Center of Excellence MIUR-Univ. Padova Workshop

Padova, Friday 29 October 2004

Toward a WoldWide Physics Analysis Framework for LHC Experiment

Stefano Lacaprara

Stefano.Lacaprara@pd.infn.it

INFN and Padova University

Outline

- What is analysis,
- What is needed fo analysis: Data and resources,
- Different approaches,
- Possible Workflow,
- Concluding comments

What is analysis

- An analysis is
  - a **user-defined job**,
  - using **private code**
  - on top of some **existing framework**,
  - which **access available data**
  - and produce some kind of **output**
  - which contains a **higher level of data reduction** compared with the input.
  - In general analysis is a chaotic, non-organized task, carried on concurrently by many independent users.

What is needed

- Data access
- Resources
- Framework for application
- Infrastructure to prepare job (including job cluster - see after -)
- Monitoring and bookkeeping
- Output management, retrieval, publication,...

Data

- How much data to analyze for a typical HEP application?
- Atom is "event": $p - p$ collision
- 1 event $\sim 1\ MB$ (RawData) $+ \sim 1\ MB$ higher level reconstructed objects
- Resources to reconstruct one event:
  - First level reconstruction $\sim$ min/ev, $1/2\ GB$ RAM, output stored
  - Higher level reconstruction typically faster
- Not really much! So, where is the problem?

How many events do we analyze??

Data (II)

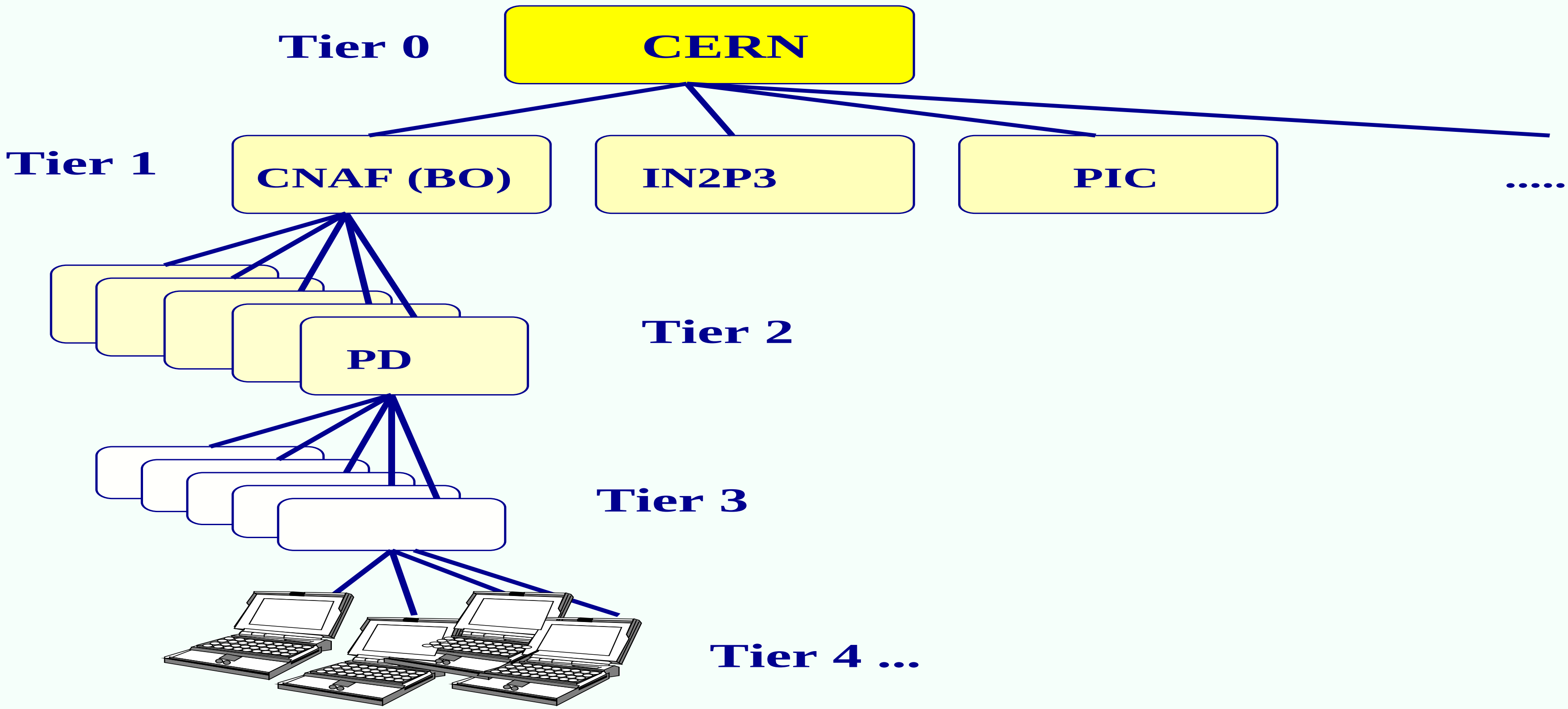
- LHC: $40\ MHz$
- Trigger - first, on-line selection- down to $\sim 100\ Hz$
- 1 LHC year: $10^7\ s$
- 10^9 events per year $\Rightarrow 1\ PB\ ==\ 1000\ TB$

- Plus simulated events... Today we have $\sim 10^8$ simulated events
- Moreover not just one user, but $\mathcal{O}(1000)$

How to deal with this??

Data (III)

- Distributed analysis approach (GRID)
- Multi Tier hierarchical structure for data and analysis
- Each tier-n contains less and less data: used by regional users



What is Data?

- Multi-tier data
  - Raw data (as read-out from CMS)
  - Reconstructed hits, calorimeter cells, ...
  - Reconstructed high level objects (tracks, clusters, ...)
  - Physical objects (electrons, muons, jets, ...)
  - Composed physical jets ($Z \to \mu\mu$, $H \to ZZ\mu\mu ee$, ...)
  - Physical distribution (histograms, ...)
  - ...
  - Publications!
- Different physics analysis access different data tier
- In addition: non event data (calibration, alignment, geometry...)
- Data Provenance
- Crucial aspect! Must know always how a particular event have been processed, reconstructed, which calibration, which reconstruction program, version etc...

What is Data? (II)

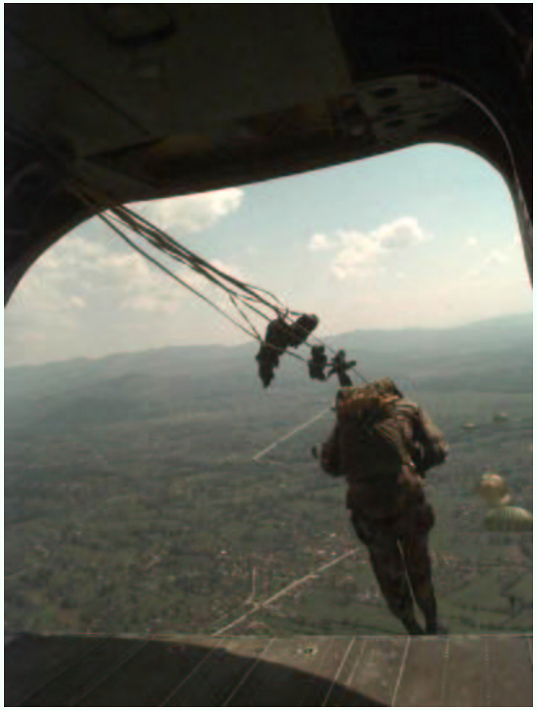
- Typical physicist access data at Dataset level
- Dataset Key element for data model: collection of events with common feature (eg taken in a given period, pre-selected with given topology, etc...)
- Need to follow abstract user request (*"give me all event with* 4 *muons in the final state"*) down to real data
- Data is distributed in files, user does not want to knows about it, want to access events, or event collection
- Large use of MetaData at various level to define abstract information about data to answer user request
- Multi level catalogs to identify which files (or fraction of) will be actually accessed by application
  - Dataset catalog: abstract, user oriented
  - File catalog: low level, application oriented

Distributed Data

- Big complication comes from data distribution approach!
  - Data can be anywhere (Tier-0, Tier-1,2,n)
  - Data is typically replicated in different location (also for redundancy)
- For effective usage of distributed resources and data need a match between the two
- **Resource Broker** accept abstract user request and match the request with available resources (computing elements CE) and data availability (storage element SE)
- enforce a *soft* locality of data: send jobs close to the data
- Soft: sometime is better to move data to job... Big problem in balancing the two approaches!
  - User may want to replicate data for efficient use (laptop)
  - Need Replica tools and catalogs

Access to resources

- GRID middle-ware
  - Remote access, authorization, authentication, ...
- How to use the resources (CE)?



- *Paratrooper* approach
- The job carries with him everything which is needed
- Data, software, infrastructure, environment,...
- How much is needed?

Access to resources (II)

- Pre-allocation approach
  - Distribute data on Tier-n according to some schema and priority
  - Pre-install on remote resources all the infrastructure analysis job will need (sw, env, ...)
  - Publish info about resource availability so that resource broker can match offer and domand
  - Send with job only your analysis application
- Pilot approach
  - Don’t trust fully what resource publish...
  - Small testing application lands on remote resources
  - Check if everything is ok, prepare environment for true application
  - Pull real analysis application and run it

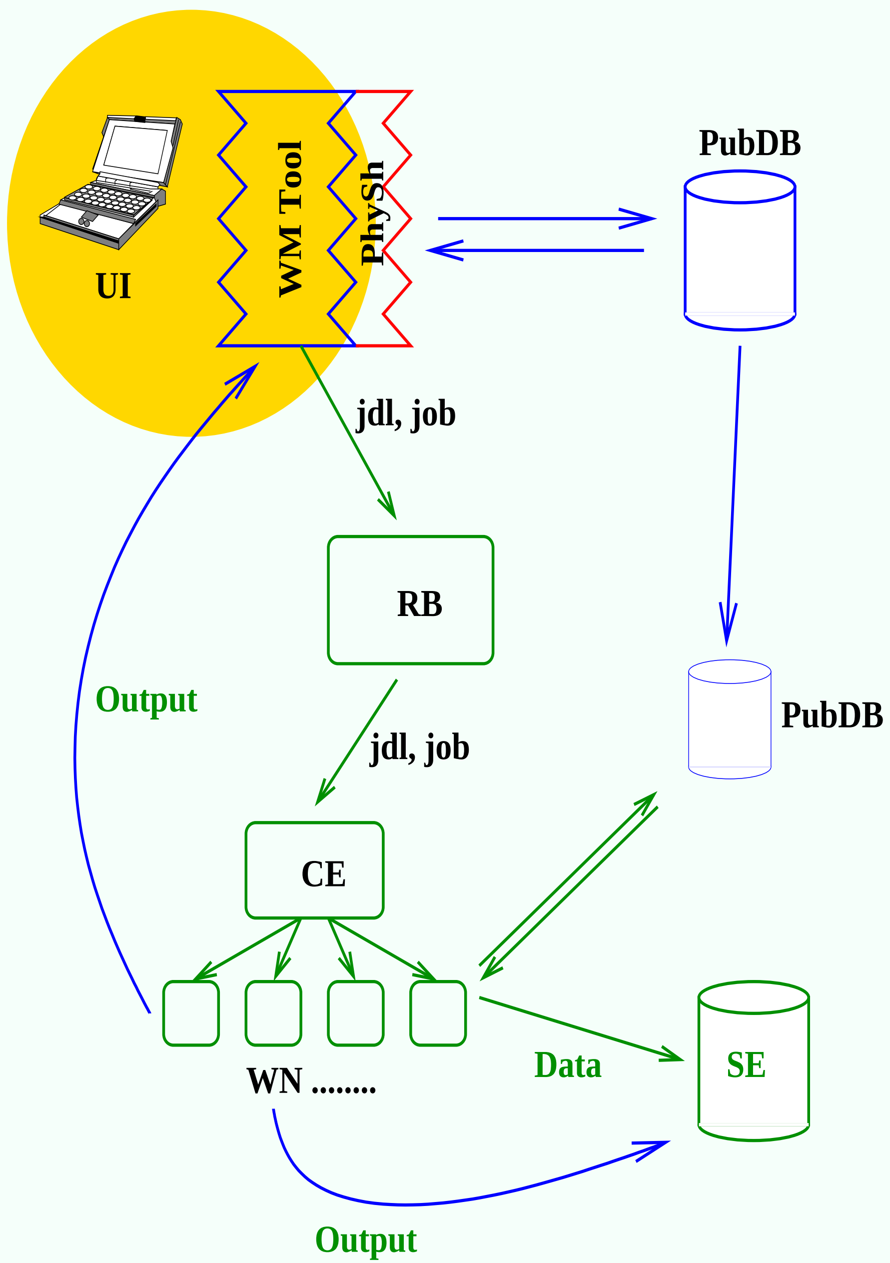
Job clustering

- Clustering (aka “poor man parallelization”)
  - Events are independent
  - Analysis job access many event can analyze/reconstruct them independently and then merge the results
  - Effective use of resources split the dataset in small chunks
  - Analyze every chunk with independent CPU (also on different site!)
- Do use large farm of processors (with common network and data storage) rather than large parallel processor

Job clustering

- Parallel analysis of single event not pursued
  - CPU time/event not so big!
  - Event cannot be easily separated in independent sub-events
  - Cross link between sub-events important
  - Big fluctuation in CPU time for reconstruction/analysis of sub-events
  - Felt as “too complex” for a physicist-lent-to-computer-science approach...

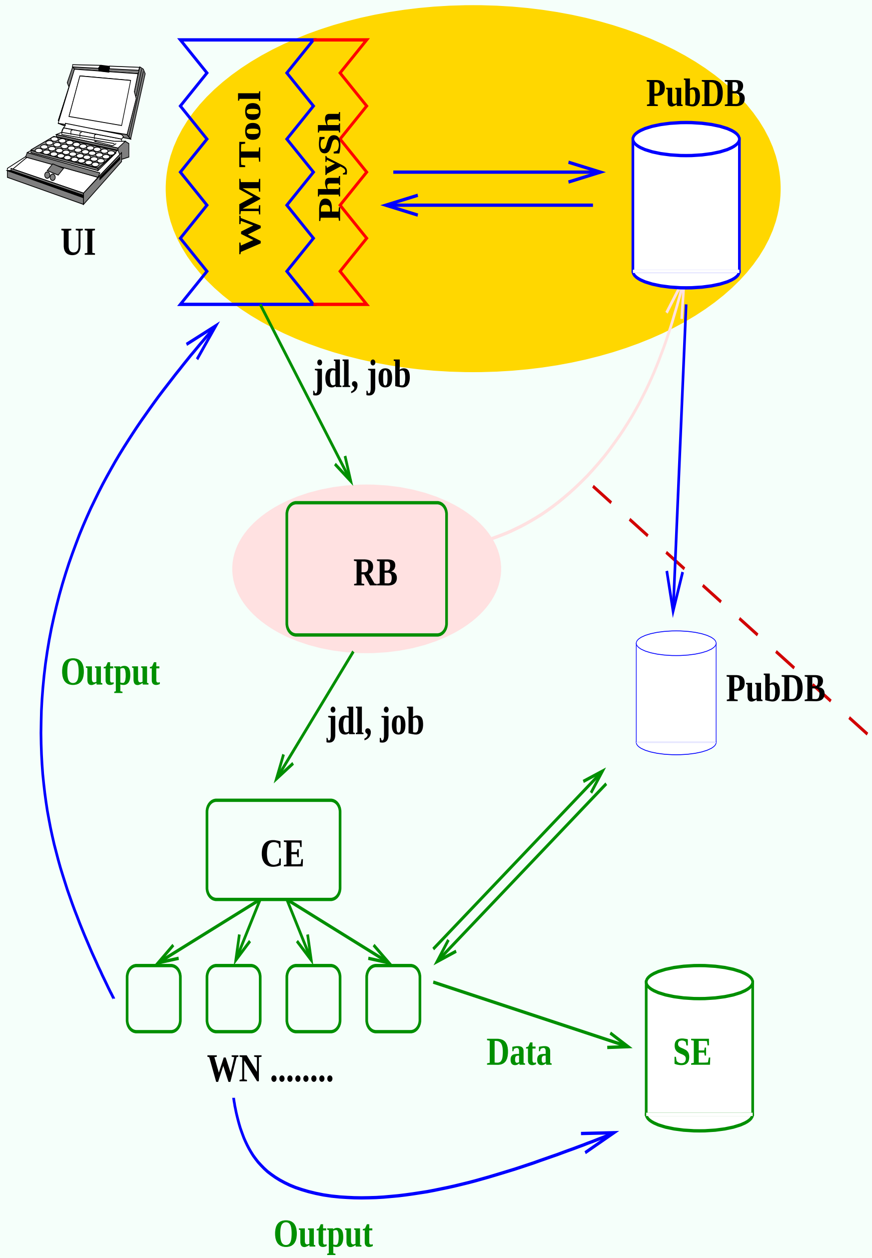
Workflow



Schema of workflow

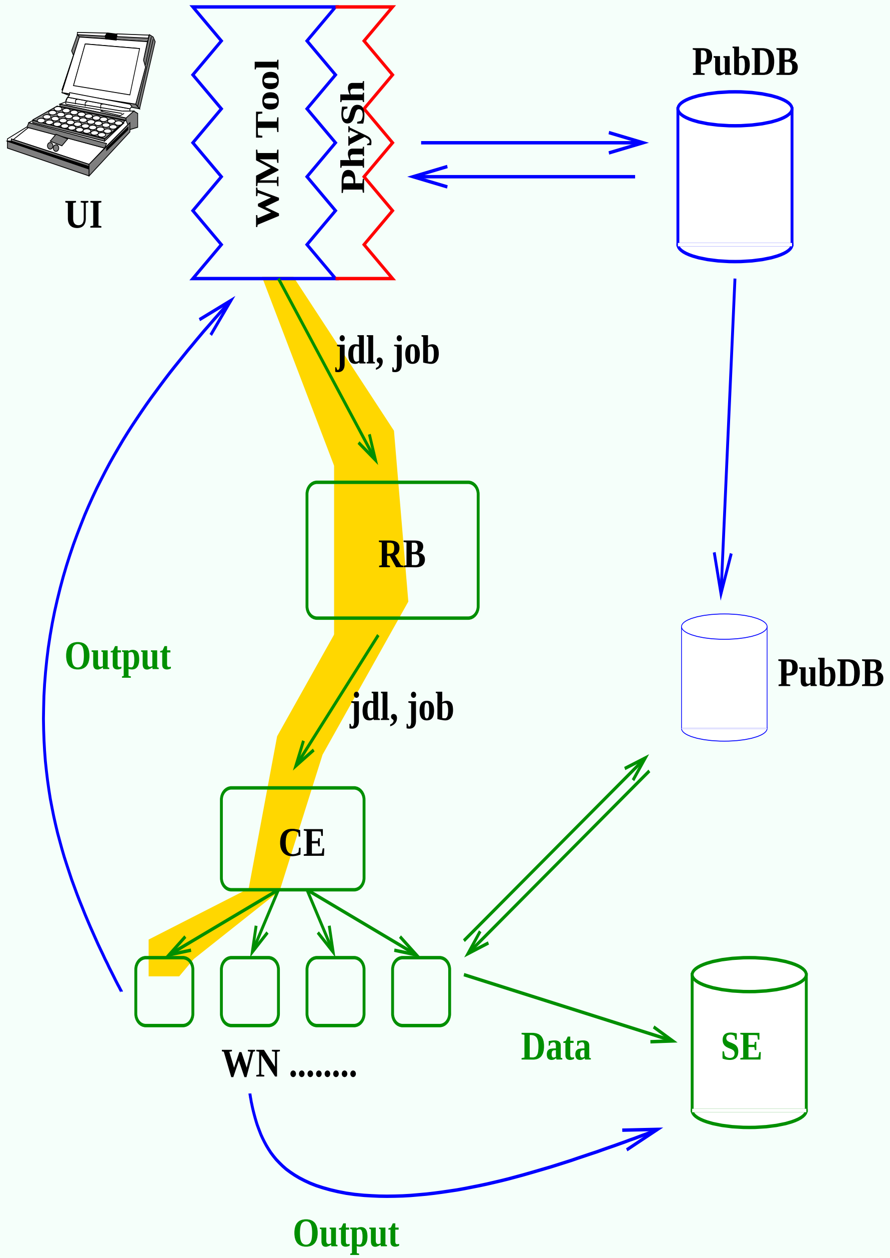
- **UI: User Interface** human access to GRID resources
- Computer (can be you your laptop) with proper middle-ware for authentication and access to GRID
- User develop and test his code on local node, accessing local data
- Want to submit private code to access a given Dataset

Workflow (II)



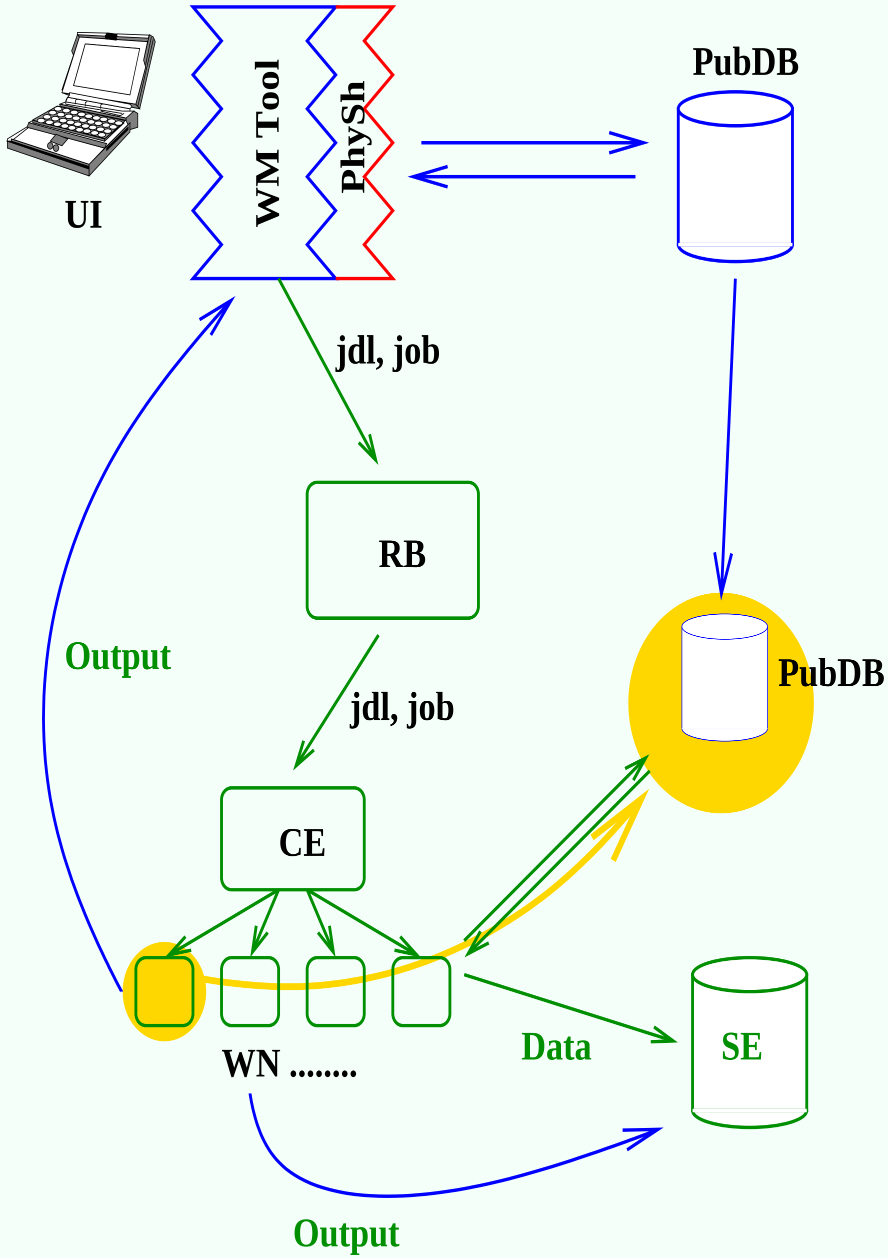
- First query to Dataset catalog to discover available datasets
- Dataset Discovery
- Resolve abstract request to concrete location: *Dataset XYZ is in Padova and CERN*
- Foresee dataset splitted into n different sites ($1/2$ in PD, $1/2$ in Madagascar)
- Put information about dataset availability on Job Description MetaData
- Perform job splitting according to user requirements and data distribution

Workflow (III)



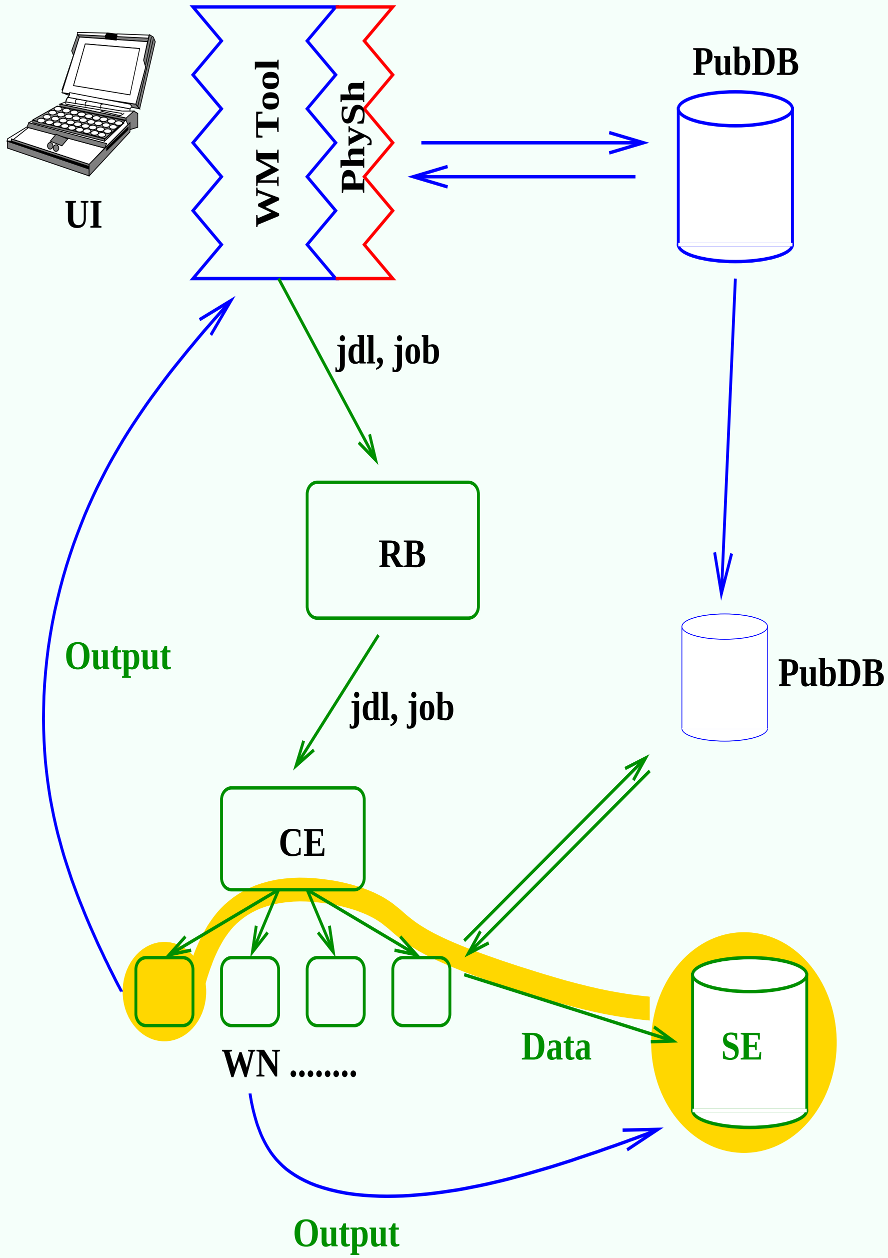
- Submits jobs to Grid Resources
- Job land to CE according to jdl specification
- Uses pre-installed “official” sw plus private libraries
- Complication for job clustering:
  - Want to send just once private stuff
  - Best splitting should also take into account resources available

Workflow (III)

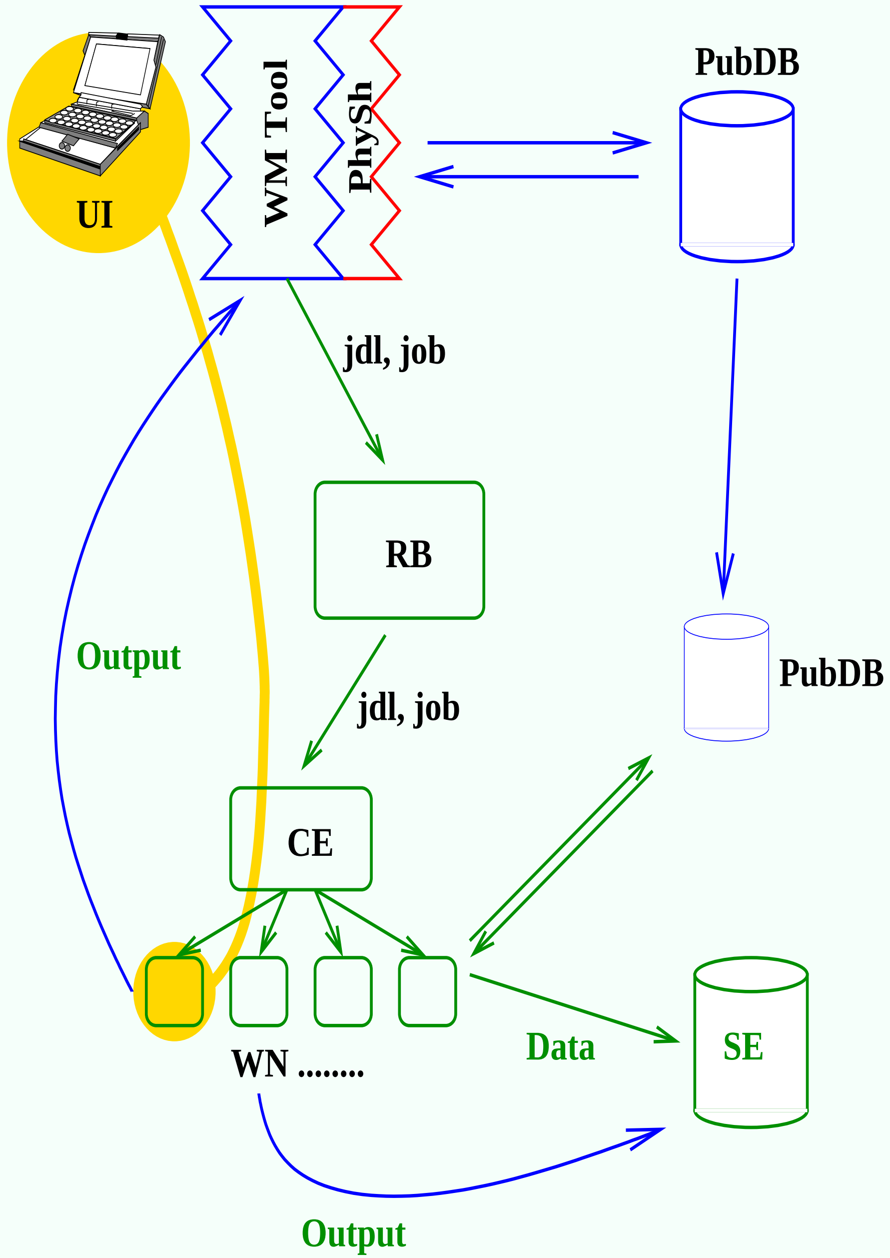


- Submits jobs to Grid Resources
- Job land to CE according to jdl specification
- Uses pre-installed “official” sw plus private libraries
- Complication for job clustering:
  - Want to send just once private stuff
  - Best splitting should also take into account resources available
- Job contact locale (to job) database for file catalog
- Here the abstract “event” request is translated into “file” request

Workflow (III)

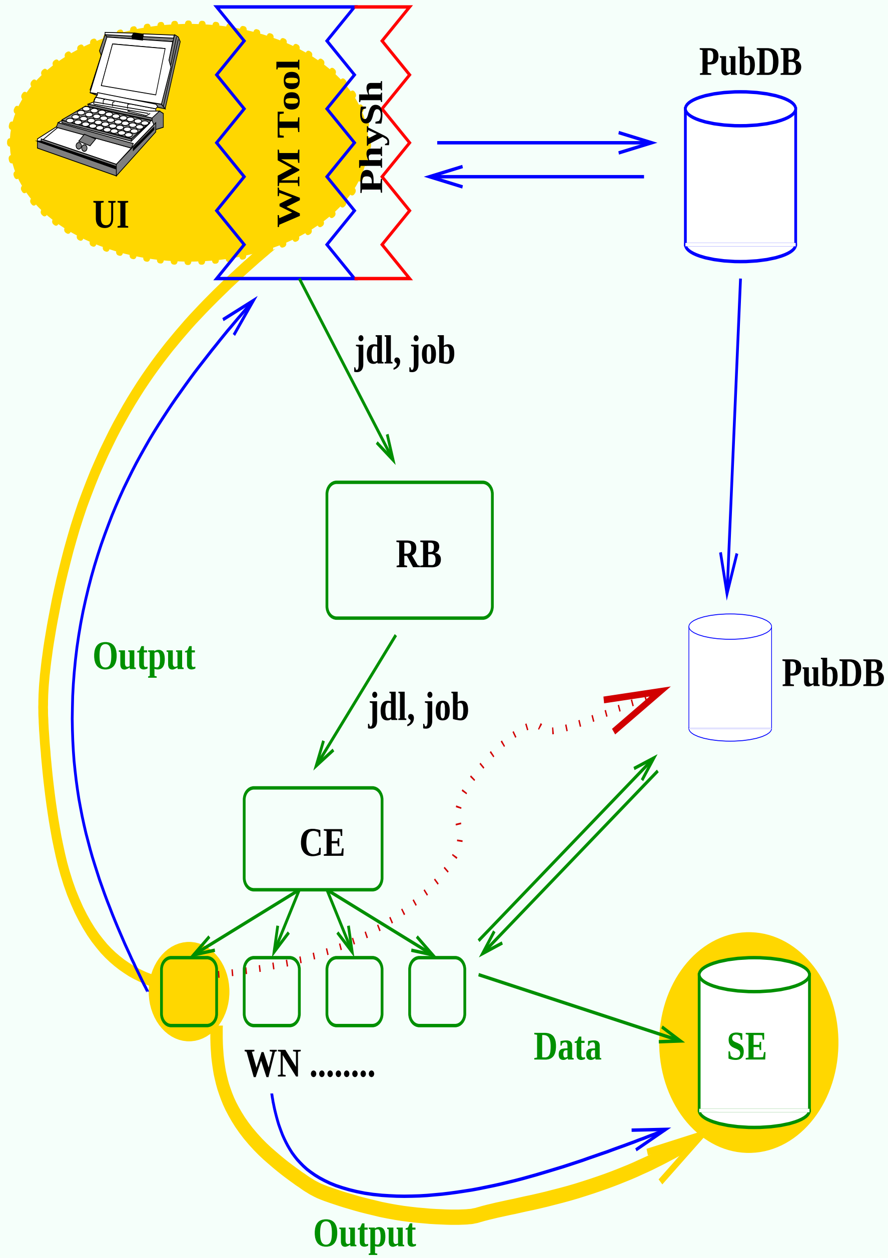


- Submits jobs to Grid Resources
- Job land to CE according to jdl specification
- Uses pre-installed “official” sw plus private libraries
- Complication for job clustering:
  - Want to send just once private stuff
  - Best splitting should also take into account resources available
- Job contact locale (to job) database for file catalog
- Here the abstract “event” request is translated into “file” request
- Run the executable accessing local data
- Or copy locally data (if requested) and access it
- definition of *local* depends on bandwidth and latency



- Monitoring of applications
- Bookkeeping (for provenance, etc...)

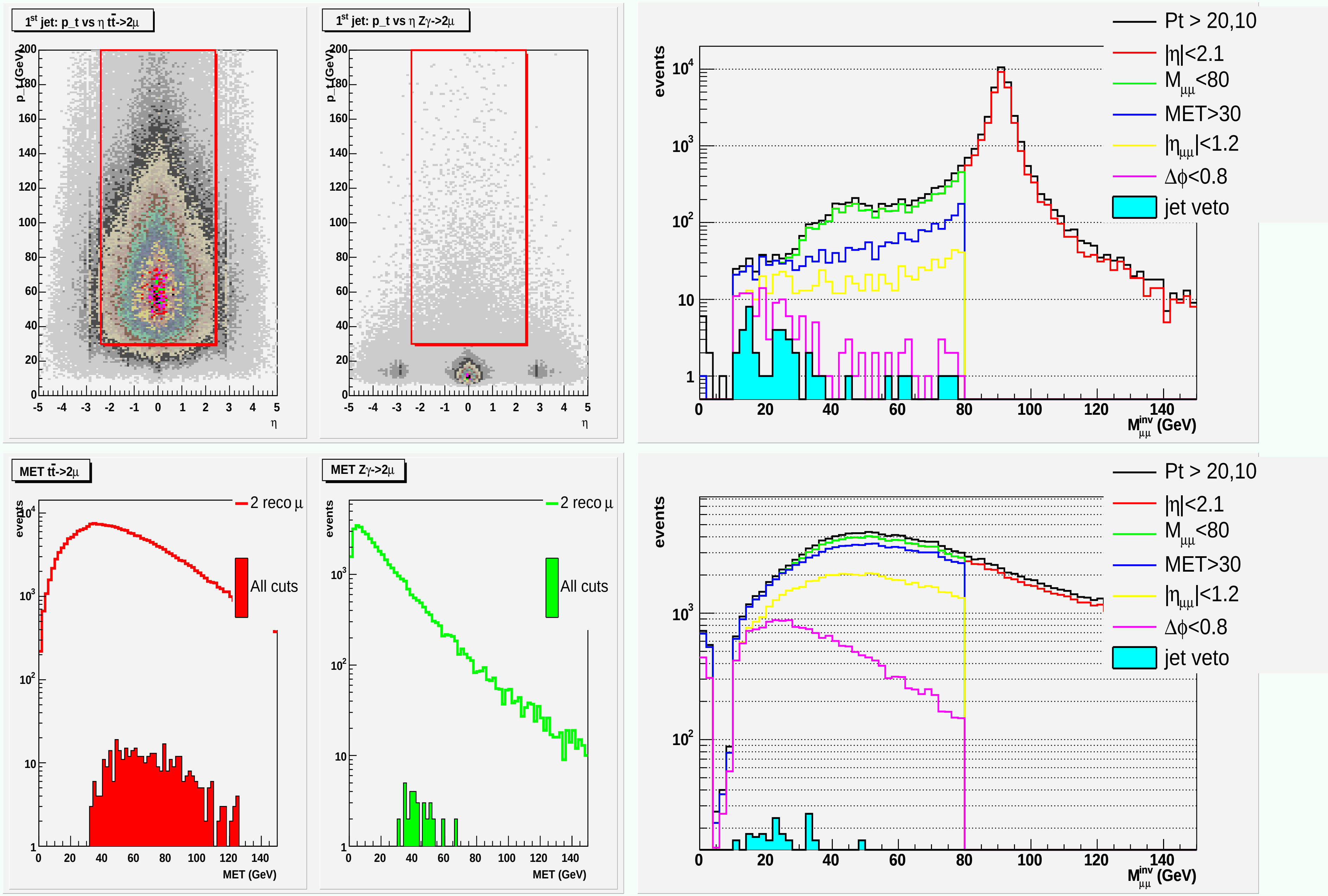
Workflow (IV)



- Monitoring of applications
- Bookkeeping (for provenance, etc...)

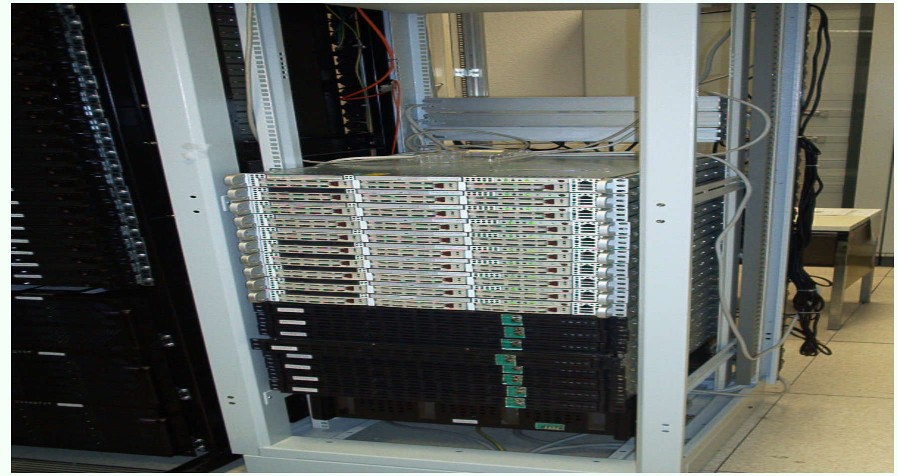
- Job output produced by executable sent back to user
- Or saved on remote resource for later distributed access
- Eventual publication on group wide usage and bookkeeping

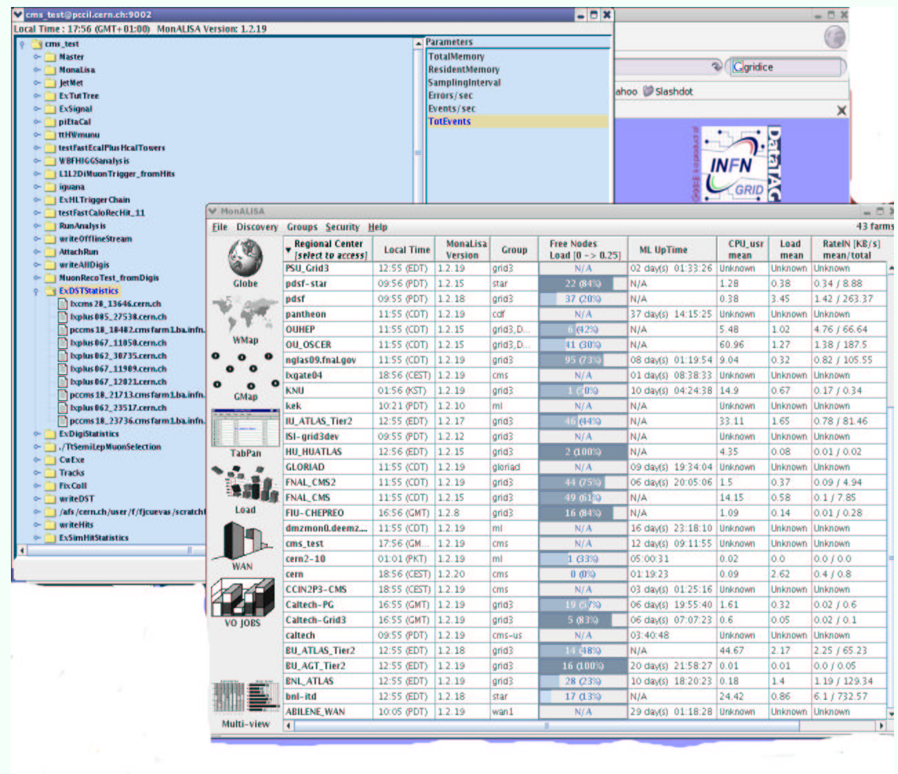
It works!


1st jet: p_t vs η tt̄->2μ
1st jet: p_t vs η Zγ->2μ
p_t (GeV)
η
events
Pt > 20,10
|η|<2.1
Mμμ<80
MET>30
|ημμ|<1.2
Δφ<0.8
jet veto
Mμμinv (GeV)
MET tt̄->2μ
MET Zγ->2μ
2 reco μ
All cuts
MET (GeV)

Monitoring

MonALISA
INFN - GridICE - Grid Monitoring S...
INFN GRID
GridICE
the eyes of the Grid
DataTAG
Site view
VO view
Job Monitoring
Geo view
Gris view
Help
about
Select Site
and/or Role
Show
Full View
pd.infn.it
grid031.pd.infn.it RB UpTime: 50-20:32 Load: 0.0-0.0-0.0 Files.: 3377 Socket: TCP(35) UDP(13) FS NA PA Full
prod-rb-01 RB UpTime: 20-21:33 Load: 0.0-0.0-0.0 Files.: 3041 Socket: TCP(13) UDP(12) FS NA PA Full
gridit001.pd.infn.it CE UpTime: 51-3:41 Load: 0.4-0.4-0.4 Files.: 3010 Socket: TCP(82) UDP(27) FS NA PA Full
egrid-se-01.pd.infn.it SE UpTime: 50-4:26 Load: 0.0-0.0-0.0 Files.: 1049 Socket: TCP(15) UDP(7) FS NA PA Full
gridit002.pd.infn.it SE UpTime: 51-3:7 Load: 0.1-0.0-0.0 Files.: 3690 Socket: TCP(20) UDP(14) FS NA PA Full
prod-se-01.pd.infn.it SE UpTime: 20-23:2 Load: 0.0-0.0-0.0 Files.: 2974 Socket: TCP(12) UDP(13) FS NA PA Full
prod-se-02.pd.infn.it SE UpTime: 48-16:33 Load: 0.0-0.0-0.0 Files.: 536 Socket: TCP(8) UDP(13) FS NA PA Full
lxcde01.pd.infn.it WN UpTime: 43-1:3 Load: 2.0-2.0-2.0 Files.: 636 Socket: TCP(52) UDP(30) FS NA PA Full
lxcde02.pd.infn.it WN UpTime: 43-5:6 Load: 2.1-2.1-2.1 Files.: 639 Socket: TCP(52) UDP(30) FS NA PA Full
lxcde03.pd.infn.it WN UpTime: 50-22:2 Load: 2.0-2.0-2.0 Files.: 643 Socket: TCP(54) UDP(32) FS NA PA Full
lxcde04.pd.infn.it WN UpTime: 51-2:3 Load: 2.0-2.0-2.0 Files.: 639 Socket: TCP(52) UDP(30) FS NA PA Full
lxcde06.pd.infn.it WN UpTime: 50-22:33 Load: 2.2-2.1-2.1 Files.: 635 Socket: TCP(52) UDP(30) FS NA PA Full
lxcde07.pd.infn.it WN UpTime: 49-4:32 Load: 2.0-2.0-2.0 Files.: 635 Socket: TCP(52) UDP(30) FS NA PA Full
lxcde08.pd.infn.it WN UpTime: 51-0:2 Load: 2.0-2.0-2.0 Files.: 636 Socket: TCP(52) UDP(30) FS NA PA Full
prod-wn-001.pd.infn.it WN UpTime: 51-2:9 Load: 2.0-2.0-2.0 Files.: 608 Socket: TCP(47) UDP(24) FS NA PA Full
prod-wn-002.pd.infn.it WN UpTime: 51-2:40 Load: 2.0-2.0-2.0 Files.: 604 Socket: TCP(48) UDP(25) FS NA PA Full
prod-wn-003.pd.infn.it WN UpTime: 51-2:38 Load: 2.0-2.0-2.0 Files.: 604 Socket: TCP(48) UDP(24) FS NA PA Full
prod-wn-004.pd.infn.it WN UpTime: 51-2:9 Load: 2.0-2.0-2.0 Files.: 604 Socket: TCP(48) UDP(24) FS NA PA Full
prod-wn-005.pd.infn.it WN UpTime: 51-2:37 Load: 2.0-2.0-2.0 Files.: 604 Socket: TCP(48) UDP(24) FS NA PA Full
prod-wn-006.pd.infn.it WN UpTime: 50-22:38 Load: 2.0-2.0-2.0 Files.: 600 Socket: TCP(48) UDP(25) FS NA PA Full
prod-wn-007.pd.infn.it WN UpTime: 50-22:9 Load: 2.0-2.0-2.0 Files.: 639 Socket: TCP(52) UDP(31) FS NA PA Full
prod-wn-008.pd.infn.it WN UpTime: 51-1:8 Load: 2.0-2.0-2.0 Files.: 604 Socket: TCP(48) UDP(24) FS NA PA Full
prod-wn-009.pd.infn.it WN UpTime: 51-0:6 Load: 2.0-2.0-2.0 Files.: 600 Socket: TCP(48) UDP(25) FS NA PA Full
prod-wn-010.pd.infn.it WN UpTime: 51-0:40 Load: 2.0-2.0-2.0 Files.: 604 Socket: TCP(48) UDP(25) FS NA PA Full
prod-wn-011.pd.infn.it WN UpTime: 51-1:6 Load: 2.0-2.0-2.0 Files.: 604 Socket: TCP(48) UDP(24) FS NA PA Full
prod-wn-012.pd.infn.it WN UpTime: 50-22:37 Load: 2.0-2.0-2.0 Files.: 607 Socket: TCP(50) UDP(27) FS NA PA Full
prod-wn-013.pd.infn.it WN UpTime: 51-0:37 Load: 2.0-2.0-2.0 Files.: 600 Socket: TCP(48) UDP(24) FS NA PA Full
prod-wn-014.pd.infn.it WN UpTime: 50-22:36 Load: 2.0-2.0-2.0 Files.: 607 Socket: TCP(50) UDP(26) FS NA PA Full
prod-wn-015.pd.infn.it WN UpTime: 51-1:7 Load: 2.0-2.0-2.0 Files.: 604 Socket: TCP(48) UDP(25) FS NA PA Full
prod-wn-016.pd.infn.it WN UpTime: 50-23:37 Load: 2.0-2.0-2.0 Files.: 600 Socket: TCP(48) UDP(25) FS NA PA Full
prod-wn-017.pd.infn.it WN UpTime: 51-2:7 Load: 2.0-2.0-2.0 Files.: 604 Socket: TCP(48) UDP(24) FS NA PA Full
prod-wn-018.pd.infn.it WN UpTime: 6-23:2 Load: 2.0-2.0-2.0 Files.: 639 Socket: TCP(52) UDP(30) FS NA PA Full





MonALISA
File Discovery Groups Security Help
Globe
WMap
GMap
TabPan
Load
WAN
VO JOBS
Multi-view

Concluding comments

- What a hard life!
- And only to **access** data!
- Then the real physic work begins
- Is it needed?
  - Requirements: allow $\mathcal{O}(1000)$ people to access $\mathcal{O}(1)\ PB/y^r$
  - If failure: failure of all LHC.
- First LHC collision in 2007: must be ready!
- Work in progress...