The Virgo computing model: present and future plans

Gergely Debreczeni
<Gergely.Debreczeni@rmki.kfki.hu>
(KFKI RMKI, Budapest)

on behalf of the Virgo Collaboration

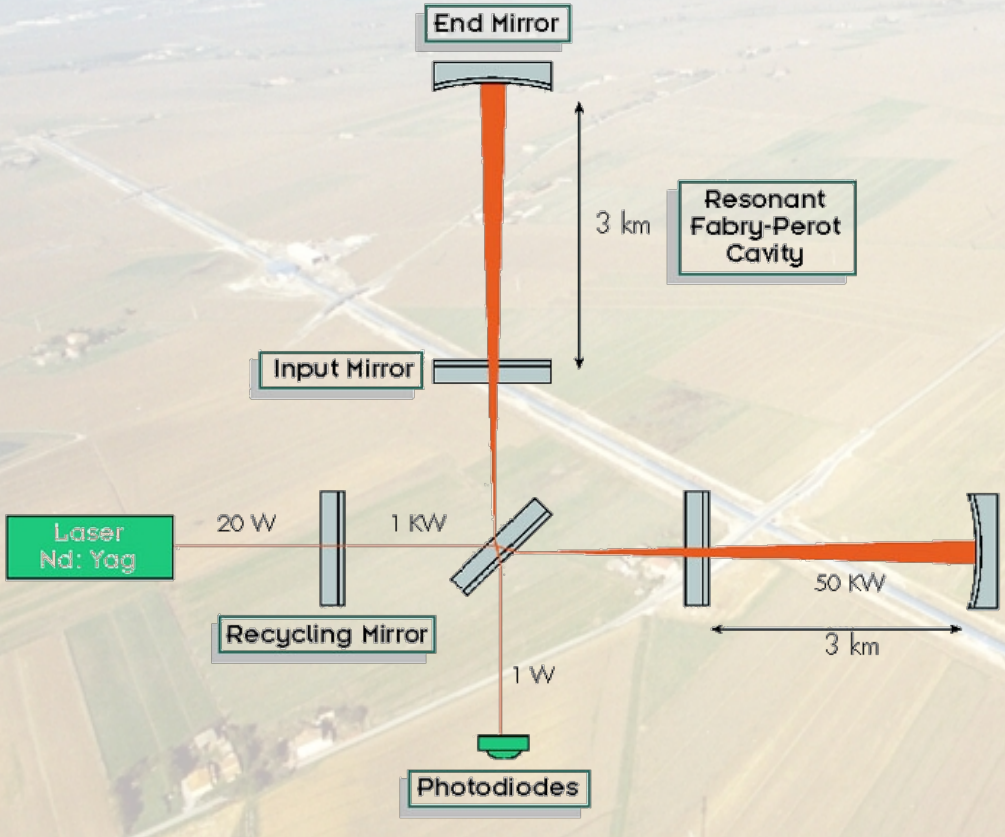
About this talk

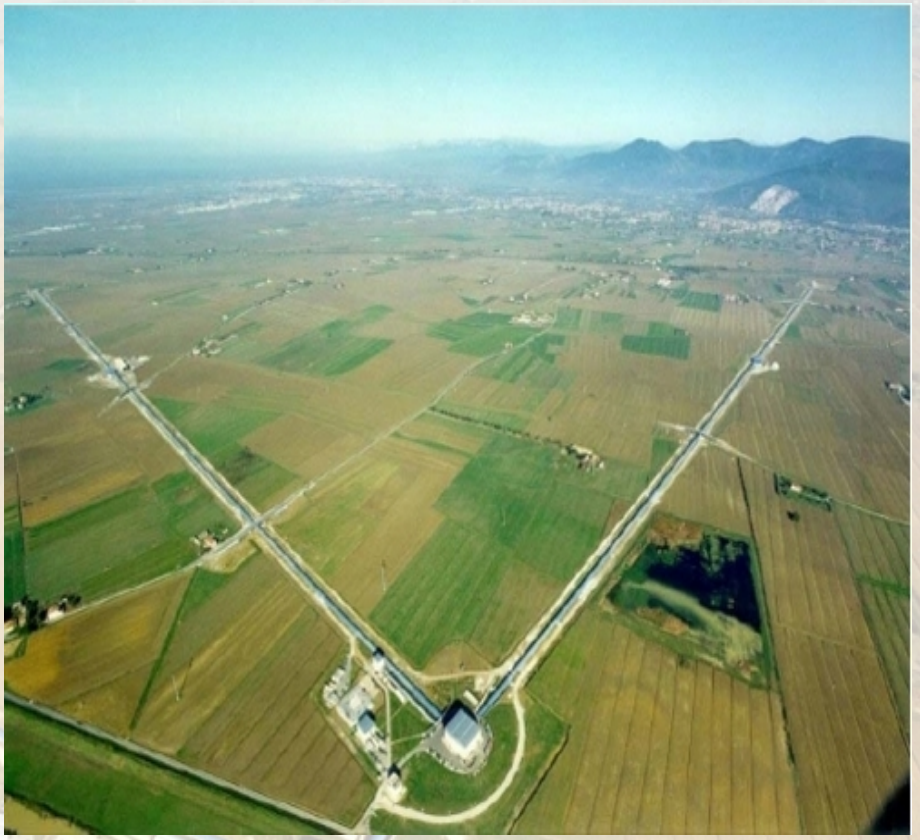
- The Virgo experiment
- Storage and computing needs
- Data transfer solutions
- Analysis on dedicated clusters
- Analysis on the Grid
- The Virgo pilot pool
- Many-core computing @ Virgo
- Future plans

The Virgo experiment

- Cascina, Italy
- 3 km-es arm length
- Fabry-Perot cavities 50/150 finesse
- 6800 m^3, 10^{-10} mbar vacuum!!

- 20 W laser
- solid concrete base, 20-50 m deep
- 1 MW power consumption
- excellent seismic isolation (10^{-10})



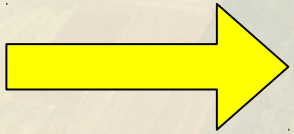


Data and computing needs

- ITF (interferometer) output sampling rate 20 kHz
- Hundreds of auxiliary channels
- Various analysises on 4 and 16 kHz
- cc. 150 day science time/year
- cc. 10.4 MB/sec data flow rate
- cc. 160 TB data / year / IF

This amount of data has to be transferred, recalibrated, downsampled, processed, analyzed (several time)

The amount of data is not overwhelming for storages available today, but the **arithmetic density** of algorithms used for the analysis varies from less significant (online analysis) to practically infinity (CW searches).



No unique/optimal computing solution for all, various approaches has to be used: such as Cluster, Grid, HPC and Many Core computing

Data transfer – solution

- **Goal:** Transfer ITF output and auxiliary channel data to permanent storage **"in-time".**

- **Currently:** Various solutions for different destinations:
  - SRB (Storage Resource Broker, Lyon)
  - BBFTP (RFC1323 implementation, CNAF) (deprecated, grid tools are now in use)
  - LDR(Ligo Data Replicator, Ligo sites).

Some of them is obsolete other not supported any more. Do the job but miss several feature, for ex. data bookkeeping.

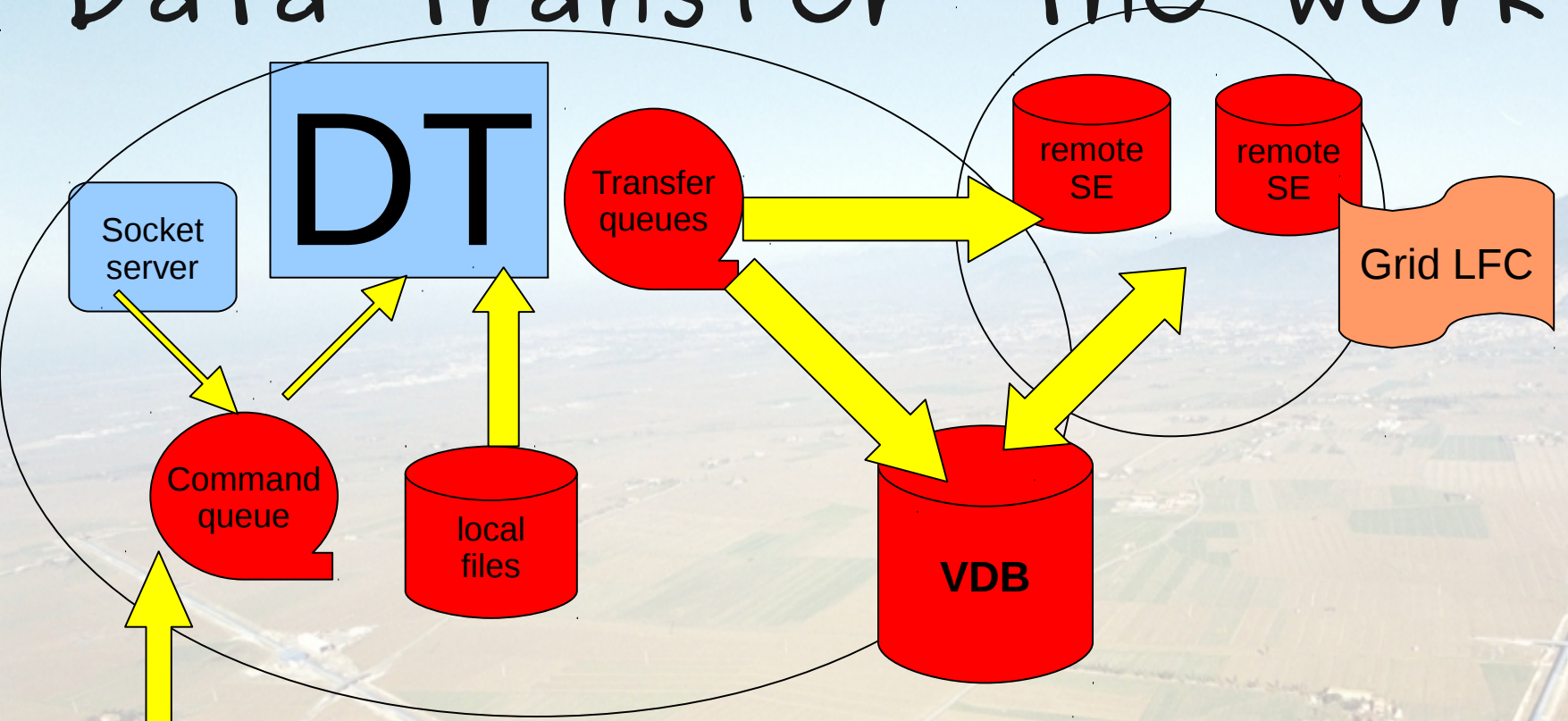
- **Use of grid tools:** FTS (File Transfer Service) was tried, but was found to be unnecessarly complicated/rigid for this task.

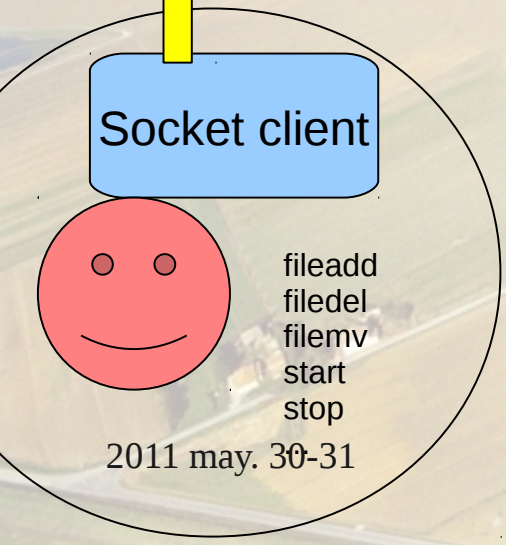
- **Solution:** lcg client tools with custom transfer and command queues (TQ/CQ), called the DT server. Under final realistic stress test! Will be used soon for Cascina – Bologna transfer.

DT server features:

- Standard interface among the most important CCs worldwide.
- LFCs provide transparent access to distributed data
- Implements communication channel with DAQ via socket server/client, new files are automatically added.

Data transfer–the workflow





- DT server features:
  - Same channel acts as user interface for manual intervention
  - Load balancing for all the endpoint(incoming/outgoing)
  - Local/remote file checksums
  - Interfaced to Grid LFC(s) (Local File Catalog) and VDB (Virgo Data Base)
  - Robustness respect to timeouts, temporal transfer failures, DT server crash (persistent FQ), lost messages (message register).

Analysis on clusters

LIGO Scientific Collaboration

VIRGO

- The Ligo and Virgo experiements closely working together, collaborations share and exchange knowledge, software, data and computing resources.

- Various analysis pipelines are running on dedicated – and so far – isolated LDG (Ligo Data Grid) clusters.

- Interconnection of clusters just happened.

- These pieces of softwares is strongly bounded to one specific batch scheduler (Condor) and require locally shared file systems which makes the port difficult to other non-Condor, non-local architectures.

The solution is

- either detach the software from Condor and make it possible to use non-local non-shared file systems with various batch schedulers (LSF, Torque, PBS, SGE, etc..)

- Create a Condor pool – EGI Grid gateway

- or set up a Pilot Pool in order to provide a homogen interface towards this hierarchical, relational workflows

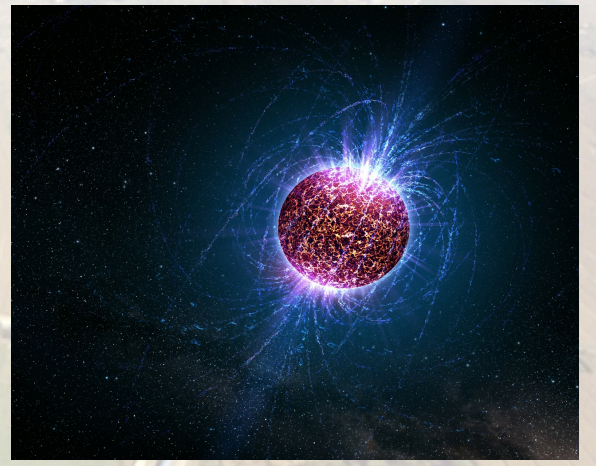
The Virgo Virtual Organisation

- Virgo is one of the 211 Virtual Organisation of the EGI Grid.
- Some Virgo analysis is running smoothly on the Grid since quite a long time !
- Ideal for CPU intensive jobs crunching "small" amount of data.

- Resources: ~34 sites, ~9000 CPU, ~50 user, several hundred TB storage

CW analysis on the Grid

- **CW (Continous Wave)** analysis searches periodic signals emitted by pulsars, isolated NS (neutron stars). In the all-sky searches we want to explore a portion of the source parameter space as large as possible (position, signal frequency and its derivatives)
- Such kind of search is computationally bound, even if very efficient (altought non-optimal in sensitivity) methods have been developed
- In particular, the Hough transform stage of the hierarchical procedure developed in Virgo is the most computationally expensive.
- The analysis scheme:



CW analysis on the Grid

- Members of the Virgo CW (Continous Wave) group ported their software for the use on Grid.
- *In this search computing power directly translates to sensitivity ... and the problem is "embarassingly paralellizable".*

Workflow:

1) Jobs submitted from a native Grid UI (User Interface) to sites supporting the Virg VO, where
2) pre-uploaded, replicated and registered input files are processed
3) Results and various outputs are also registered on Grid storage and downloaded t the UI on completition of the job group.

Advantages:

1) More resources
2) Independent of local batch systems (BQS, LSF)
3) LFNs (Logical File Name)s make it possible to forget physical storage places and architectures.
4) Experiment software installation becomes easier.

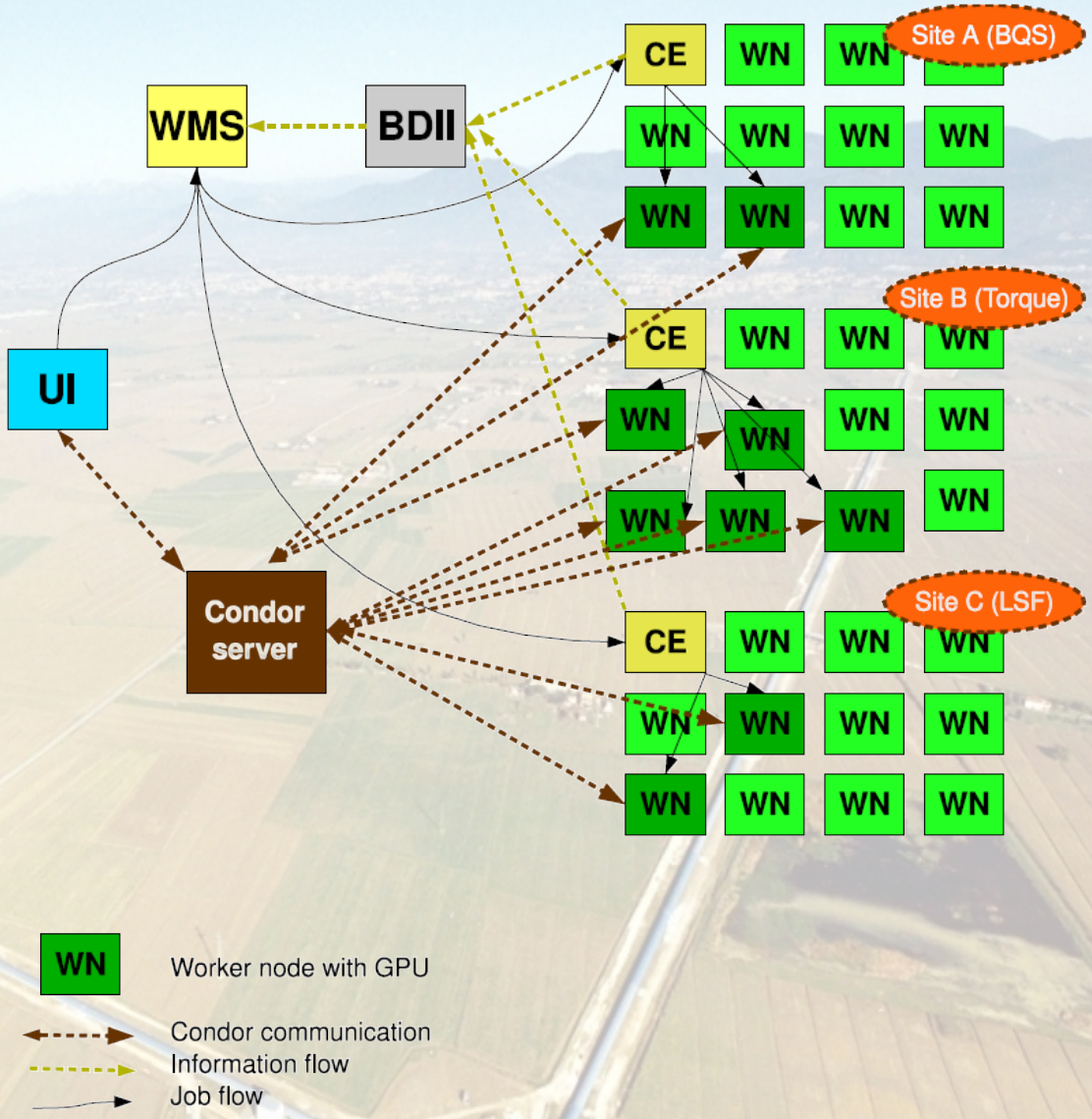
Features

1) A typical analysis run consist tens of thousand jobs running in several sites
2) A ***Supervisor*** program has been developed to manage the analysis:
    1) Basic check of Gird status and site, WMS availability
    2) Creation of submission scripts on the fly
    3) Job submission and job status monitoring
    4) Output file management
    5) Re-submission, failure handling
    6) Checkpointing in case of Supervisor crash

The Virgo Pilot Pool – I

The Virgo Pilot Pool properties:

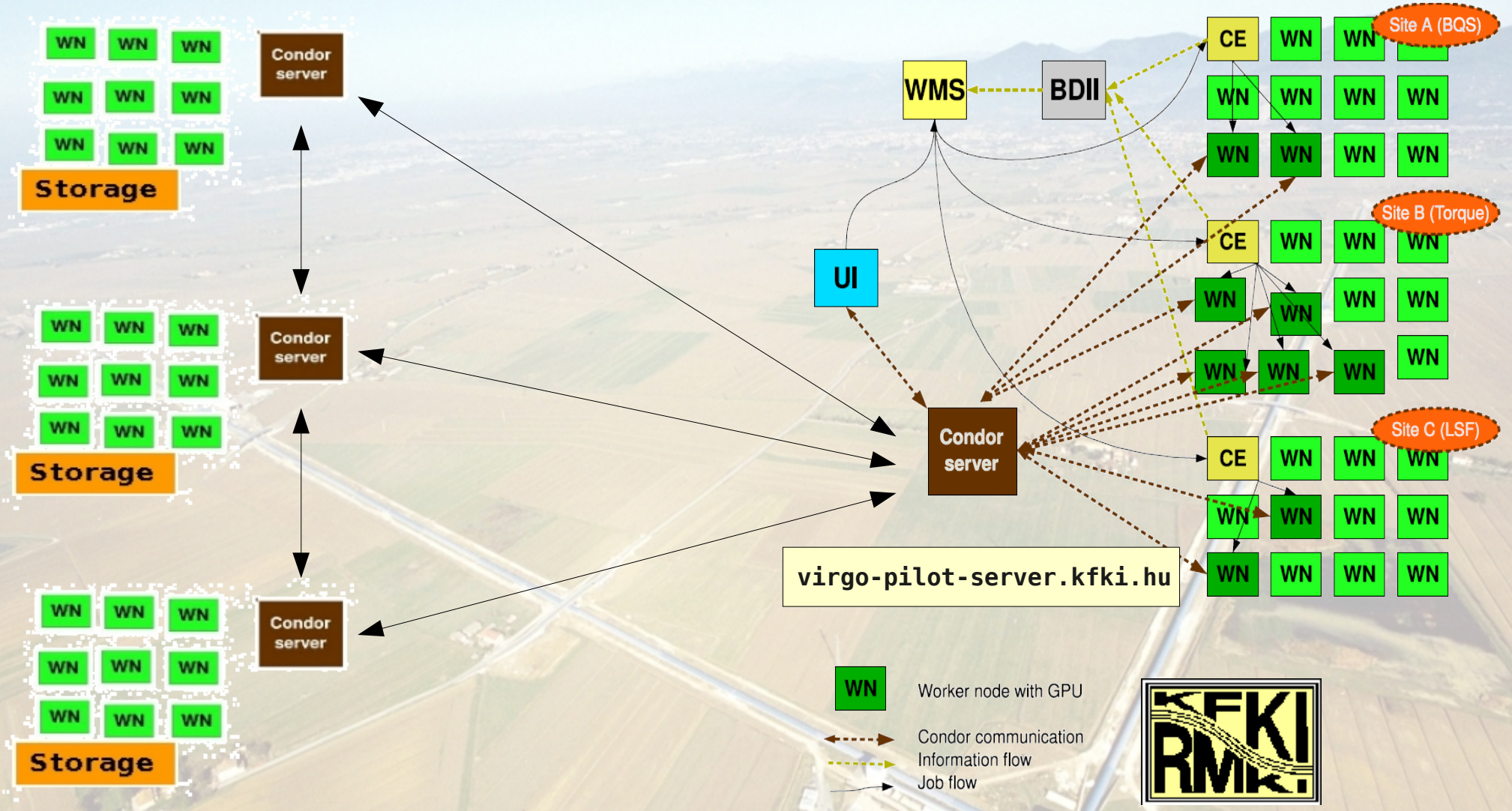
- Homogen infrastructure over the inhomogen Grid
- Less administrative interaction/delay
- User transparent mechanisms
- Low latency submission
- Global priorities
- Late-binding to resources
- No stucked-in jobs
- Improved job failure rate due to pilot prechecks
- Interactive login
- Smooth interaction interoperability with LDG/OSG.



The Virgo Pilot Pool – II

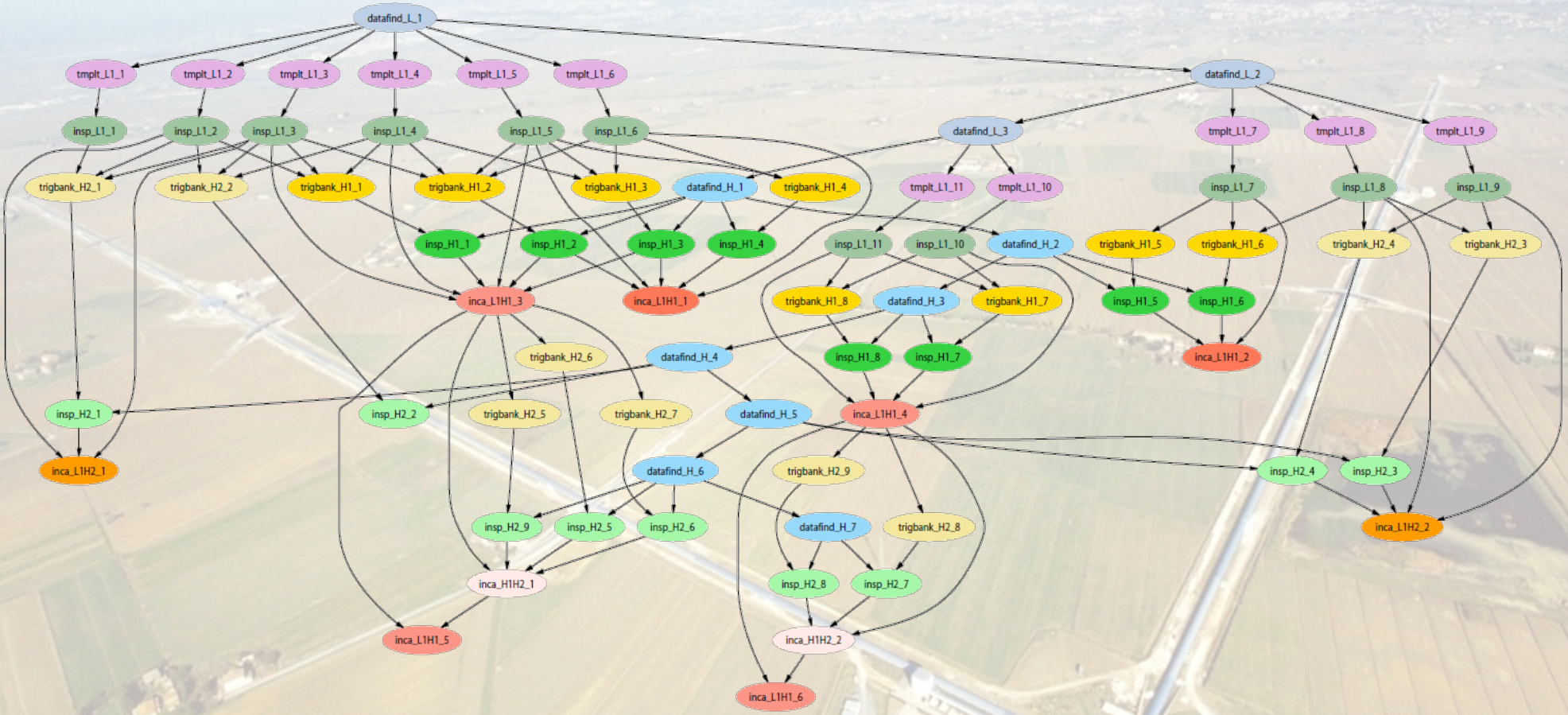
Ligo clusters

Virgo EGI sites



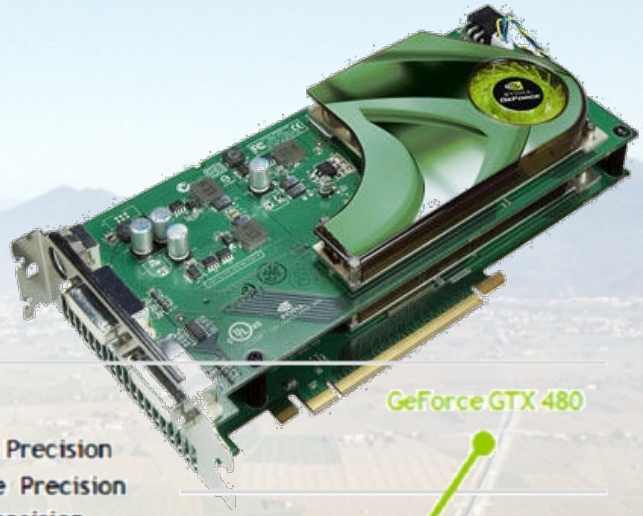
The Virgo Pilot Pool – III

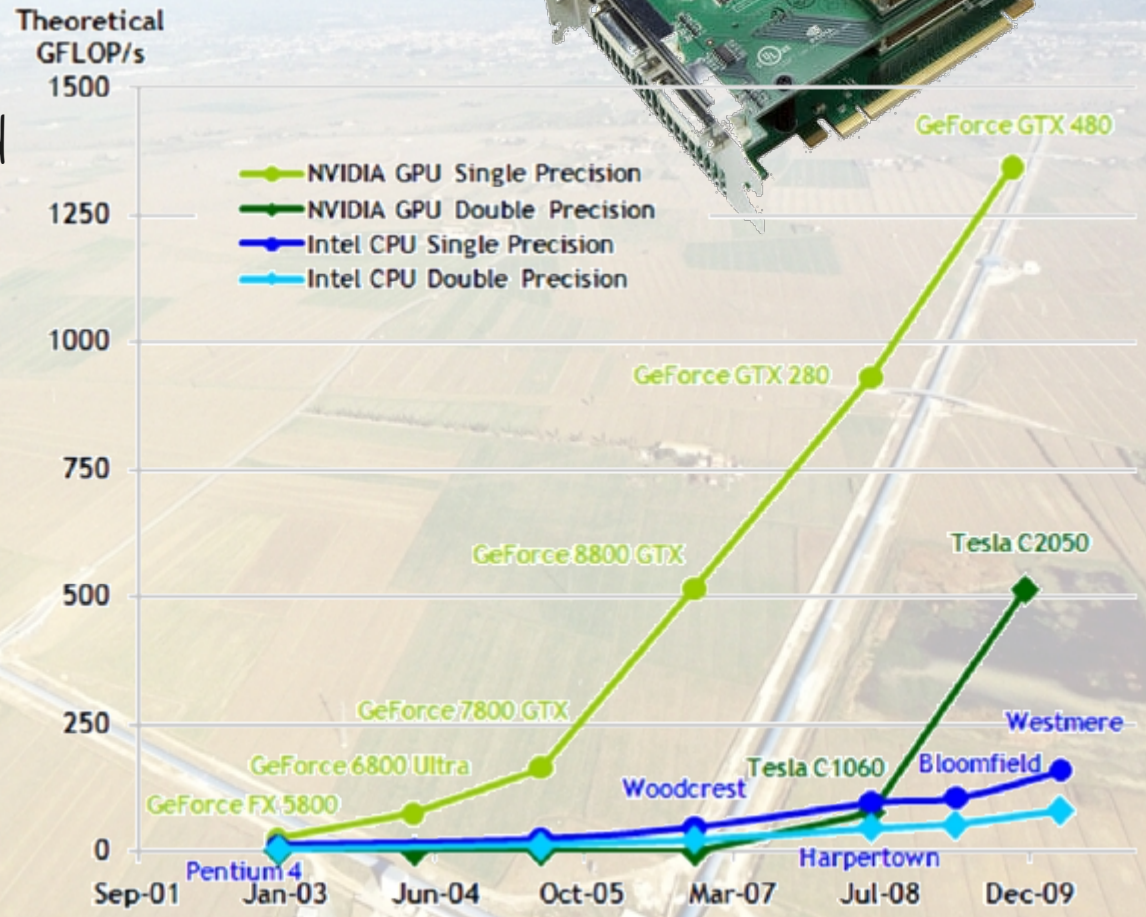
- Mapping of abstract workflows like DAGs/DAXes to the Grid is now easily possible with the Virgo Pilot server.
- Complex and relational workflow handling is missing from Cream/WMS.



Many Core Computing @ Virgo

- Motivation:
- A lot of algorithms used in gravitational wave analysis can be massively parallelized and/or a lot of independent calculations can be performed at the same time.
- **BUT:** Many Core computing developes drastically, protocols, programing languages, standards emerging slowly -> Platform independece can be crucial
- Example: 2 order of magnitude speed-up in a typical CBC pipeline





Future

- Experience showed that GW analyzis computing load significantly peaks at certain times -> it is difficult to handle this with dedicated/isolated clusters. Shared resources (grids/interconnected clusters) could help a lot !

- Relative size of data will shrink -> data transfer will not be a real issue.

- Have to be prepared for cloud computing to reach dynamical/temporal resources.

- Many-core architectures offer operation level paralellizability, definitely something to exploit !

- Make software resource independent

Summary

- Gravitational Wave searches are extensively using a wide range of various computing resources
- Different algorithms have different optimal mapping to computing architectures
- **Grid tools** help in
  - data transfer issues
  - and in running non-relational, relatively simple analysis workflows
- **Dedicated clusters** with shared file system for user jobs and hierarchical, relational workflows
- **Pilot pool** provides a solution for mapping complex workflows to Grid.

- Highly paralellizable algorithms are very well suited to **GPUs**
- Various future **manycore architectures** and programing languages will be of great help for GW-DA !

Thank you for your attention !

VIRGO

Extra slides – Grid: what is missing

- **Data transfer:** FTS is far too heavy for Virgo – size collaborations. Lack of flexibility and support for Postgres/MySQL backend. Steep learning curve.
- **Matchmaking:** A *client side* matchmaking plugin would allow easier pilot approacs and late bind to resources without having to reinvent/rewrite matchmaking for the pilot job dispatcher.
- **POSIX file access** to Grid Storage (Classic SE) and global transparent file system. – (DPMfs or NFS4 interface, etc..)
- Large **scheduling delay** / status update in some circumstances.
- **Poor user interface**, no middleware provided job submission, managment tool.