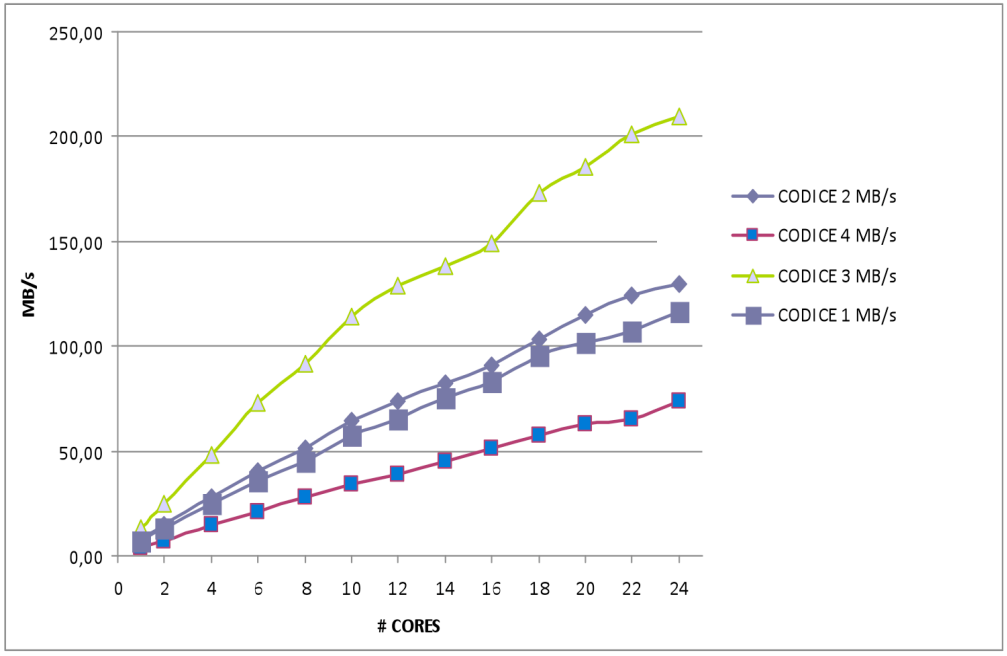
- Local and grid facilities separated with different batch servers
- 2 CEs and ~450 cores accessible via grid
- Balanced pool of 3 UIs for interactive use and grid job submission
- One dedicated UI for submission to a local PBS cluster with three dedicated WN
- A pool of three nodes running “proof” in clustered mode (“xproof”)
- A GPFS common file system serving home, local and grid (with StoRM interface) storage areas
- Software area served by two NFS server to all the local and grid nodes

Atlas Example

Test on PROOF performance

- Test sono stati fatti a Milano con contributi importanti da UD/TS e PI, utilizzando 4 tipi diversi di codice:





StoRM in Milan

- Two frontends balanced via DNS
- Four gridftp interfaces, balanced via DNS
  - Used mainly by FTS channels
- Single backend

GPFS file system

- Two master clusters of NSD servers
  - 20 machines serving all the available storage, connected via fibre channel
  - Different file systems for grid areas, local storage and users' home directories
- Client machines organized in several slave clusters according to their function (UIs, grid worker nodes, local worker nodes, proof nodes...)
  - More stable
  - Easier to configure
    - Common parameters for all the machines in the cluster
    - Not all the clients need to see all the exported filesystems
- The tests on both single node and clustered proof show that, with real analysis code, GPFS is not a bottleneck
  - N.B. s/w area exported via NFS, not GPFS

Motivations

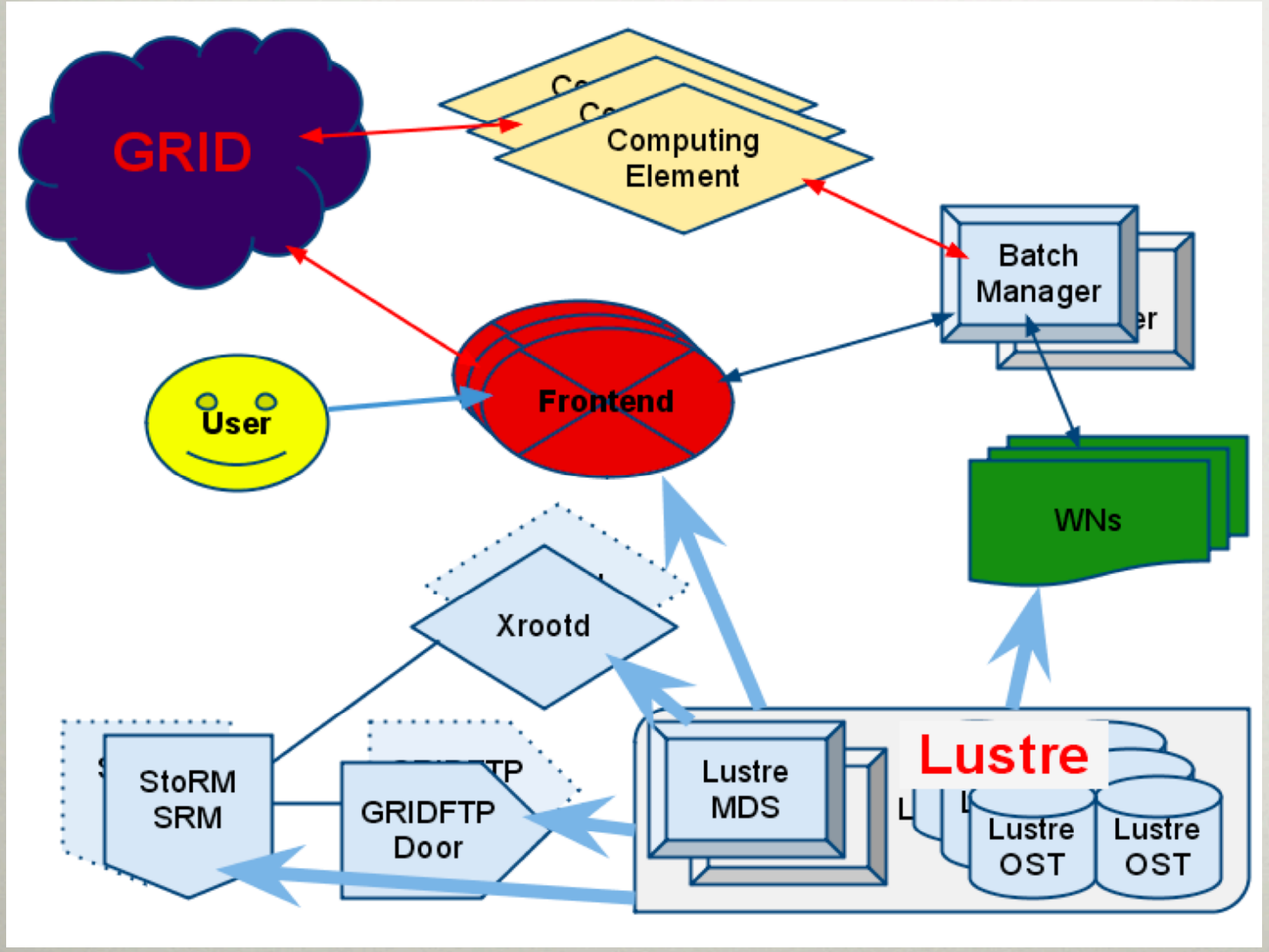
- Provide an environment supporting:
  - Local use of CMSSW (main releases + on demand)
    - local running (small tasks)
    - code development
  - Submission of jobs to the distributed infrastructure
  - Storage of limited amount of data
    - for code validation
    - for specific studies (e.g. detector studies)
  - Storage of personal data
    - needed for interactive analysis
  - Interactive activities
    - analysis with root
    - work environment (mail, browser, document editing, ...)
  - Backup of critical files
    - Hardware failure or accidental deletion

Overview of INFN-BARI farm

- ~10 different VO supported
- >1064 core
- ~140 WN
- ~500TB of Lustre storage
- SRM/gridftp supported
- Xrootd supported
- 2 multi VO frontend machine + 1 CMS dedicated frontend
- phedex node
- 1 LCG-CE + 1 CREAM-CE

General overview of the existing farm configuration at the INFN-BARI

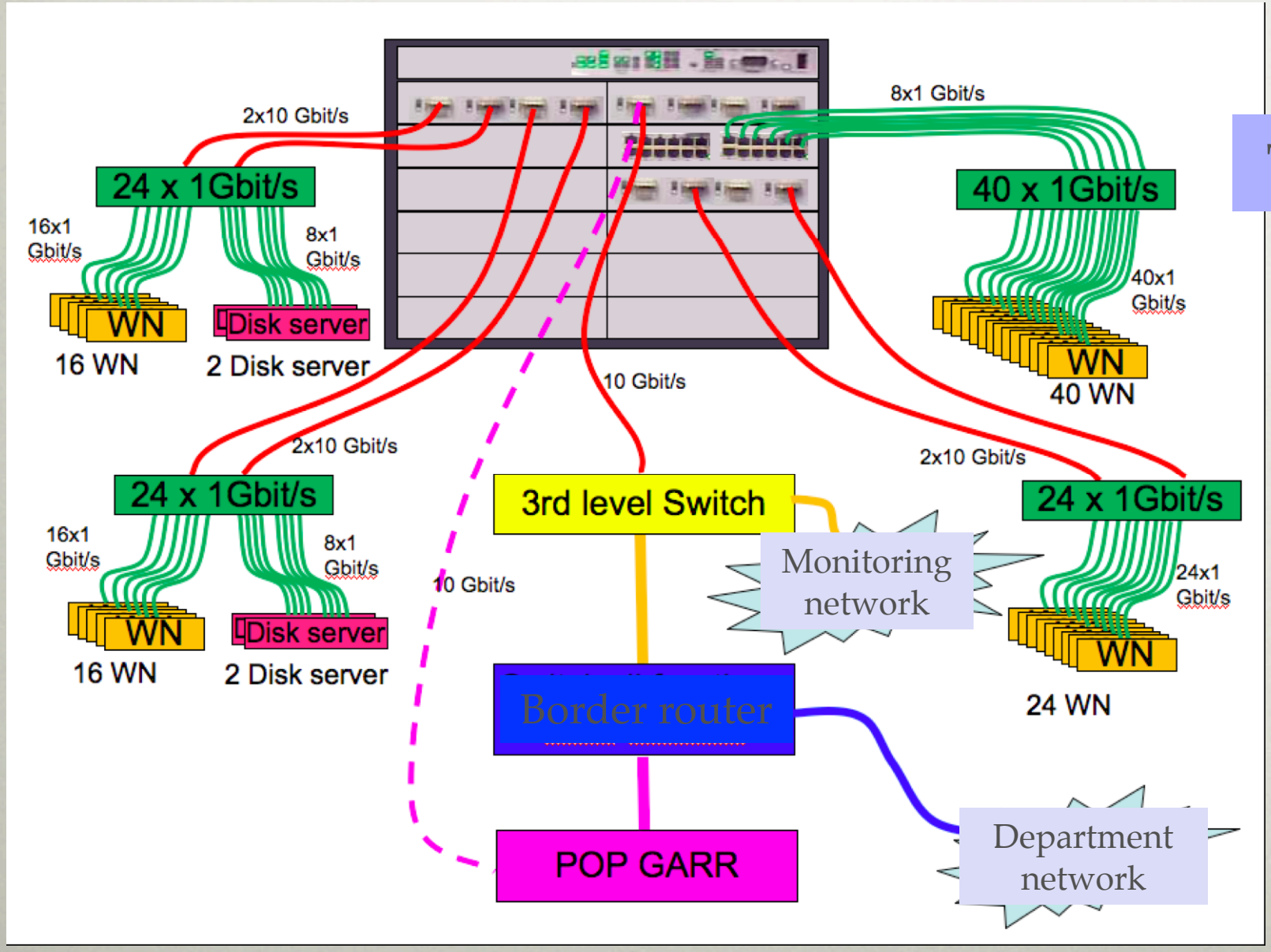
Logical View



Network infrastructure

- One of the most important pieces of the infrastructure is the network:
  - We tried to find a solution that maximize the bandwidth keeping the cost as low as possible => the goal is to provide ~1Gbit/s per each port without using a completely flat topology
- We took into account the typical behaviour that each node has into the farm:
  - wn usually read from storage servers
  - while storage server provide data on the network
  - so we can put WN and storage server on the same edge switch in order to better use of channel among the core switch and the edges ones

General overview of the existing farm configuration at the INFN-BARI



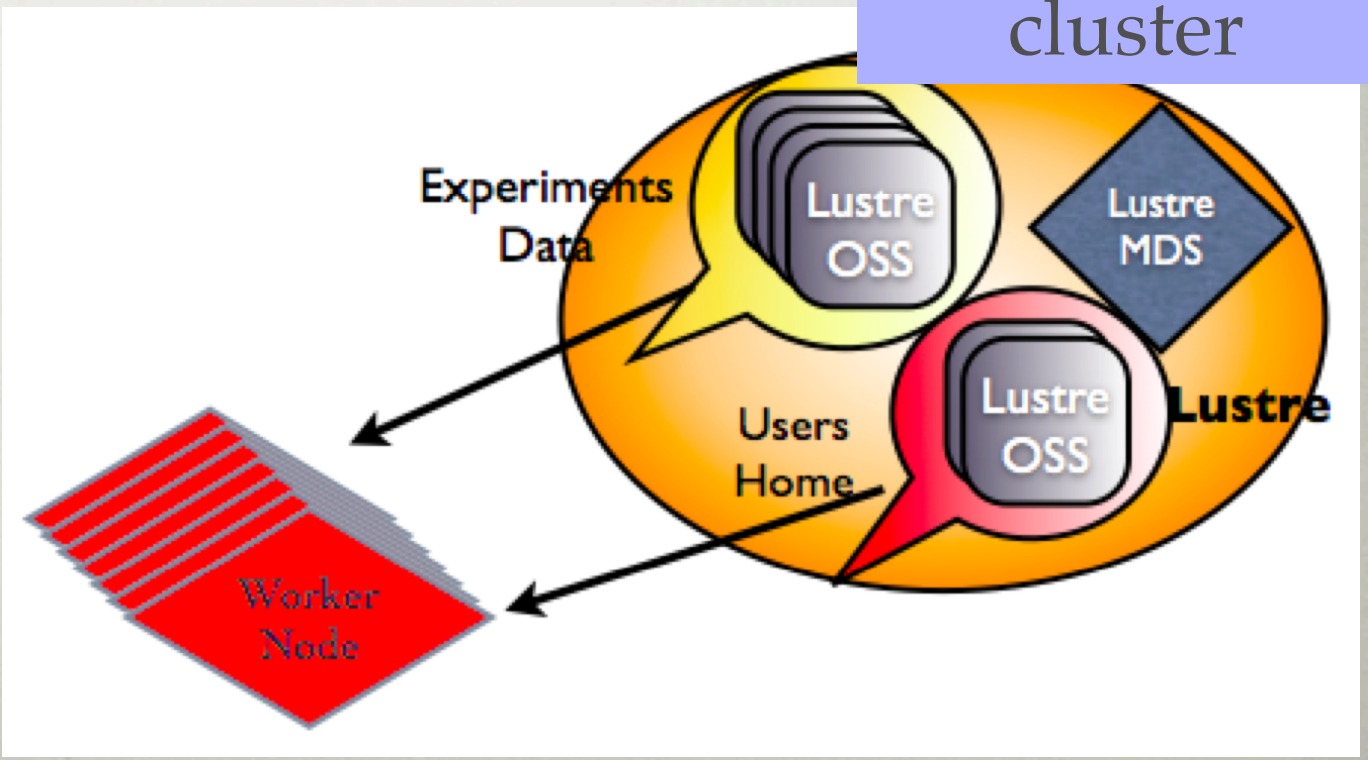
The network

Storage overview of the INFN-BARI computing centre

- The storage area is unified in order to reduce the manpower requested to keep the farm running:
  - Single infrastructure for experiment data and home
  - The storage servers could be different

- The storage area is unified in order to reduce the manpower requested to keep the farm running:
  - Single infrastructure for experiment data and home
  - The storage servers could be different

The Lustre cluster



Feature of this infrastructure: Interactive Jobs

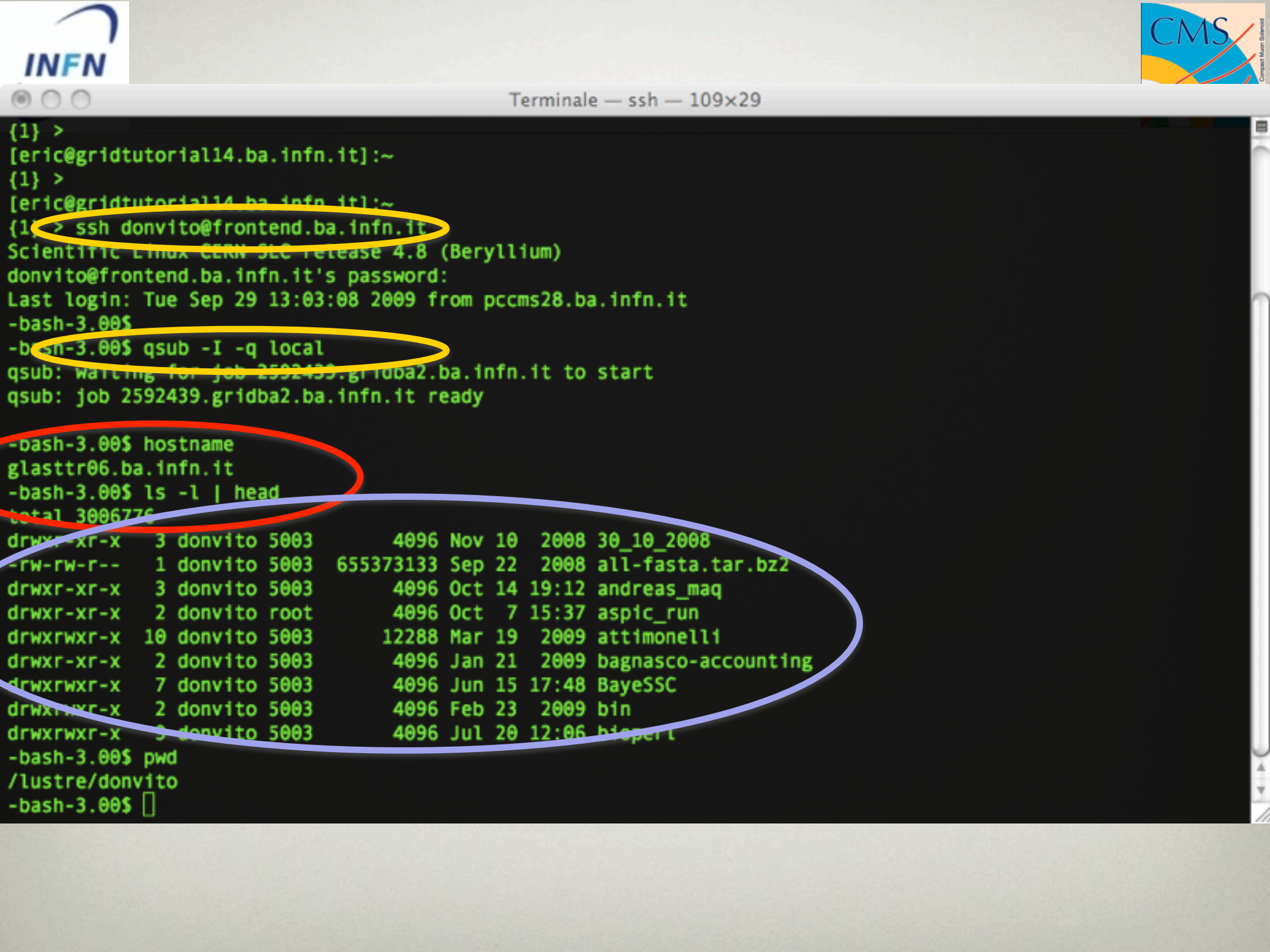
- The classic interactive cluster requires:
  - Ad-hoc configuration
  - to find the right size of the dedicated resource as the number of users increments
- It could happens that the cluster got overloaded when a lot of users are working on it

We tried another solution:

- The user can submit local batch jobs
- But it could also use available WNs as interactive resource
- The batch manager choose the right CPU to execute job
  - This guarantee to the user to have a dedicated CPU during all his work
- There is only one cluster to be configured and maintained
  - This reduced the requested manpower
- The cluster could increase in size, dinamically, depending on the user requests

Feature of this infrastructure: Interactive Jobs

- This feature is obtained by means of “Interactive jobs” on Torque
  - LFS has similar functionality too
- The maui configuration is tuned a bit in order to guarantee the high priority on those jobs
  - An home made daemon is used to be sure that the queue time of an interactive job is always less than 1min
- Interactive jobs can be “suspended” and recovered afterwards
  - Using “screen”
- No hard limit on number of concurrent “interactive sessions”
- You can run also multiCPU interactive jobs


Terminale — ssh — 109×29
{1} >
[eric@gridtutorial14.ba.infn.it]:~
{1} >
[eric@gridtutorial14.ba.infn.it]:~
{1} > ssh donvito@frontend.ba.infn.it
Scientific Linux CERN SLC release 4.8 (Beryllium)
donvito@frontend.ba.infn.it's password:
Last login: Tue Sep 29 13:03:08 2009 from pccms28.ba.infn.it
-bash-3.00$
-bash-3.00$ qsub -I -q local
qsub: job 2592439.gridba2.ba.infn.it ready
-bash-3.00$ hostname
glasttr06.ba.infn.it
-bash-3.00$ ls -l | head
drwxr-xr-x 3 donvito 5003 4096 Nov 10 2008 30_10_2008
-rw-rw-r-- 1 donvito 5003 655373133 Sep 22 2008 all-fasta.tar.bz2
drwxr-xr-x 3 donvito 5003 4096 Oct 14 19:12 andreas_maq
drwxr-xr-x 2 donvito root 4096 Oct 7 15:37 aspic_run
drwxrwxr-x 10 donvito 5003 12288 Mar 19 2009 attimonelli
drwxr-xr-x 2 donvito 5003 4096 Jan 21 2009 bagnasco-accounting
drwxrwxr-x 7 donvito 5003 4096 Jun 15 17:48 BayeSSC
drwxr-xr-x 2 donvito 5003 4096 Feb 23 2009 bin
-bash-3.00$ pwd
/lustre/donvito
-bash-3.00$

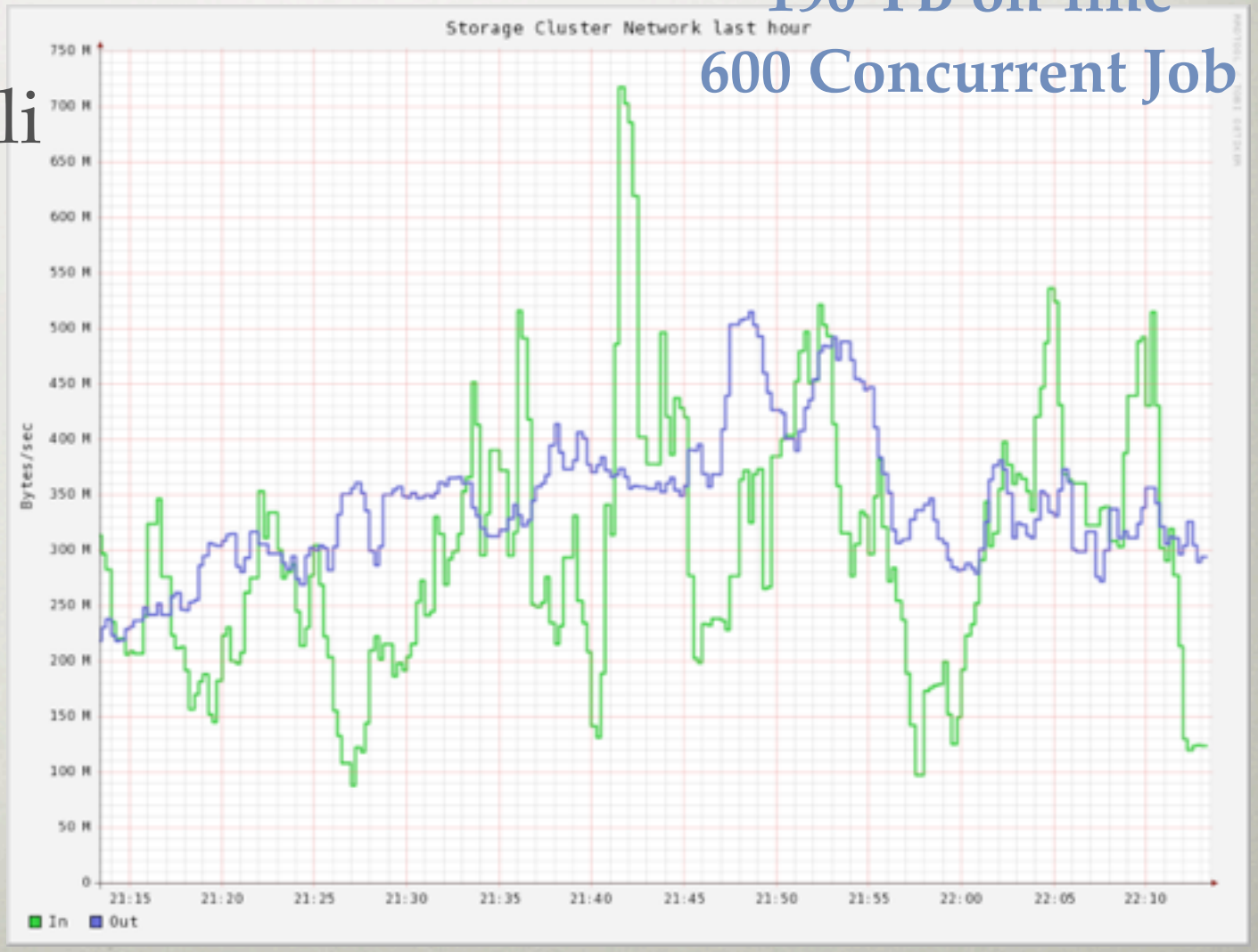
Feature of this infrastructure

- The "/store/user/crab_name" data are available as a local file-system with posix access
- User can access/write/modify those data
- Or publish locally produced data on DBS/phedex under the standard paths /store/…
- Use of default ACLs on storage areas:
  - setfacl -d -m g:cms:rw /lustre/cms/store
  - setfacl -d -m u:defilippis:rw /lustre/cms/store/user/ndefilip/
- It is possible to use CRAB local PBS submission for fast high-priority tasks using the tier2 CPUs
- Lustre caches accessed files on the client memory: running many times on the same files do not requires reading data from the disk servers each time
- It is possible to mount (i.e. via sshfs) the "/lustre" storage on its own desktop
- We do daily backup (in a separated disk server) of a small critical area (~50GB for each user)

Statistics of the infrastructure

- 88 utenti locali registrati
- > 4500 job locali eseguiti al giorno
  - > 1500 per CMS
- 2000 login per mese

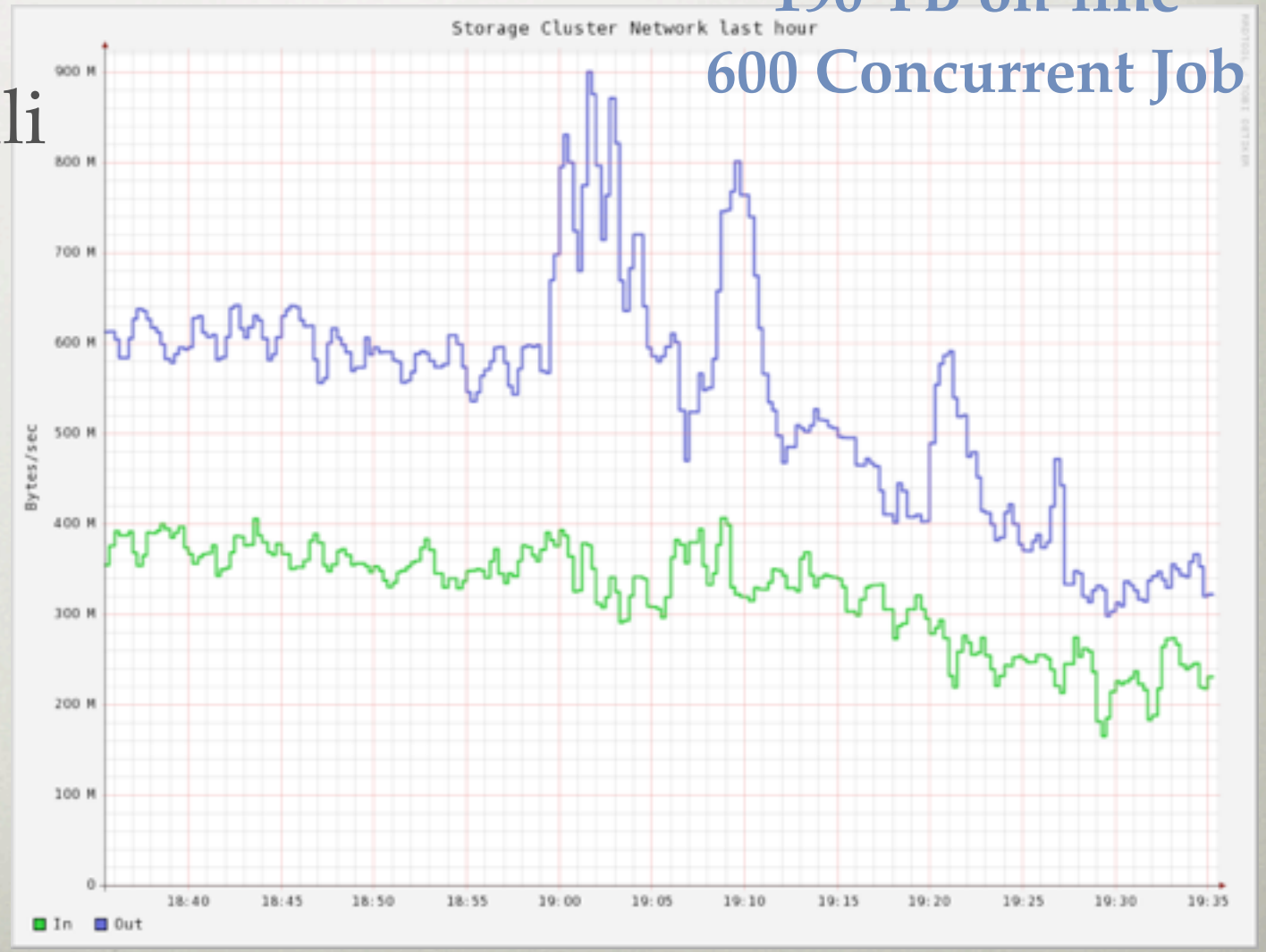
5 disk servers
4x1Gbit/s each
190 TB on-line
600 Concurrent Job



Statistics of the infrastructure

- 88 utenti locali registrati
- > 4500 job locali eseguiti al giorno
  - > 1500 per CMS
- 2000 login per mese

5 disk servers
4x1Gbit/s each
190 TB on-line
600 Concurrent Job



Future works

- Using the same infrastructure we will try to exploit PROOF *[This will be of interest for both Alice and CMS local users]*
  - Jobs will be submitted to torque batch system and act as pilot jobs:
    - they will connect to PROOF master and execute the task
  - The files will be accessed with the standard access patterns
    - Lustre for CMS and Xrood for Alice
- Working on reliability and fault tolerance for some highly critical storage area

Acknowledgement

- Alice: Andrea Dainese, Antonio Franco, Nico Di Bari

- Atlas: David Rebatto, Roberto Agostino Vitillo

- CMS: Claudio Grandi, Giacinto Donvito, Vincenzo Spinoso