Toward a common Gravitational Wave detectors distributed cyber infrastructure

Franco Carbognani – EGO

on behalf of the LIGO and Virgo collaborations

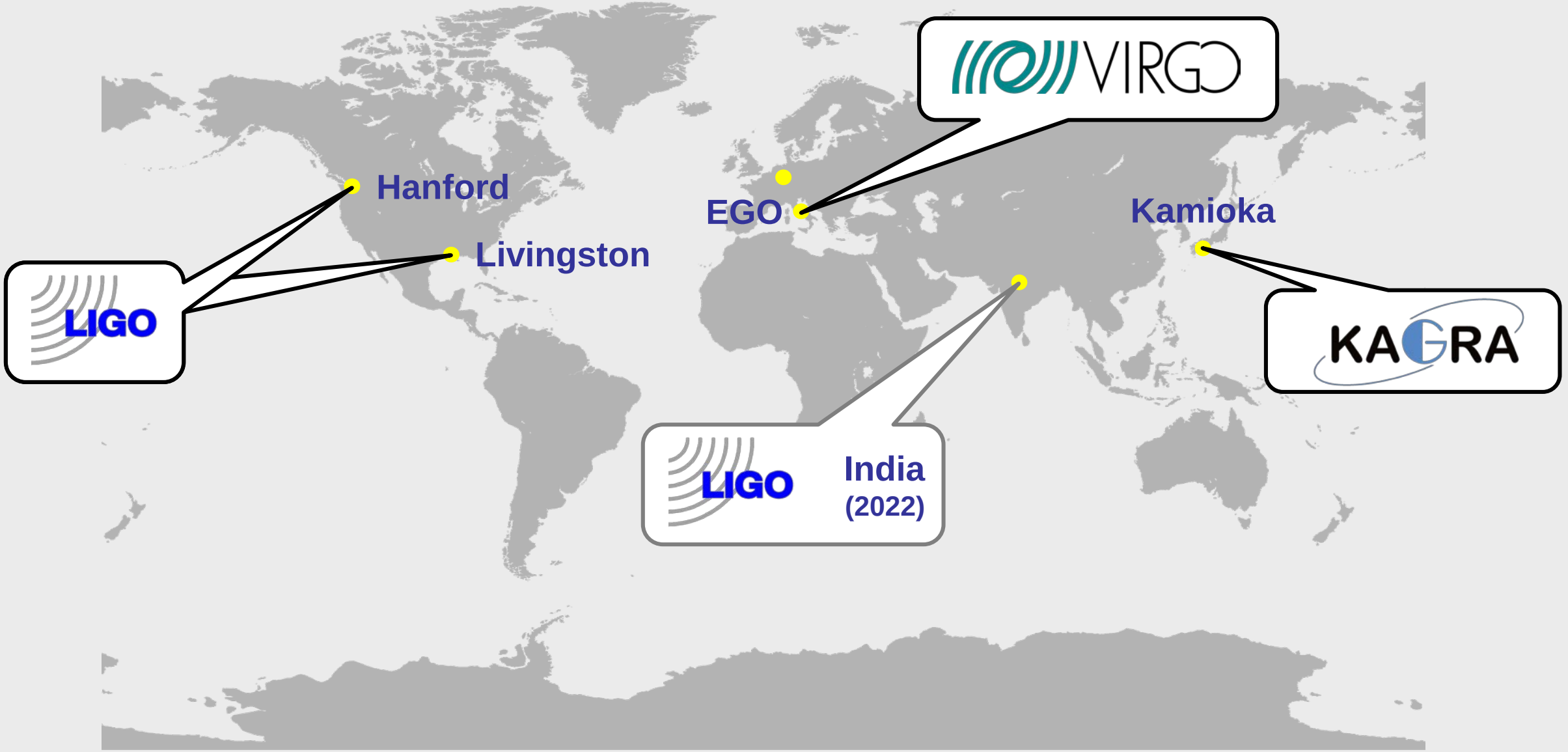
ESA/ESO SCIOPS Workshop | 19-22 Nov 2019, ESAC, Madrid, Spain

Overview

- A Worldwide Network: The International Gravitational Waves observatory Network (IGWN)
- Data and data flow schema
- Timelines and trends
- What we need
- The common solutions and the common tools
- Where are we?
- Conclusions

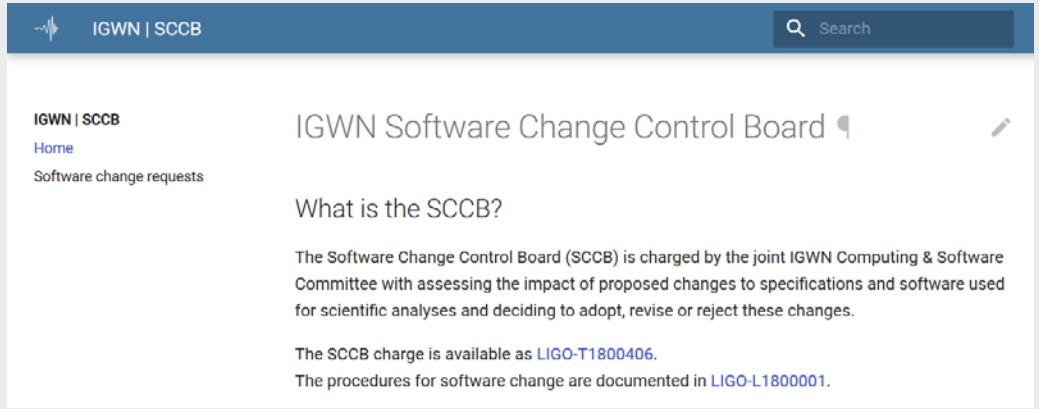
Caveat: this (partial) summary sees things from the Virgo point of view!

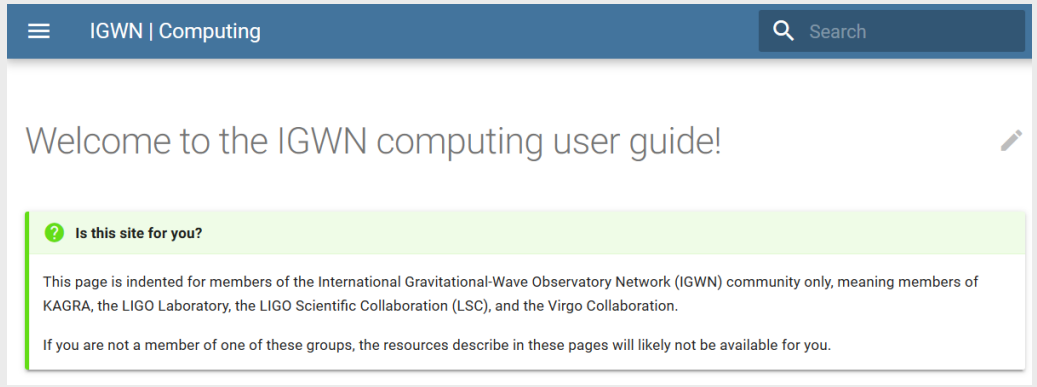
Worldwide Network



- International Gravitational Waves observatory Network (IGWN)
  - A coordination effort aimed at jointly discussing the computing policy, management, and architecture issues of LIGO, Virgo, and KAGRA.
  - *igwn.org* will host common Virgo/LSC services, and KAGRA is joining
  - Migration planning for common services underway (GitLab, Wiki, etc.)
  - Started discussion on sharing common critical services support over the collaborations time zones
- Plans for migration from legacy tools to common, mainstream tools are being finalized
  - Data management and transfer (Rucio & Kafka)
  - Software management (GitLab, Cmake ,Conda, CVMFS)
  - The tools (and the general architecture) are chosen to be consistent with many large scale computing projects such as WLCG

- A large number of common operations and coordination boards
  - Weekly IGWN CompComm is the main coordination forum
    - KAGRA representatives just joined
    - This is now a six-chairs committee (Franco Carbognani, Stefano Bagnasco, Peter Couvares, Duncan McLeod, Ken-ichi Oohara, Nobuyuki Kanda)
  - Weekly/Biweekly LIGO and Virgo Dev/Ops calls (with joint participations)
  - Weekly OSG-IGWN prod calls
  - IGWN Software Change Control Board
- F2F sessions at LVC Collaboration meetings
- Semi-annual F2F IGWN High-Throughput at Georgia Tech
- Other thematic F2F meetings





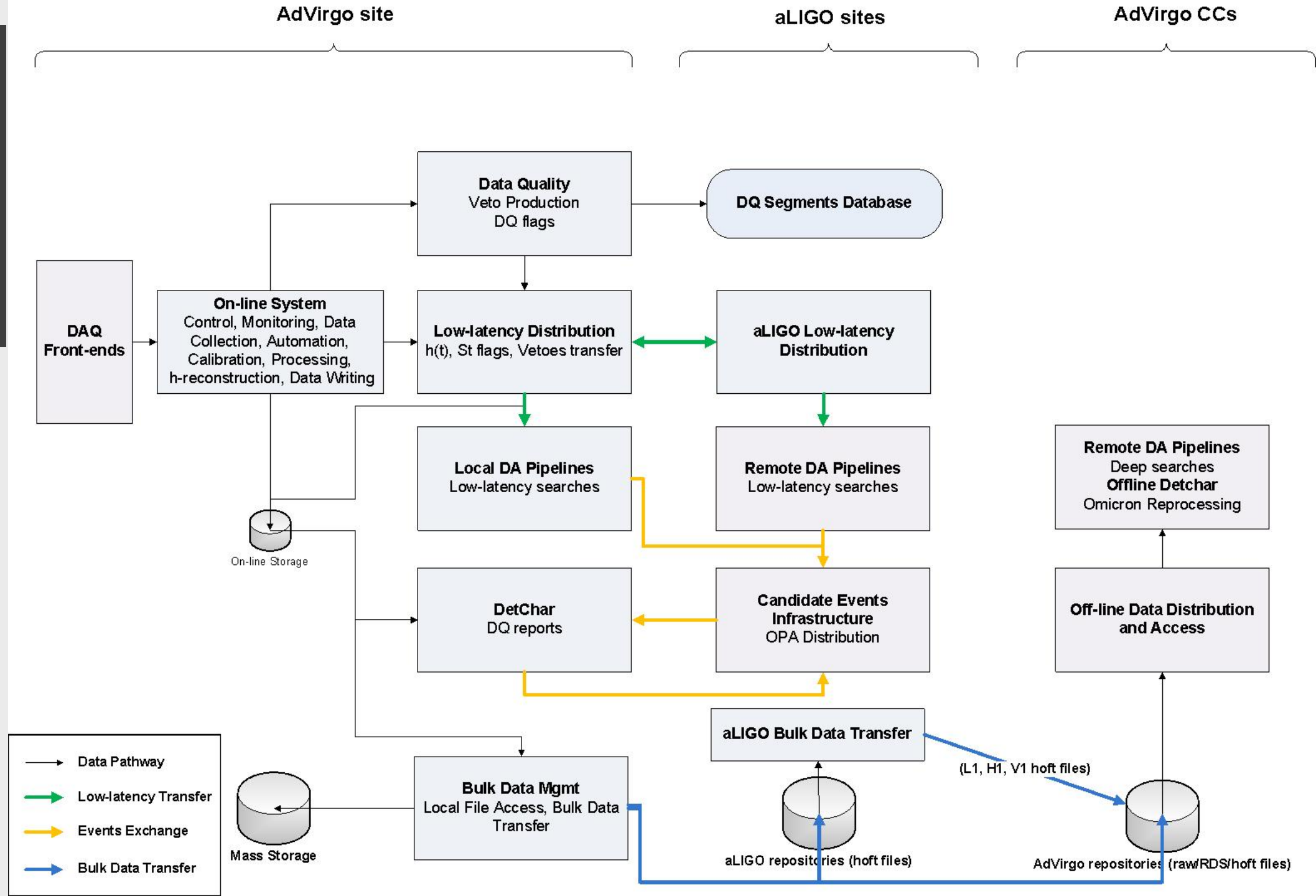
Raw Data, ~1PB/yr:

- **Full Bandwidth Raw**, not exported (7.5-10 TB/day)
- **Raw Data:** downsampled, calibrated and uncalibrated h(t) (3-4 TB/day)
- **A few levels of reduced data sets** for various uses

Data for physics, ~ 5TB/yr/detector:

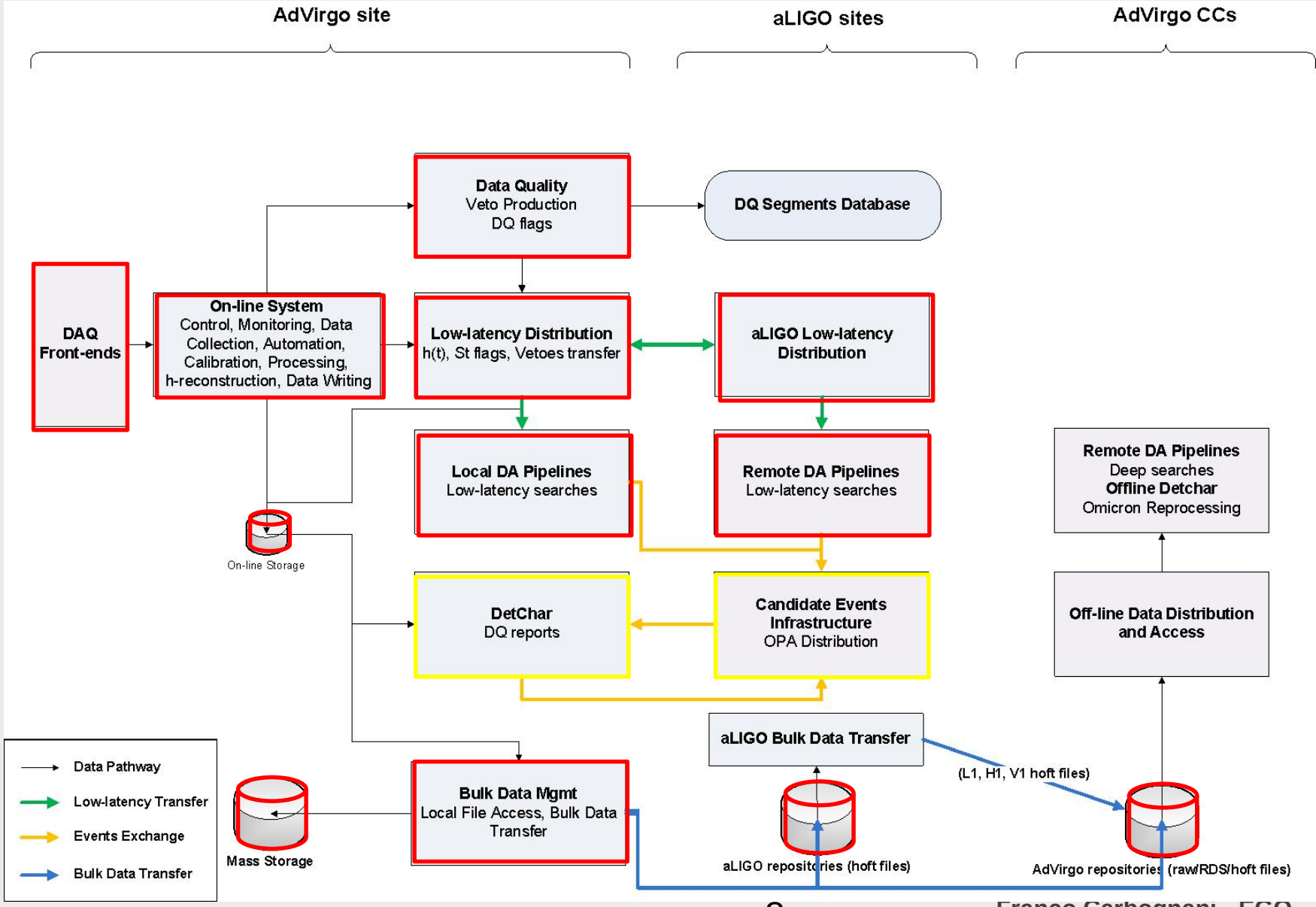
- Virgo h(t): calibrated «strain» data
  - sampled at 10 KHz, as ~1kSec frame files
  - Includes state vector (data quality flags, vetoes,…)
- LIGO h(t)
  - Copied online to EGO for low-latency searches and exported to CCs for offline analysis
- KAGRA h(t)
  - Coming soon…

The Data Flow Schema

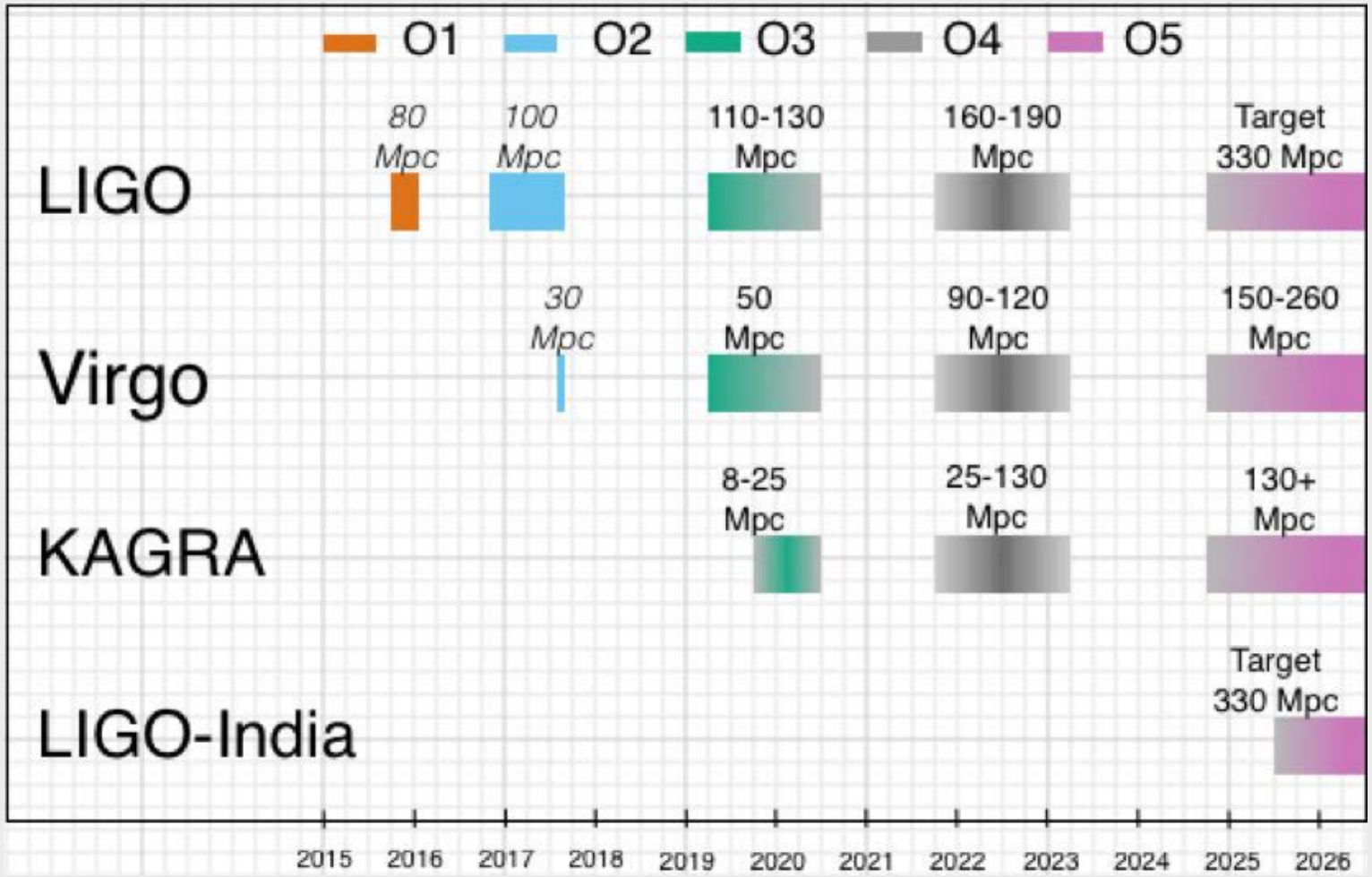


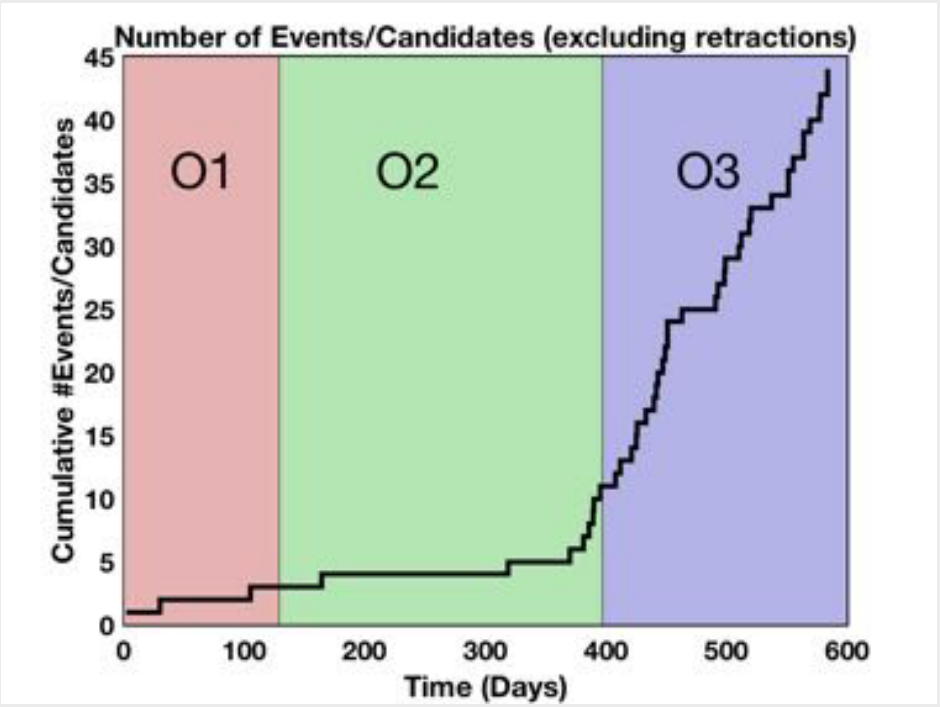
Data Flow: The GW170814 case

1. The signal arrives
2. Data composed into frames
3. Calibration of the data
4. Veto, DQ flags production
5. h(t) transfer
6. Low-latency matched-filter pipelines
7. Upload to GraceDB
8. Data written into on-line storage
9. Low-latency data quality
10. Low-latency sky localization
11. GCN Circular sent out
12. Data written into Cascina Mass Storage
13. Data transfer toward aLIGO and CCs



Timelines and Trends



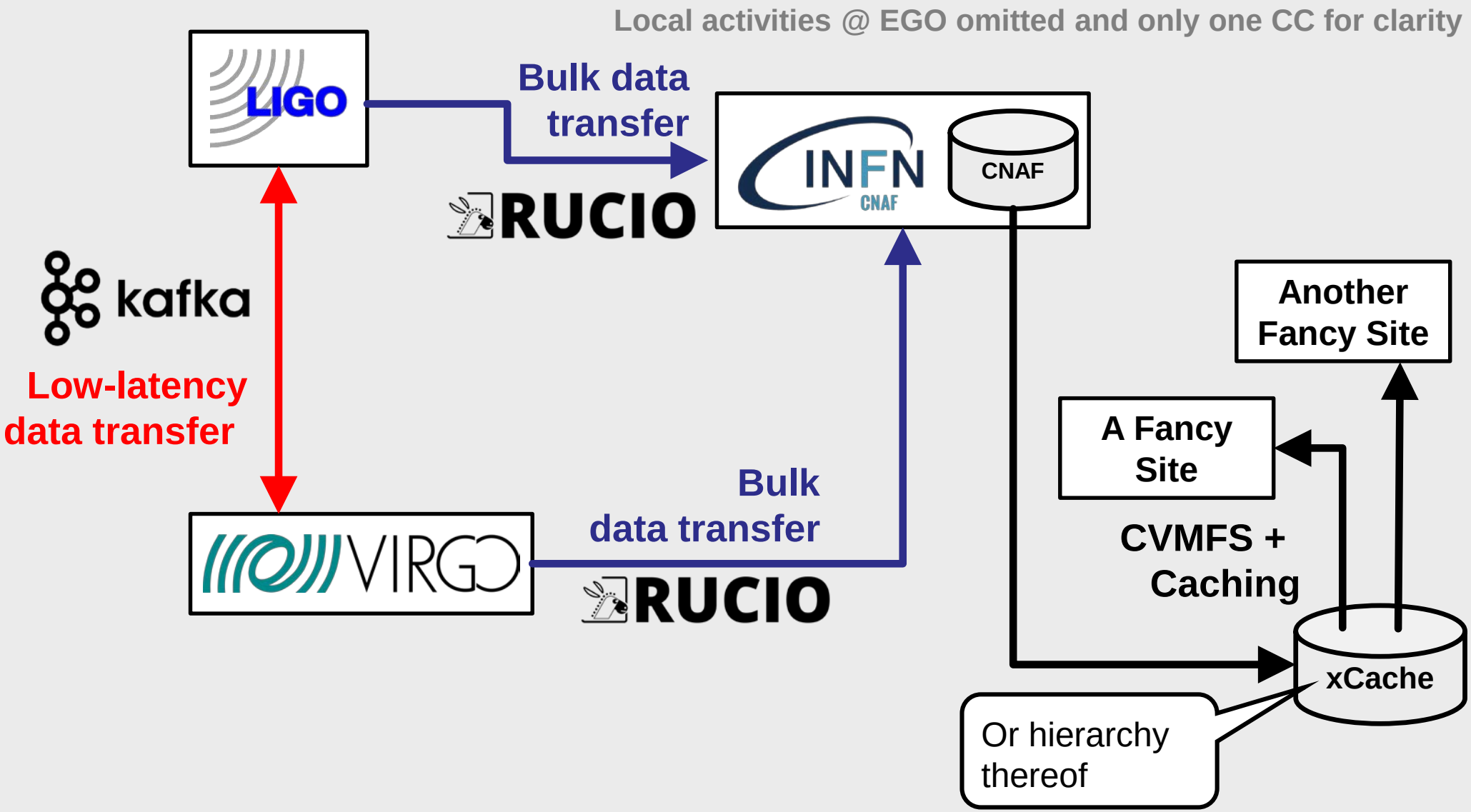


What is needed

The geographical separation of the detectors and the different timescales involved imply the creation of a common distributed cyber infrastructure which must guarantee:

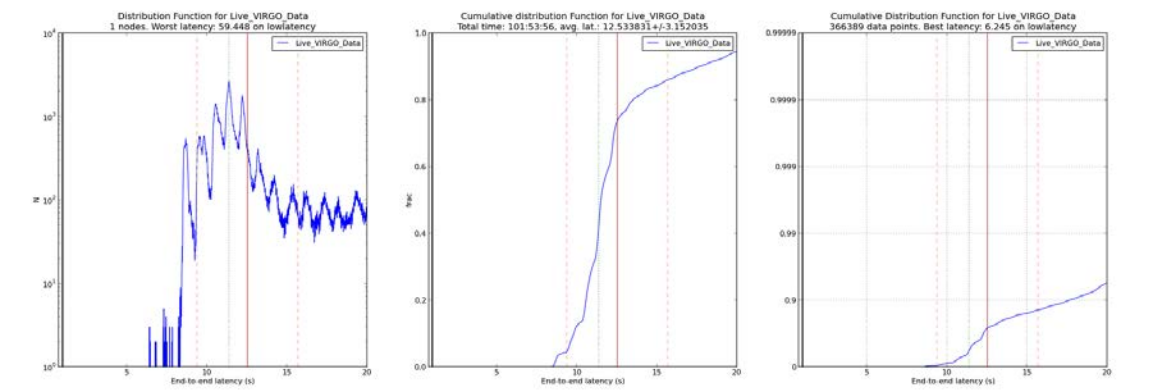
- Adequate storage and computing resources, for commissioning, detector characterization and low-latency searches
- Fast communication among the different observatories and computing clusters for low-latency searches
- Reliable bulk data transfer to custodial storage in CCs
- An ubiquitous and uniform running environment for off-line deep searches on dedicated resources and heterogeneous infrastructures
- An homogeneous model for data distribution, bookkeeping and access

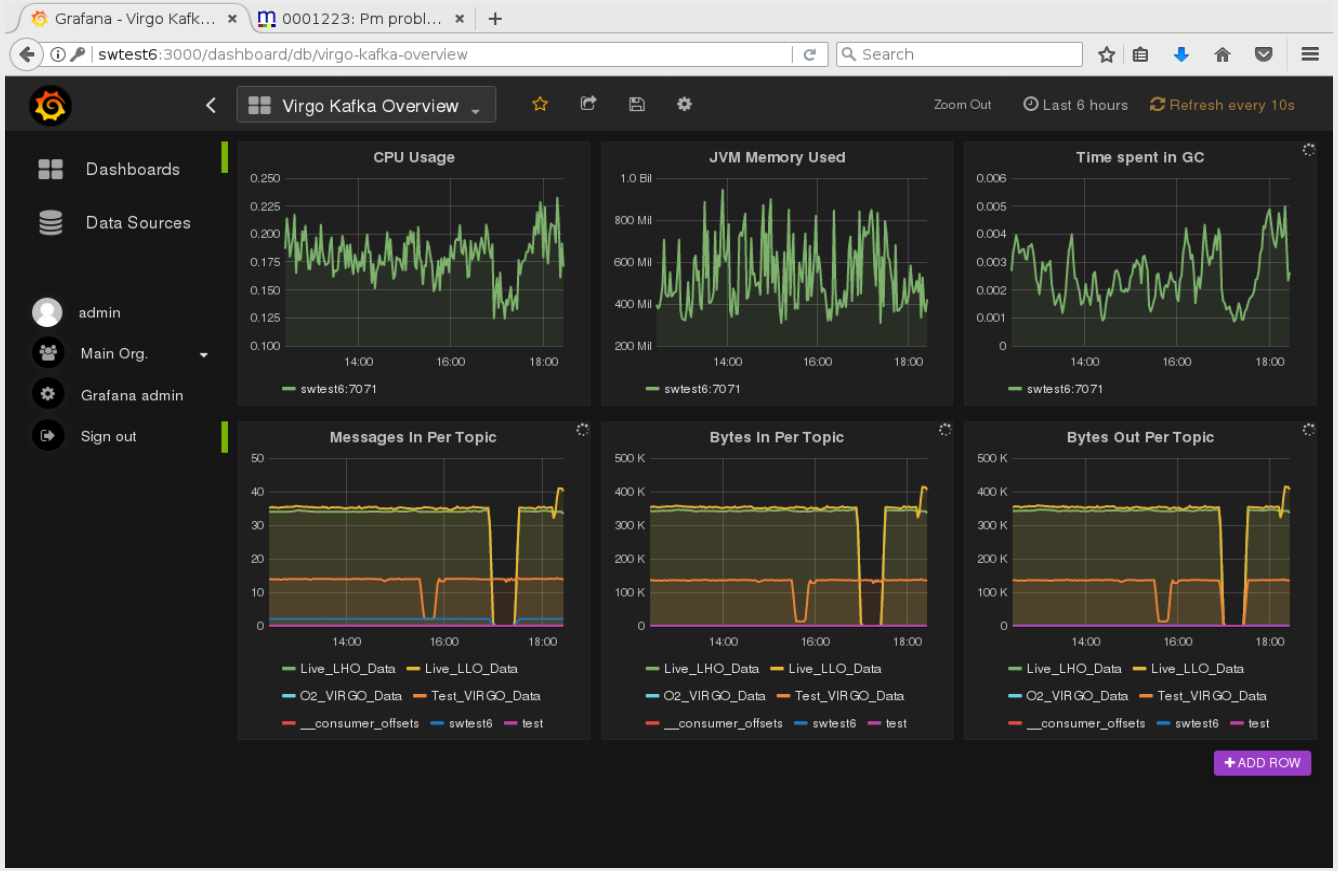
Common Solutions: Data



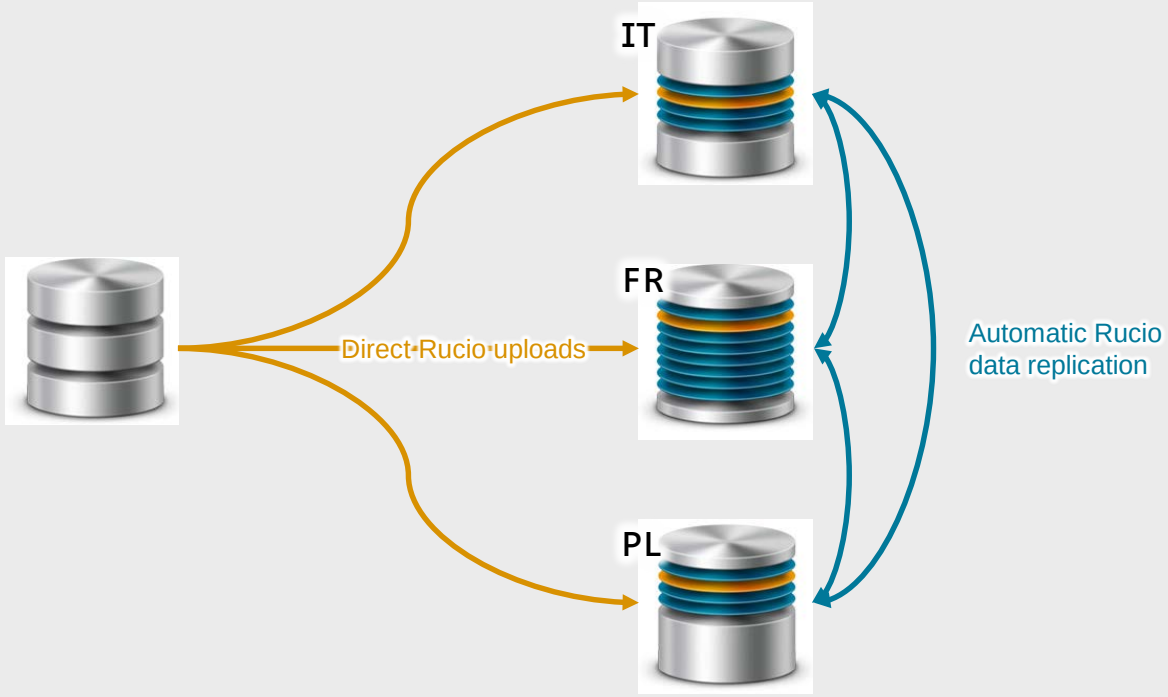
Fast Communication: Kafka

- Kafka is a modern message-query which embed a smart fail-over mechanism and is being used by aLIGO for all other Low Latency links.
- Parallel live data Kafka streams being compared with the ones based on Virgo legacy solution
- Plans to deploy in production for O4 (link to CIT then direct to LLO and LHO)





Bulk Data Transfer: Rucio



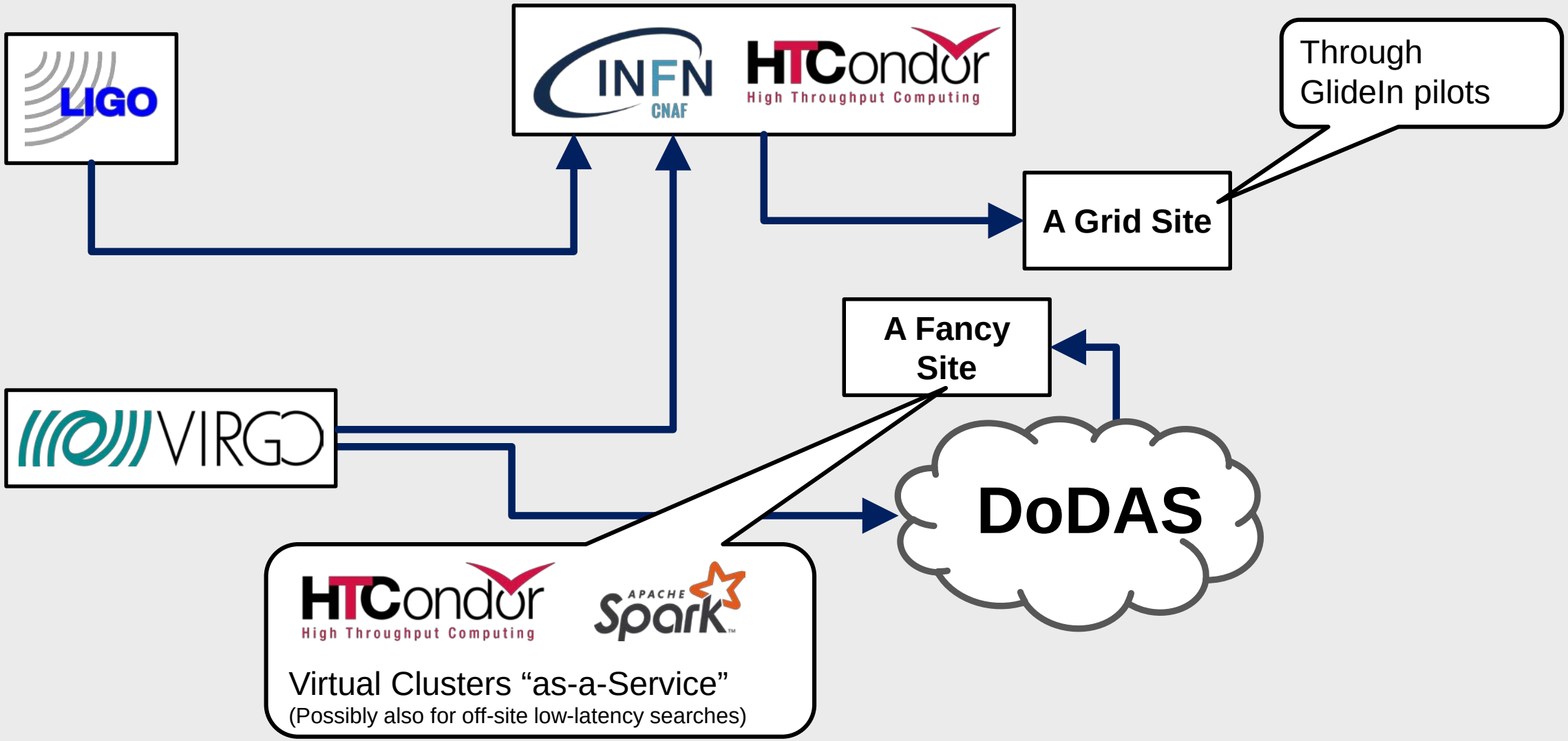

```
[root@rucio-client fuse-posix]# tree /ruciofs

/ruciofs
|-- rucio-server-1
|   |-- tests
|   |-- user.gfronze
|   `-- user.root
|       |-- subfolder
|       |   `-- test-file-1.txt
|       `-- test-file-2.txt
`-- rucio-server-2
    |-- production
    |-- simulation
    `-- user.gfronze
        |-- analysis-output
        |   `-- output-31102019.txt
        `-- test-file-3.txt
```

- Direct transfers and third party copies towards CNAF and CCIN2P3 storage endpoints work thank to a Virgo-made improvement of Rucio.
- CCIN2P3 storage endpoint is equipped with a compatibility layer which revealed to be totally transparent to Rucio.
- Test of a Rucio-handled EGO circular buffer expected for the next months.
- Official Rucio POSIX representation under development by Virgo-Torino group: access Rucio catalog as a standard filesystem.
- Open points before production:
  - Deploy the infrastructure in a load balancing and fault resilient environment
  - Perform further tests to check required features

Common Solutions: Jobs

Local activities @ EGO omitted and only one CC for clarity



Use HTCondor as a uniform job submission layer

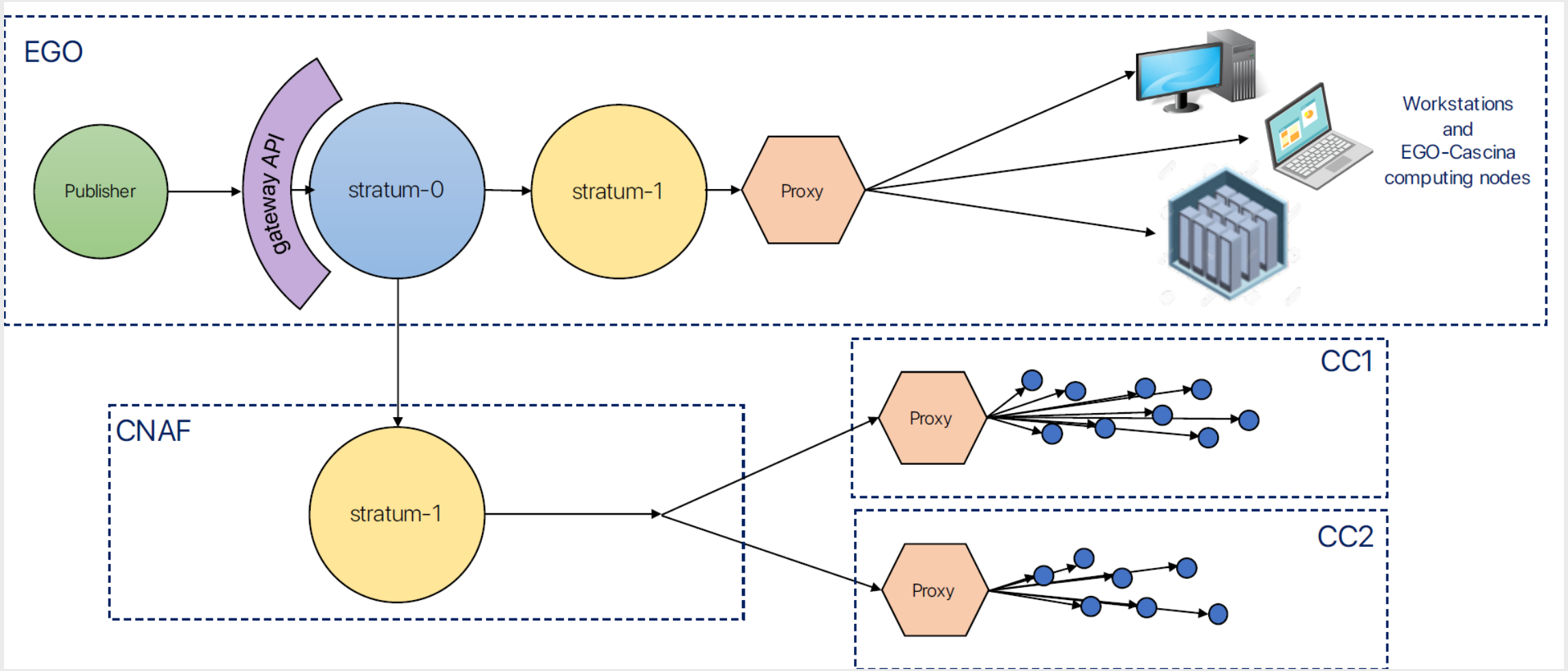
- To local HTCondor clusters where possible
  - Some pipelines already rely on HTCondor DAGs for dependencies
- To common Condor Pools
  - Exploit Grid resources through GlideIn pilots
- To on-demand HTCondor Virtual Clusters
  - Through DoDAS Distributed on-Demand Analysis Services
  - Exploit heterogeneous or opportunistic resources
  - Cater to special needs like off-site low-latency searches

Common Tools: Software Management

- Transition from Virgo Cascina SVN to IGWN GitLab common software archive
  - Mirrored relevant packages from Virgo SVN to git.ligo.org
  - Will migrate to IGWN.org pending administrative and governance decisions
  - Training sessions successfully organized
  - A IGWN Software Archive Organization Doc soon to be released
- Transition from CMT to CMake for software build
- Transition to Conda for software environment definition and software packaging
  - Several modules have been packaged successfully in Conda, work steadily underway
  - Next step adaptation of existing higher-level tools to work in the new environment

Common Tools: Software Management

- Full deployment of CVMFS for software distribution from Virgo Cascina done
  - Just started software publication



Data Analysis Tools and Pipelines

- Next, need to interact with pipelines to help them exploit the new possibilities
  - Several activities already started, will need to regroup and start supporting them: this will happen in the coming weeks
  - The final goal: run any pipeline “on the infrastructure” using as many resources as compatible, minimizing the need for the developers and users to know the details of it.

Where are we?

Low Latency data distribution

Kafka

Bulk data transfer

Rucio

Software packaging and distribution

CVMFS (+ Conda + Singularity)

Data distribution

CVMFS + StashCache

Workload management

HTCondor and, possibly, DoDAS

Data cataloguing and bookkeeping

Probably Rucio again

Mostly decided except implementation details, under deployment or active development

Conclusions

- IGWN means a coordinated and common infrastructure between the collaborations
- Making a large technological step on many tools
  - To provide interoperability between Virgo, LIGO (and KAGRA) and enable effective convergence towards IGWN
  - To reduce maintenance effort on proprietary/legacy solutions
  - Tests on Rucio and Kafka satisfactory, working on deployment strategy
  - Migration to modern software management tools well underway
- Still much planning work at this stage
  - But also testing and deployment going on
  - Most of the planned architecture should be ready on the O4 timescale
- Designing a common robust and scalable long term architecture