

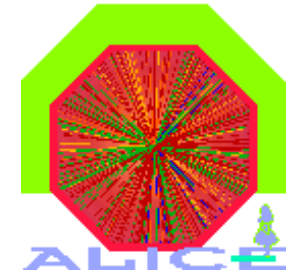
# The CERN Analysis Facility

## A PROOF Cluster for Day-One Physics Analysis

J. F. Grosse-Oetringhaus,  
CERN/ALICE

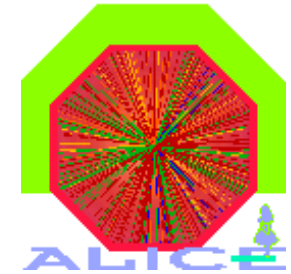
CHEP 2007

# Content



- Quick introduction to PROOF
- The ALICE experiment
- The CERN Analysis Facility (CAF)
  - Concept and Structure
  - Test setup
- Evaluation & Development
  - Datasets, data distribution & flow, staging from AliEn, CASTOR
  - Monitoring
  - Groups, disk quotas, CPU fairshare
  - Integration with ALICE's software framework AliRoot

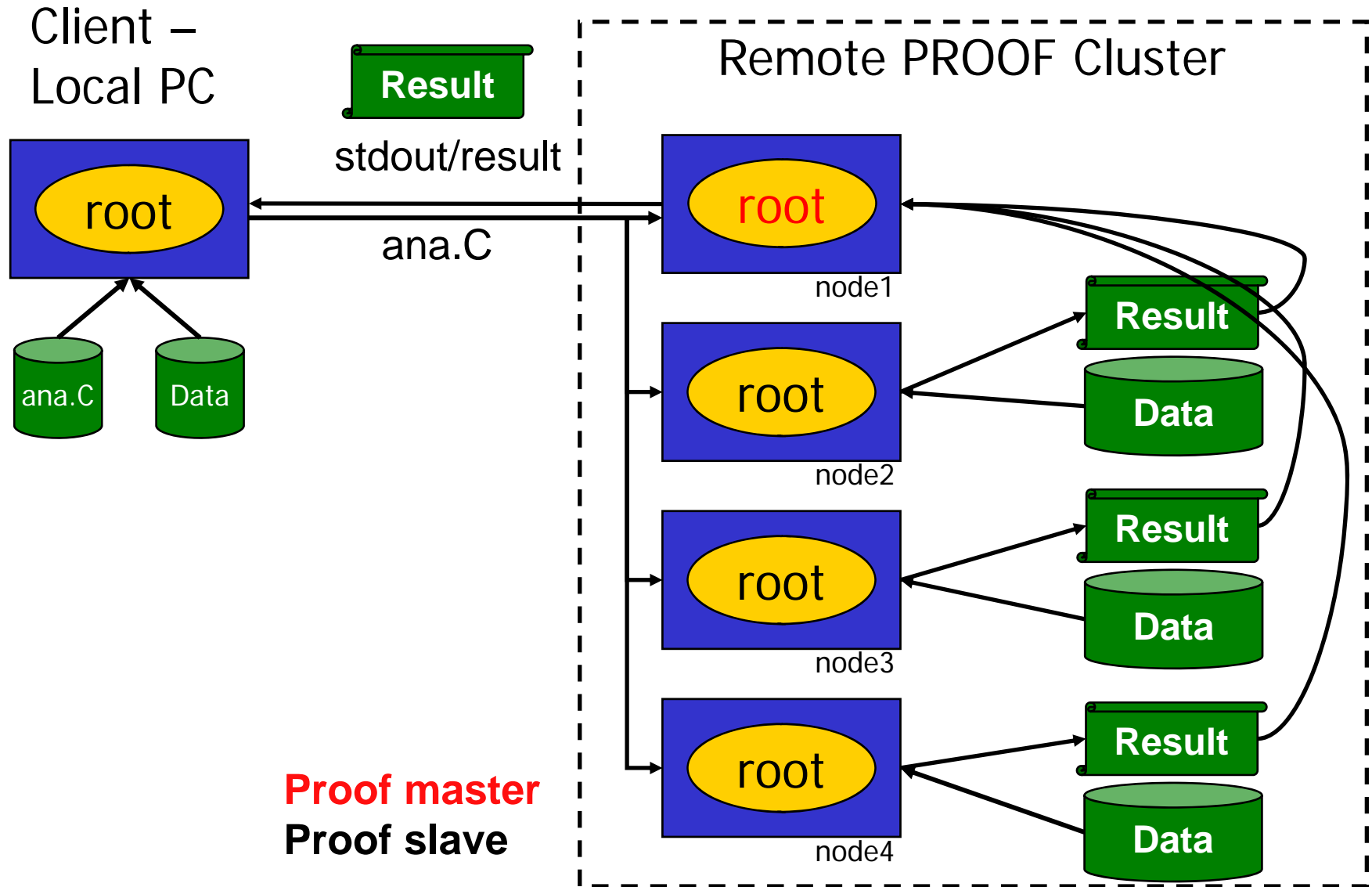
# PROOF



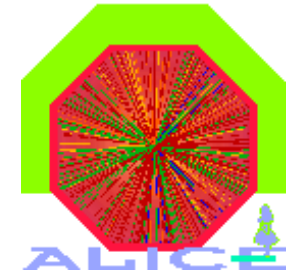
- Parallel ROOT Facility
- Interactive parallel analysis on a local cluster
  - Parallel processing of (local) data
  - Output handling with direct visualization
  - **Not** a batch system
- The usage of PROOF is transparent
  - The same code can be run locally and in a PROOF system (certain rules have to be followed)
- PROOF is part of ROOT

More details:  
F. Rademakers  
[307] Th, 15:20

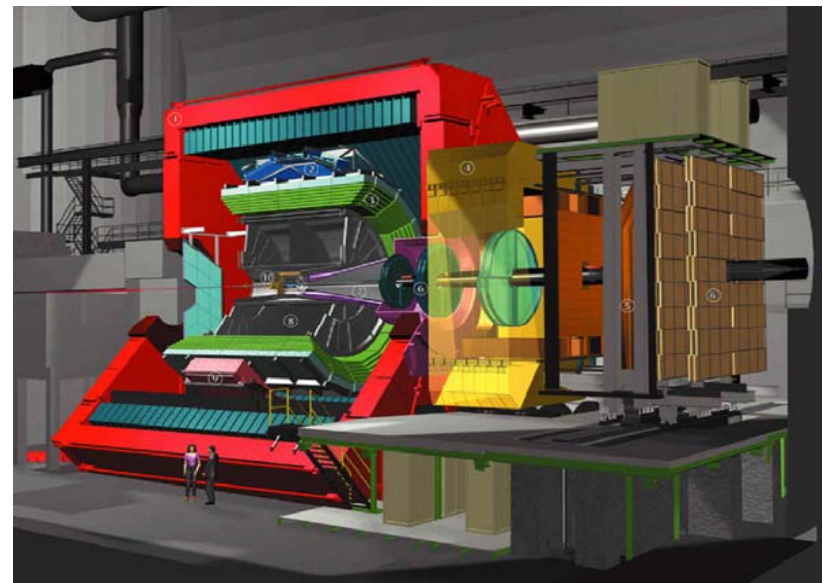
# PROOF Schema



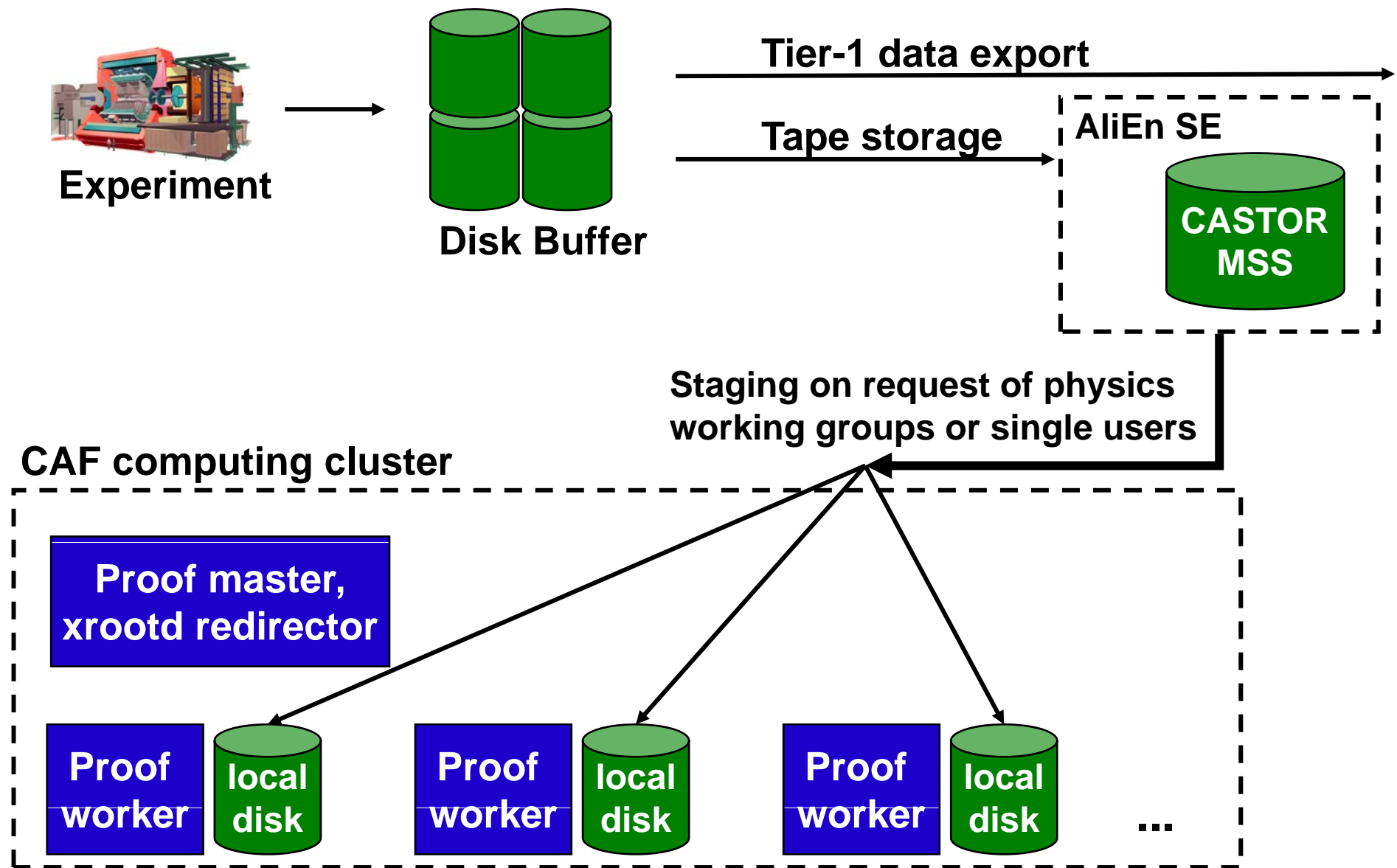
# PROOF for ALICE



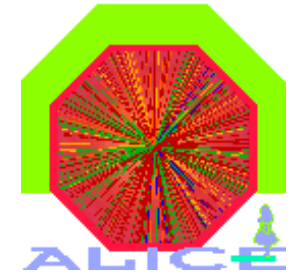
- ALICE (A Large Ion Collider Experiment) will study strongly interacting matter and the transition to the quark-gluon plasma
  - The Grid is intensively used for simulation (now), reconstruction, storage and analysis (from 2008)
  - A cluster called CERN Analysis Facility (CAF) running PROOF will allow
    - Prompt analysis of pp data
    - Pilot analysis of PbPb data
    - Fast simulation and reconstruction
    - Calibration & Alignment
    - Design goal: 500 CPUs, 100 TB of selected data locally available
- Focus: Fast response time



# CAF Schema

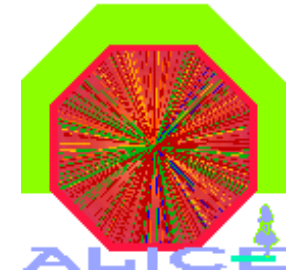


# CAF – Test Setup



- Test setup in place since May 2006
  - 40 “standard” machines, 2 CPUs each, 250 GB disk
  - 2 Partitions: development (5 machines), production (35 machines)
- The cluster is a xrootd pool
  - 1 Machine: PROOF master and xrootd redirector
    - Several possible
  - Other machines: PROOF workers and xrootd disk servers
    - Access to local disks → Advantage of processing local data
- Testing led to a considerable amount of bug reports (most fixed 😊) and feature requests

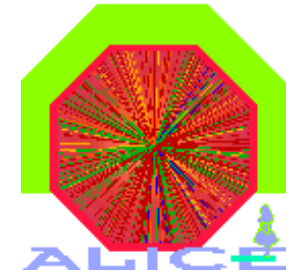
# Data distribution



- Currently: semi-automatic
  - Staging of each single file has to be requested (can be done by the user)
  - xrootd + staging script used to select the staging disk server and actually stage the file
  - Available files communicated by text files containing list of files
  - 30K files distributed (MC) = 3M pp events, total size ~ 1.5 TB
- Under development by ALICE + PROOF team: Dataset concept (see next slides)



# PROOF Dataset Features

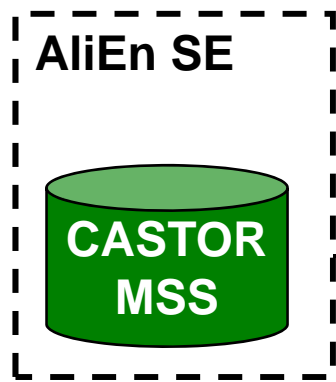
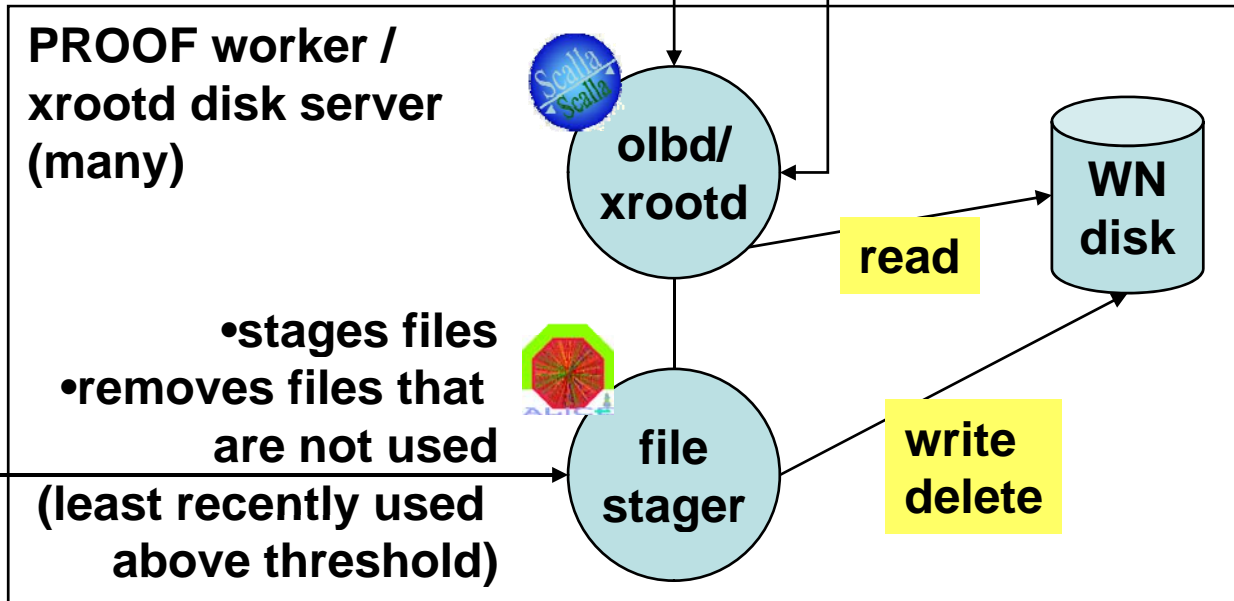
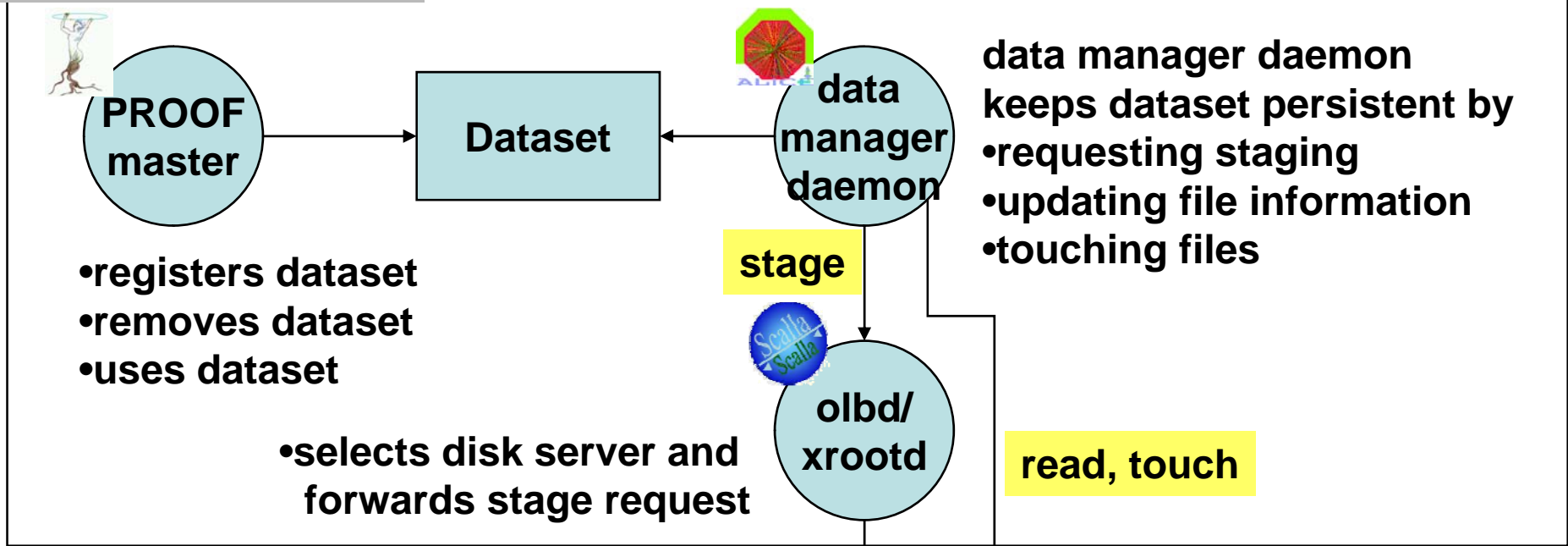


- A dataset represents a list of files (e.g. physics run X)
  - Correspondence between AliEn dataset and PROOF dataset
- Users register datasets
  - The files contained in a dataset are automatically staged from AliEn/CASTOR (and kept available)
  - Datasets are used for processing with PROOF
    - Contain all relevant information to start processing (location of files, abstract description of content of files)
- File-level storing by underlying xrootd infrastructure
- Files available in several datasets to be staged only once
- Quota-enabled
- Datasets are public for reading
- Global datasets

**More details on AliEn:  
P. Saiz [443], Mo, 15:20**

# Dataset concept

## PROOF Master / xrootd redirector

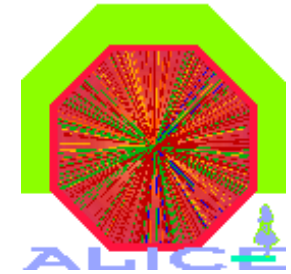


...

# Dataset in Practice

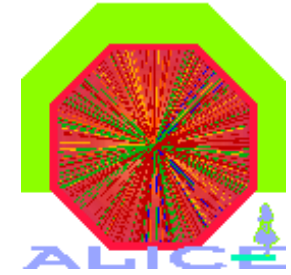
- Create DS from AliEn collection
  - `ds = TGridResult::GetFileCollection()`
- Upload to PROOF cluster
  - `gProof->UploadDataSet("myDS", ds)`
- Check status: `gProof->ShowDataSet("myDS")`
  - TFileCollection myDS contains: 1000 files with a size of 53758817272 bytes, 100.0 % staged  
The files contain the following trees:  
Tree /esdTree: 100000 events  
Tree /HLTesdTree: 100000 events
- Use it: `gProof->Process("myDS", "mySelector.cxx")`

# Groups & Quotas



- Users are grouped
  - E.g. sub detectors or physics working groups
  - Users can be in several groups
- Quotas on disk usage and targets for CPU fair-share (both under development) are enforced on group level
- How to introduce and obtain feedback about the system from ALICE's users?
  - Monthly tutorials at CERN (180 users followed so far)
  - Data challenge data **only** available via PROOF and AliEn Grid

# Monitoring with MonALISA

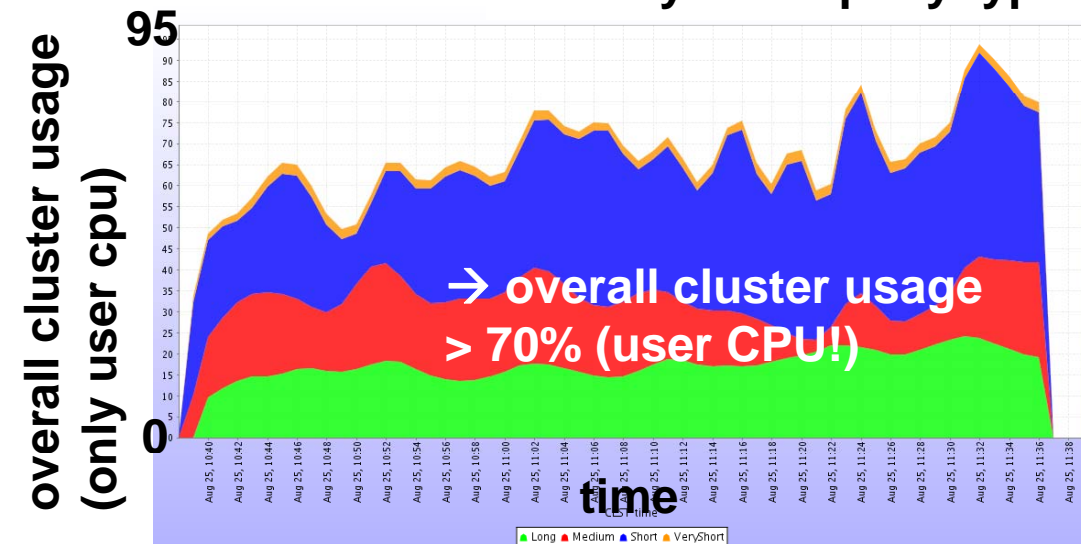
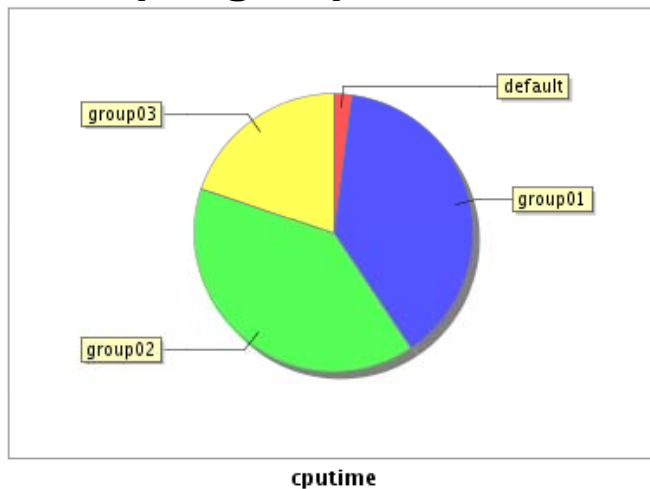


- Cluster (machine-level) with ApMon
- Query statistics
  - Sent at end of each query
- CPU quotas: Consolidation done by ML
- Disk quotas: Visualized by ML

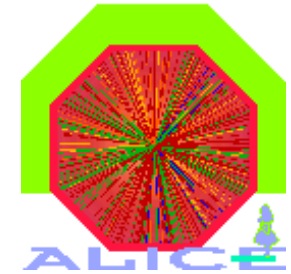
Watch live:  
[pcalimonitor.cern.ch](http://pcalimonitor.cern.ch)  
“CAF monitoring”

Aggregation plot of CPU used by each query type

CPU per group



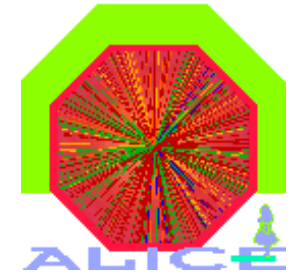
# ALICE's software framework



- AliRoot is based on ROOT: ROOT + set of libraries = AliRoot
- AliRoot on shared space (AFS), libraries loaded after connecting to the PROOF cluster (few lines macro)
  - "converts" ROOT running on PROOF worker into AliRoot
- Case 1: End-user analysis, Comparison to MC
  - Only a few small libraries are needed (ESD, AOD)
- Case 2: Detector debugging, calibration, alignment
  - Full framework needed (access to raw data, clusters, ...)
- Optional user (PWG) libraries
  - Distributed as PROOF packages (par files)
- Wrapper classes (inherit TSelector) enabling access to ALICE's data structures available for both cases
- Integrated with ALICE's analysis framework (based on TTask)

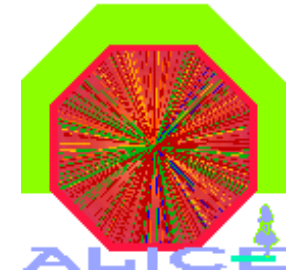
**Details on AliRoot:  
F. Carminati [446]  
Mo, 14:00**

# Outlook



- Finalize datasets & CPU fairshare system
  - Benchmark staging
- Increase cluster size
- Exercise data processing during the full dress rehearsal & cosmic data taking

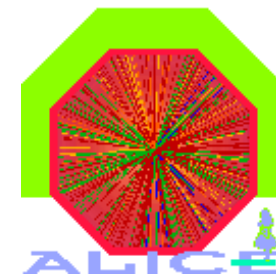
# Summary



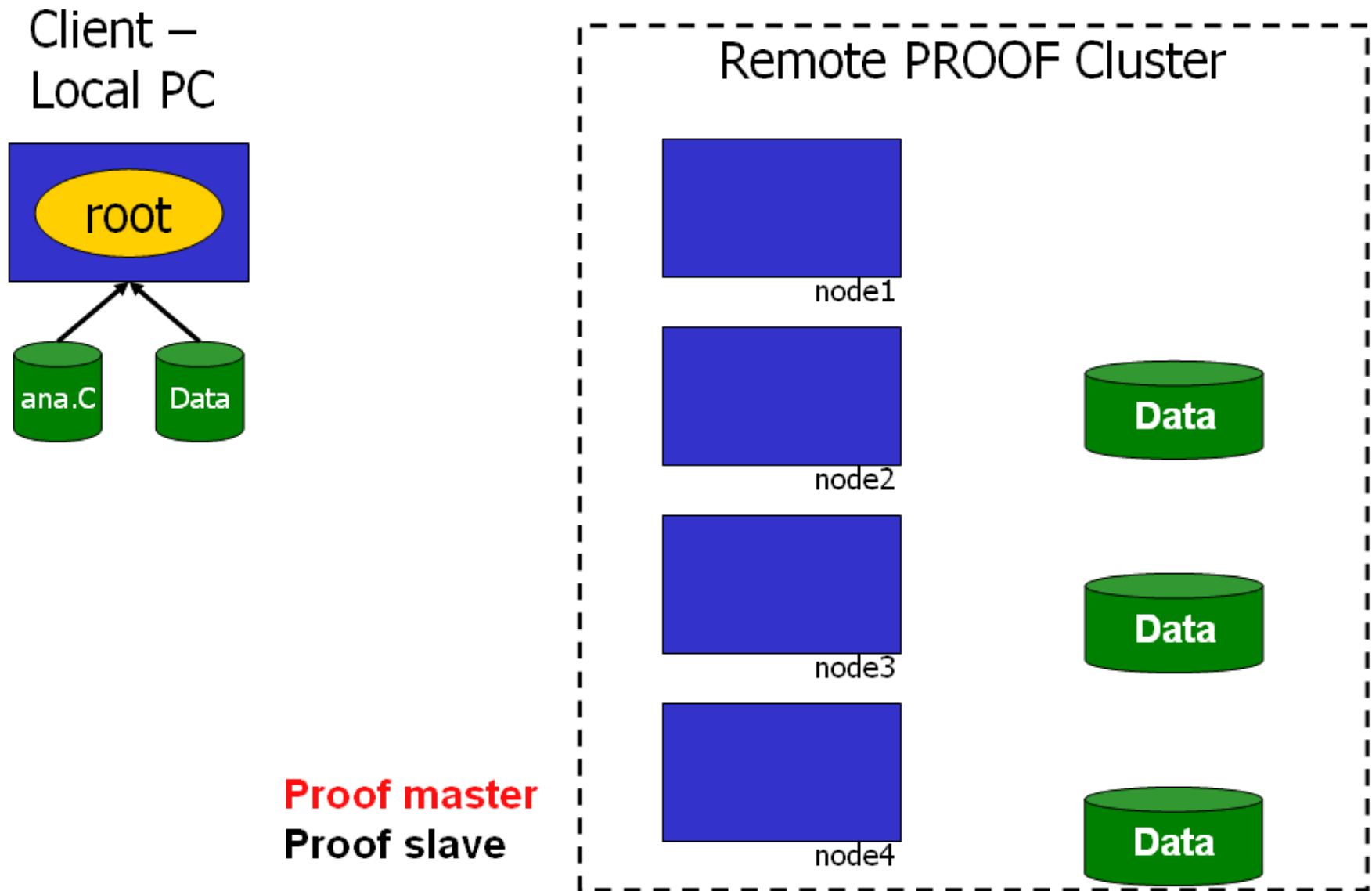
- ALICE uses PROOF on a cluster called CERN Analysis Facility. It will allow prompt and pilot analysis, calibration/alignment, fast simulation and reconstruction  
→ Fast response time
  - ALICE computing model only foresees a CAF at CERN, maybe also good solution for other centers?!
- A test setup is in place since May 2006
  - Users are trained in tutorials and use the system for "analysis"
- Active collaboration with ROOT team
  - Contribution from ALICE to PROOF development
  - Dataset concept and CPU quotas implemented and under test
- ALICE's data processing will benefit and benefits already now from such a system



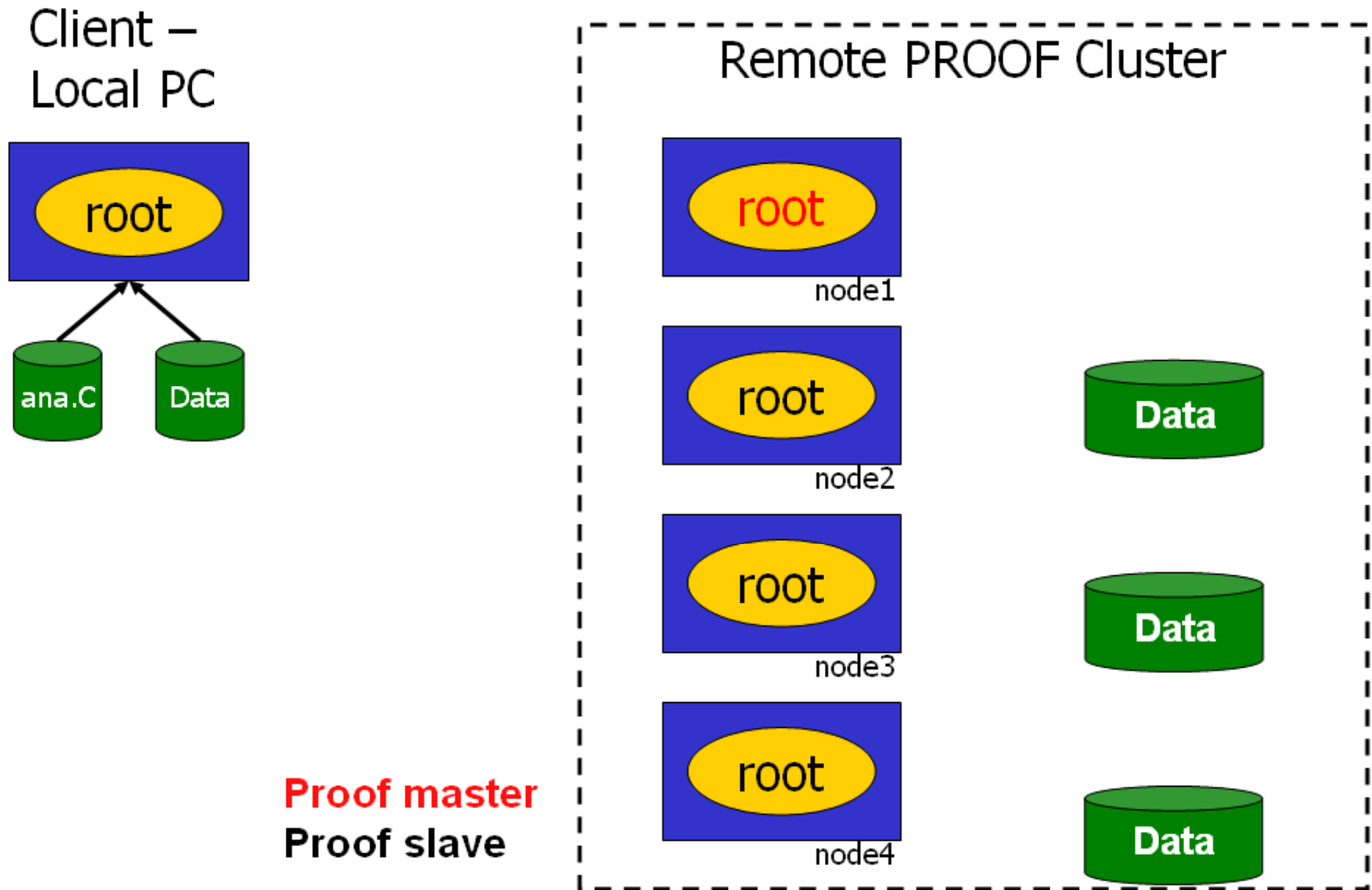
# Backup



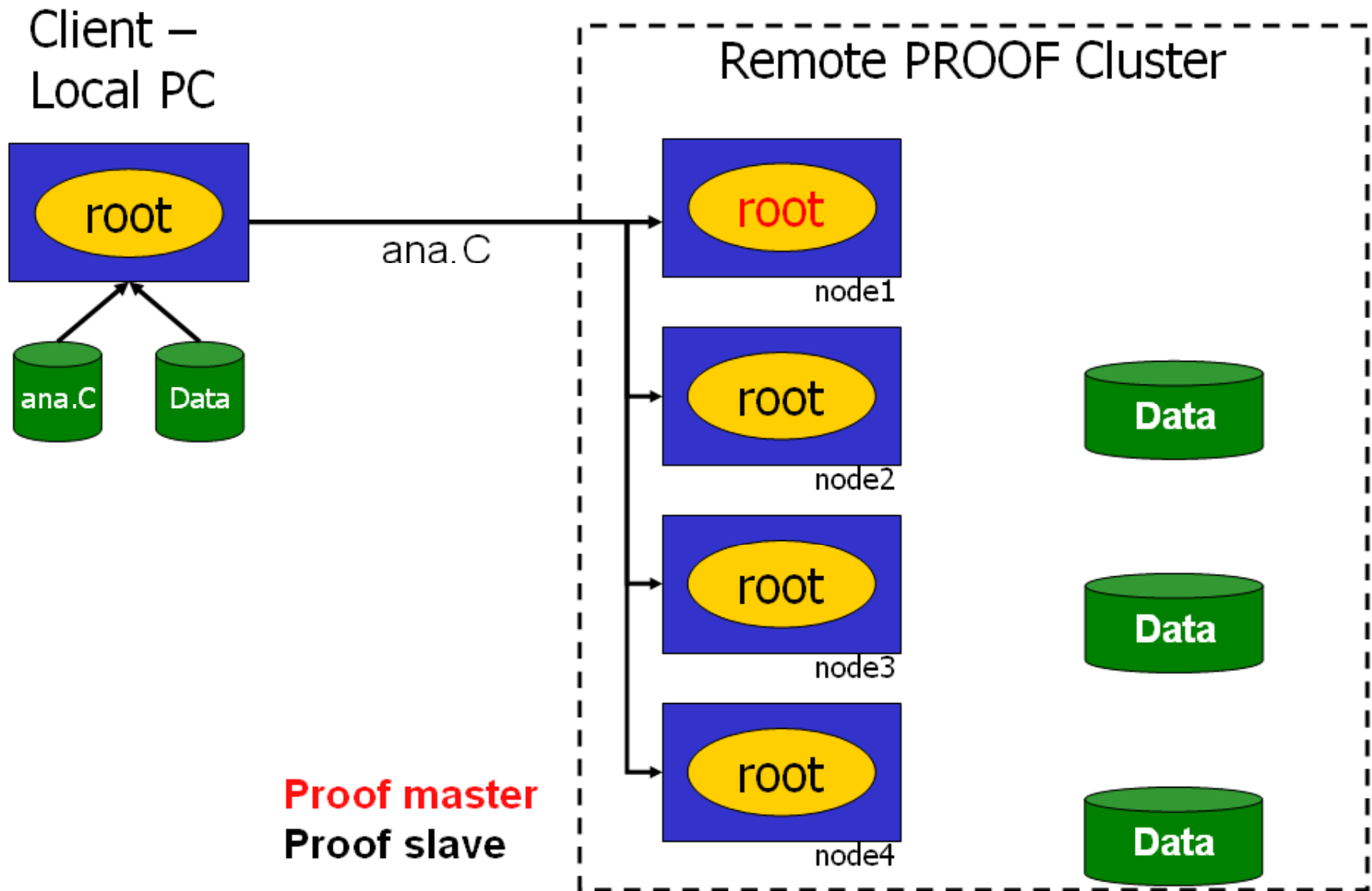
# PROOF Schema



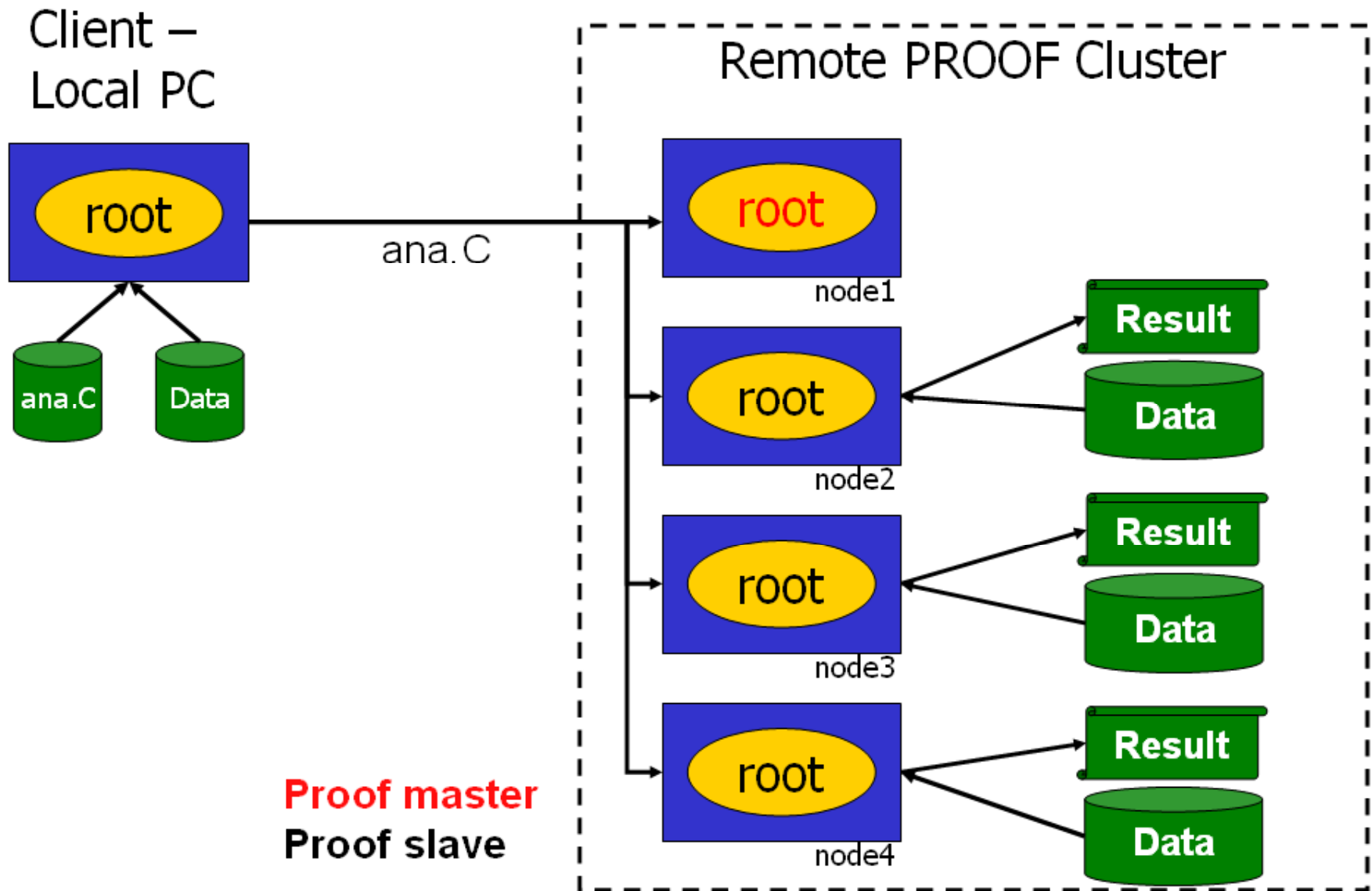
# PROOF Schema



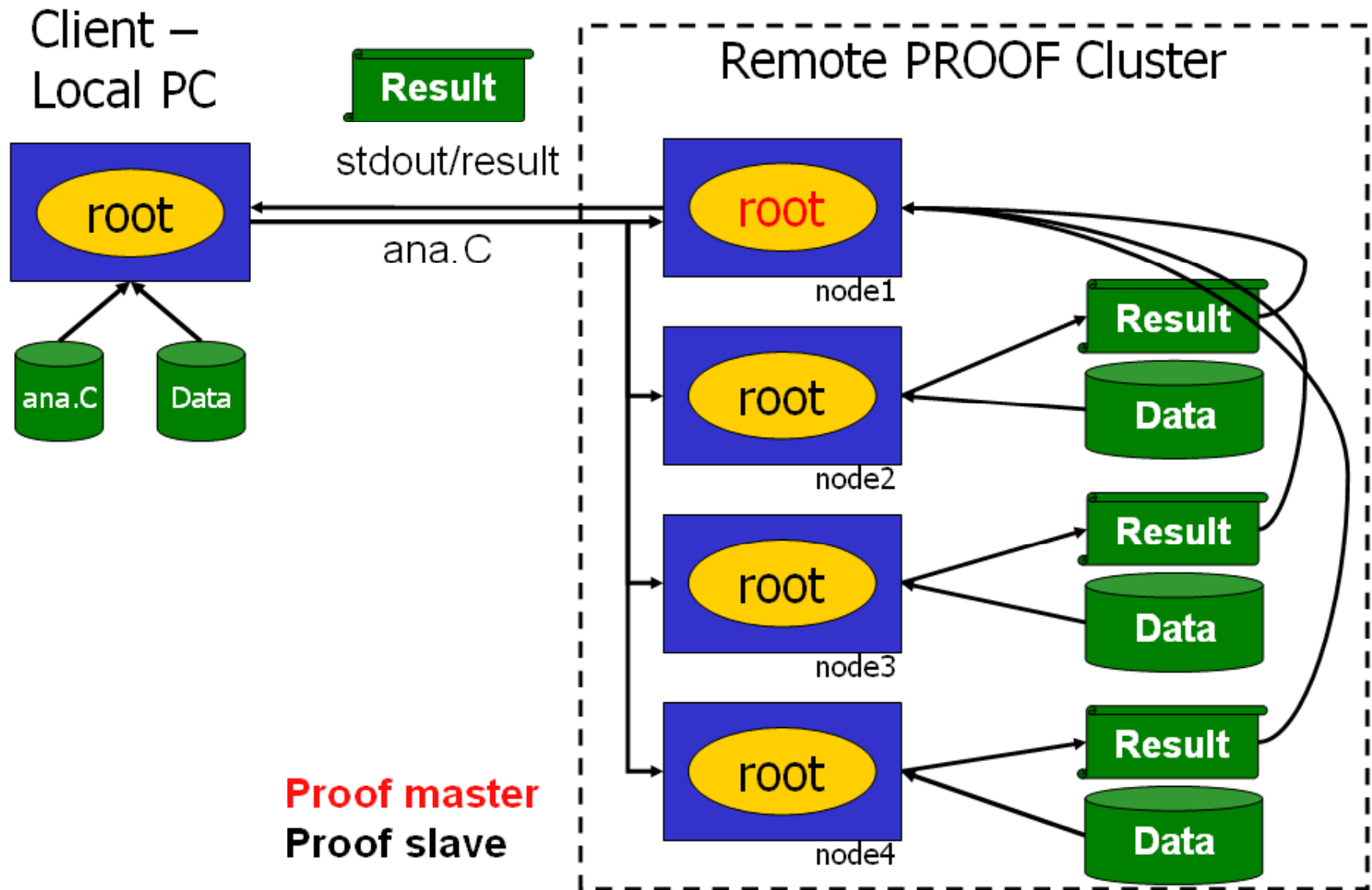
# PROOF Schema



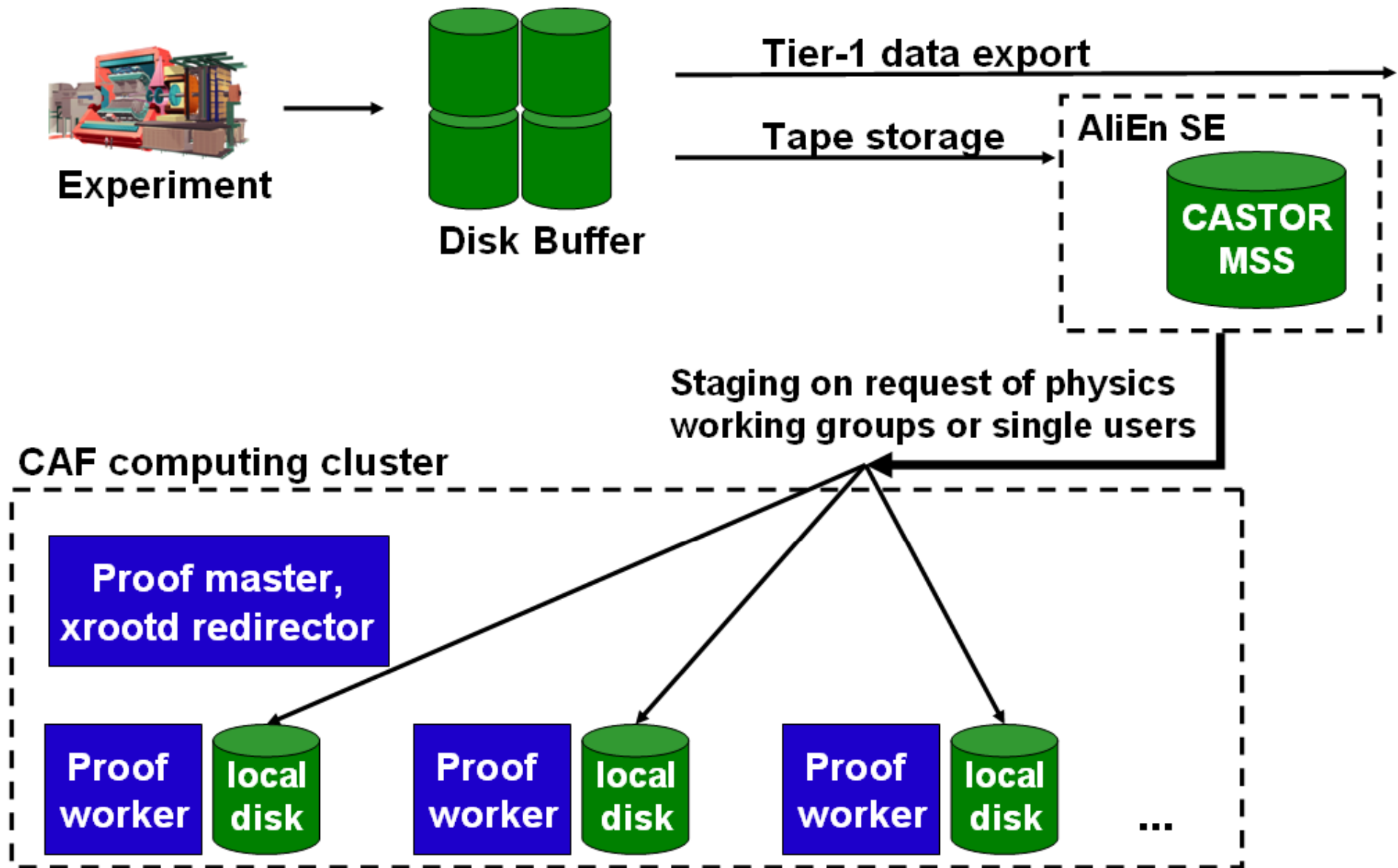
# PROOF Schema



# PROOF Schema



# CAF Schema



# Dataset concept

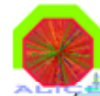
## PROOF Master / xrootd redirector



**PROOF master**

- registers dataset
- removes dataset
- uses dataset

**Dataset**



**data manager daemon**

- data manager daemon keeps dataset persistent by
- requesting staging
  - updating file information
  - touching files

**stage**



**olbd/xrootd**

- selects disk server and forwards stage request

**read, touch**

**PROOF worker / xrootd disk server (many)**

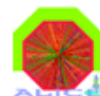


**olbd/xrootd**

**read**

**WN disk**

- stages files
- removes files that are not used (least recently used above threshold)



**file stager**

**write delete**

...

