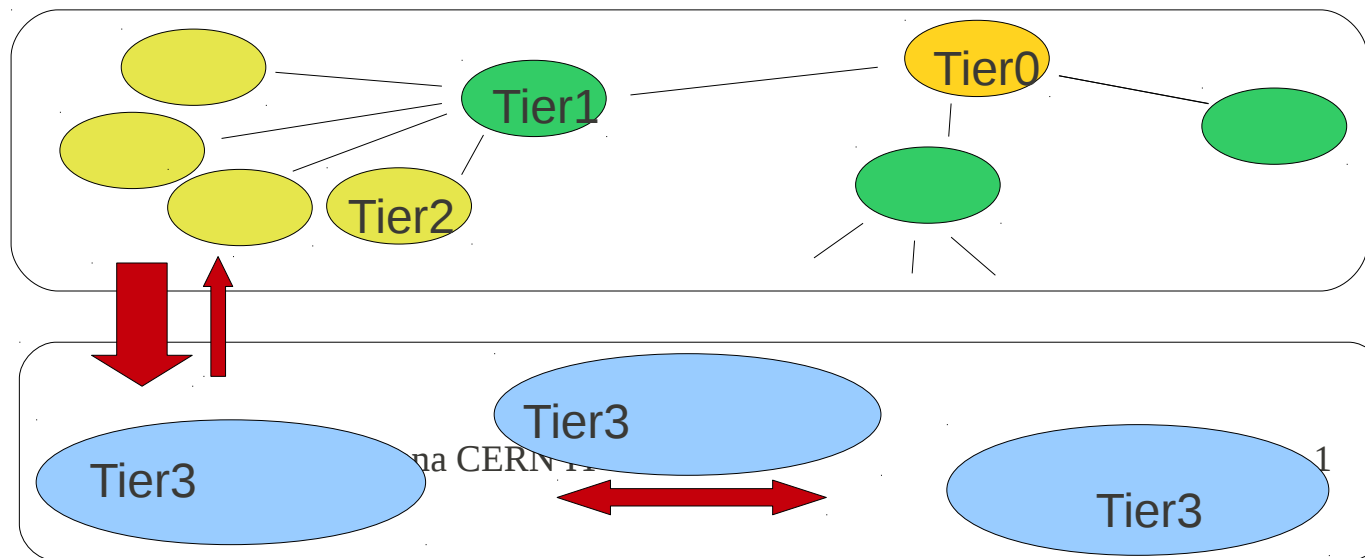


# ATLAS Tier3 recap

- Collaboration-wide effort to have a sustainable expansion of computing resources
  - Focus on analysis
  - Focus on simplicity
- One size fits all?
  - Different (but coherent) models across a wide range of facility
    - \_ From national analysis facilities, to expansion of existing Tier3 to university facilities
- What is really important
  - Go for client-based solution reducing the number of services
  - Extensive use of caches (starting with “other” data, e.g. condition db files)
  - Relatively isolated new layer (T0/T1s/T2s works and must not be affected by the Tier3s prototyping and operations!)
    - \_ Laboratory for ATLAS Distr. Computing evolution

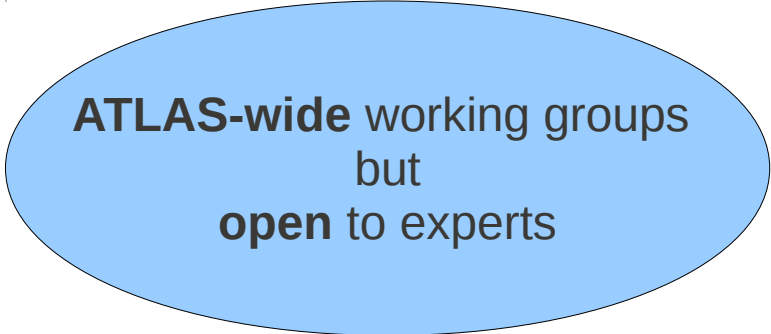


Data storage technologies:  
xrootd, distr, file systems,  
dpm (no srm), ...

# ATLAS Tier3 wgs

- 6 working groups (delivering final reports now)
  - **DDM-Tier3 link**
    - Simone Campana (CERN)
  - **Distributed storage (Lustre/Xrootd/GPFS)**
    - Rob Gardner (Chicago) and Santiago Gonzalez de la Hoz (Valencia)
  - **Software / Conditions data Working Group**
    - Alessandro de Salvo (INFN Roma) and Asoka da Silva (TRIUMF)
  - **PROOF Working Group**
    - Wolfgang Ehrenfeld (DESY) and Neng Xu (Wiscosin)
  - **Tier 3 Support**
    - Dan van der Ster (CERN)
  - **Virtualization working group**
    - Yushu Yao (LBL)

Massimo Lamanna CERN IT



**ATLAS-wide** working groups  
but  
**open** to experts

# Links with this workshop?

- **ATLAS Tier3 looks not too distant from the “p2p” system**

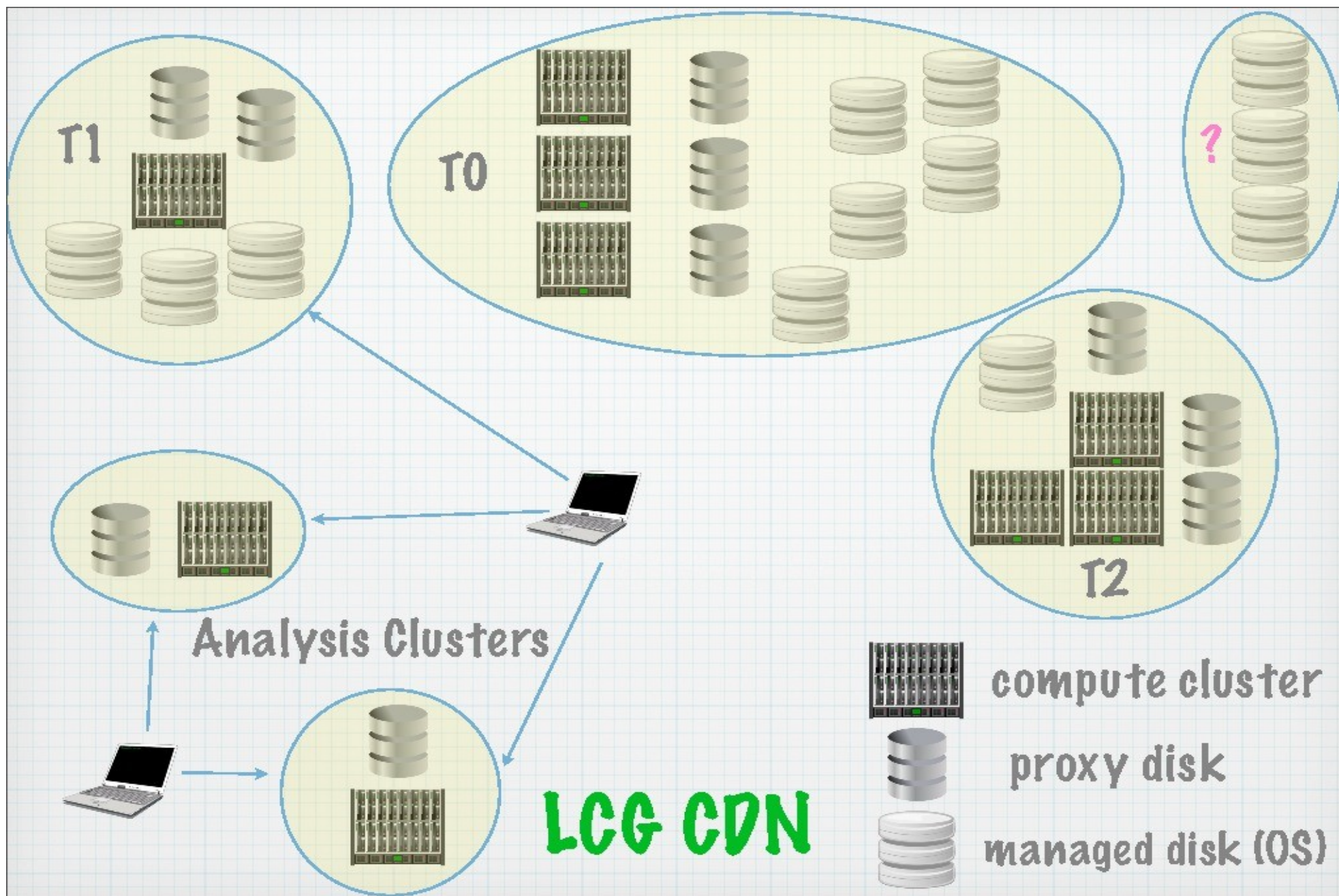
- Minimal system (tier3g)

- Software: CVMFS injecting
    - Condition data: via SQUID (from Oracle DBs)
    - Condition data files: CVMFS
    - Events: xrootd without catalogue (neither local nor central)

ATLAS T3 ongoing discussion  
resonating with Jeff's contribution

- **P2P T3?**

- Event storage acting entirely as a cache (in the T3 POW but low priority so far)
  - As all other T3 caches extracting and reinjecting data from “grid and services” (existing T0/T1s/T2s)
    - Preload data via synchronous download (dq2get a bit like wget) or asynchronous (dq2get via FTS)
    - Jobs assume data are present. Missing file triggers files recall (xrootd via global redirector)



# Demonstrator

- **Building blocks:**
  - Run gangaRobot (hammerCloud) over a “long” period of time.
    - One can actively make the cache incomplete for example by removing a fixed fraction of data (leaving some data sets with “hole” and remove some datasets entirely)
    - Let users activity to go on in parallel
  - Compare with/without the “cleaning” script running
- **Many open points**
  - T0/T1s/T2s is the “**origin server**”
  - Decide the size of the proxy network ( $\# \text{tier3} - 1$ )
  - Can we run it on a hybrid network (xroot + lustre sites)?
    - Difficult in the present system (but cfr. Brian's talk on Hadoop)
- **Time scale:**
  - To be discussed within the Tier3 activity and ADC