Status of PROOF

G. Ganis / CERN

Application Area meeting, 24 May 2006

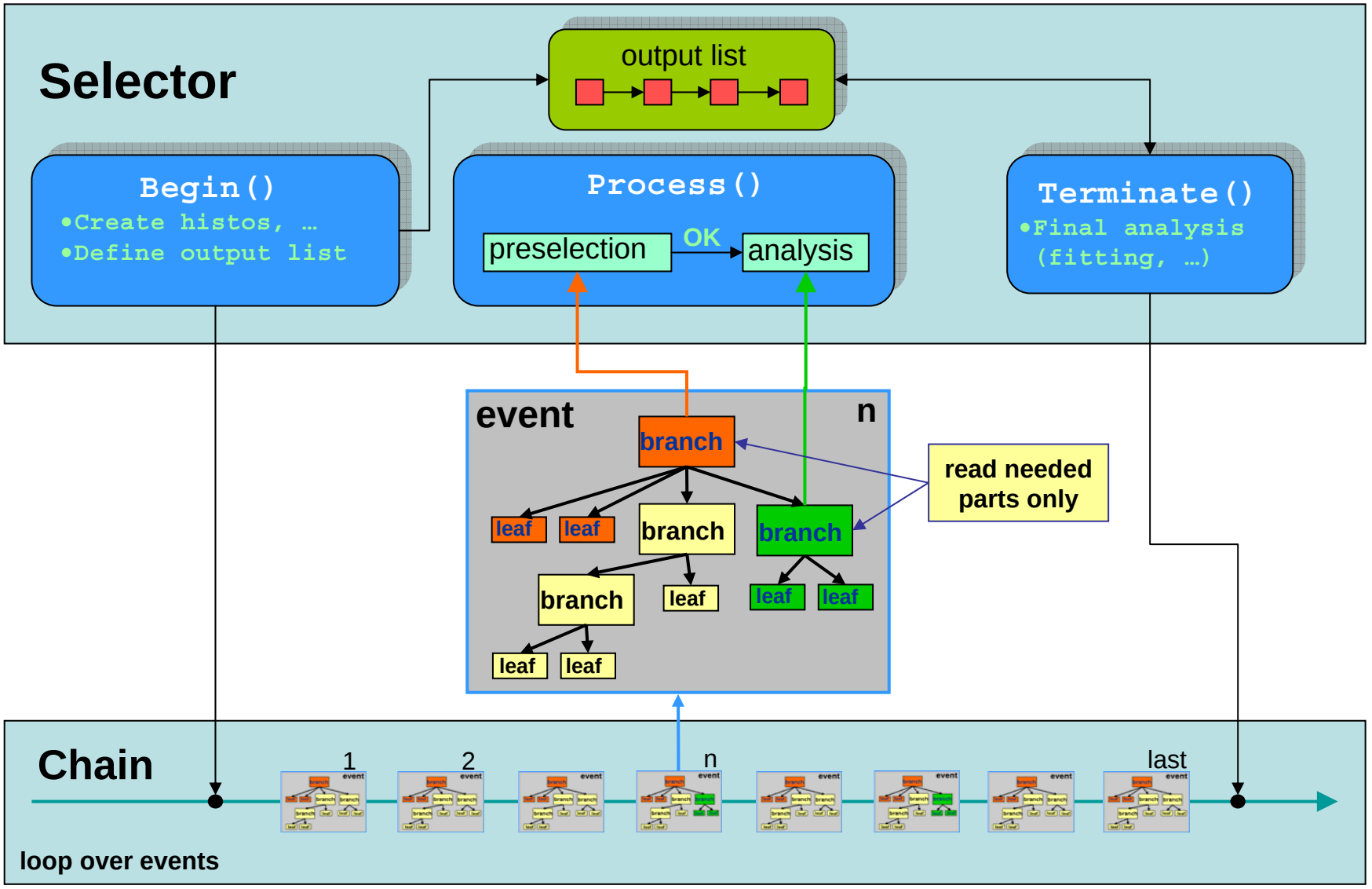
Outline

- Reminder about PROOF
- Recent developments and status
- Near future
- Testing at CAF
- Quick Demo

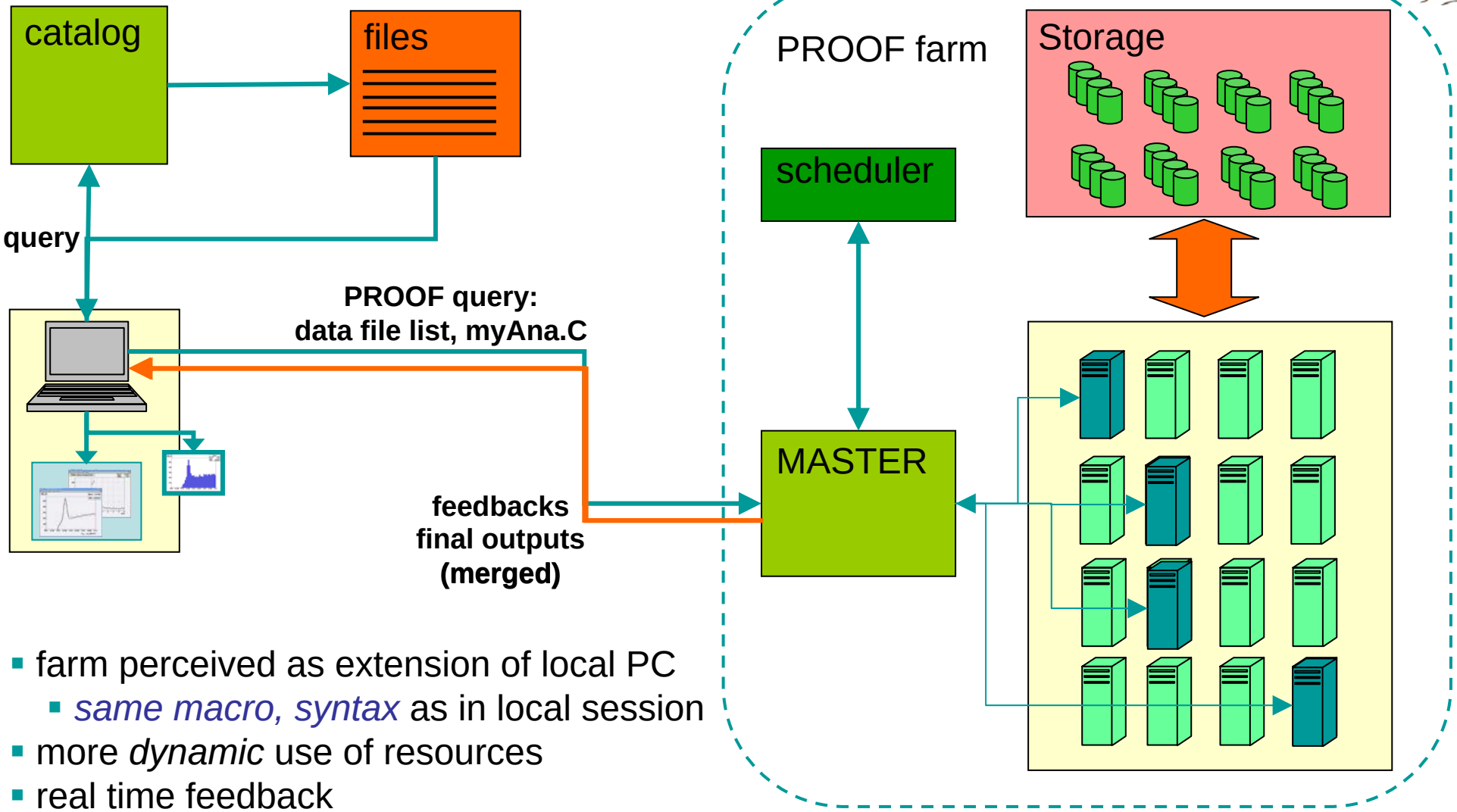
Outline

- Reminder about PROOF
- Recent developments and status
- Near future
- Testing at CAF
- Quick Demo

The ROOT data model: Trees & Selectors

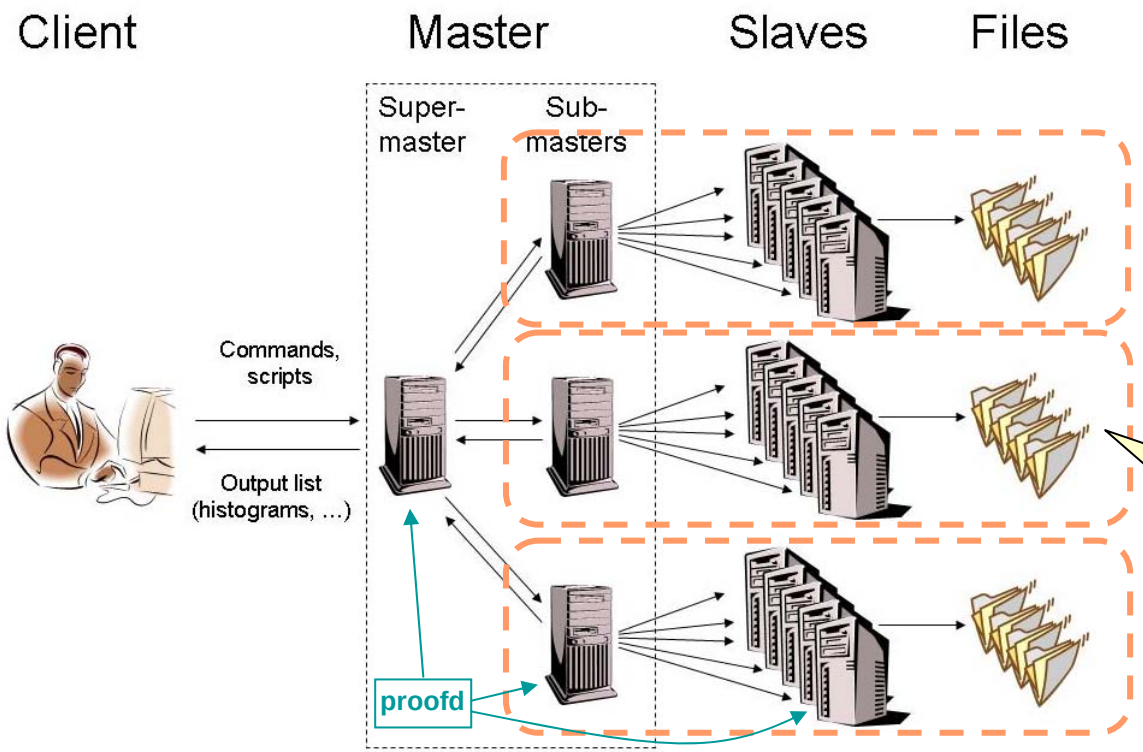


The PROOF approach



- farm perceived as extension of local PC
  - *same macro, syntax* as in local session
- more *dynamic* use of resources
- real time feedback
- automated *splitting* and merging

PROOF – Multi-tier Architecture



adapts to cluster of clusters or wide area *virtual clusters*

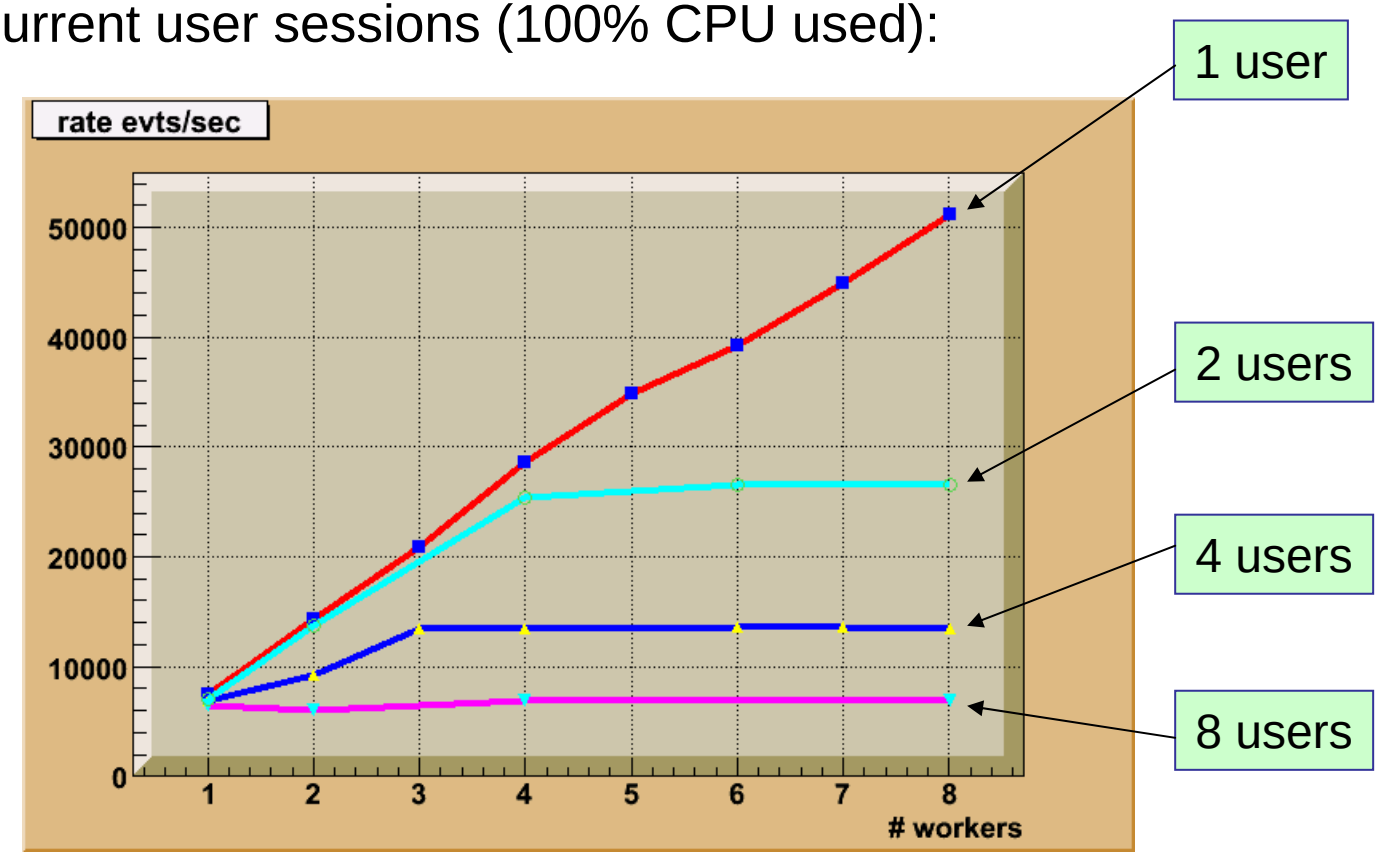
Geographically separated domains; heterogenous machine types

less important — good connection ? → VERY important

Optimize for data locality or efficient data server access

PROOF – Scalability

- CAF, 4 dual Xeon machines
- CMS selector, 120 MB data (290 files), distributed locally
- Strictly concurrent user sessions (100% CPU used):



- No inefficiency introduced by PROOF internals

Outline

- Reminder about PROOF
- Recent developments and status
- Near future
- Testing at CAF
- Quick Demo

Recent developments

- Goals:
  - support for interactive-batch mode
    - stateless connection
    - multi-sessions
  - user-friendliness
    - management tools
    - GUI

Sample of analysis activity

AQ1: 1s query produces a local histogram

AQ2: a 10mn query submitted to PROOF1

AQ3->AQ7: short queries

AQ8: a 10h query submitted to PROOF2

Monday at 10h15 ROOT session on my laptop

BQ1: browse results of AQ2

BQ2: browse temporary results of AQ8

BQ3->BQ6: submit 4 10mn queries to PROOF1

Monday at 16h25 ROOT session on my laptop

CQ1: Browse results of AQ8, BQ3->BQ6

Wednesday at 8h40 Browse from any web browser

New connection layer based on XROOTD

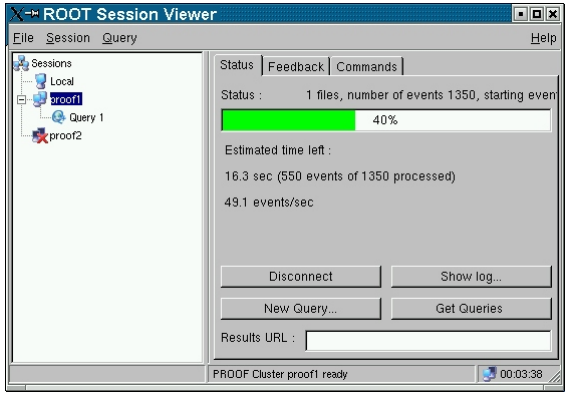
- Interactive batch requires a coordinator on the server side
- Candidate: XROOTD
  - light weight top component (networking, protocol handler)
  - new protocol implemented as a plug-in to launch and control PROOF server sessions
- Non-destructive disconnections handled naturally
  - stateless connection
- Xrd/olbd control network can be exploited to circulate information
- Can use same daemon for data serving and PROOF serving

PROOF management tools

- Data sets
  - distribution of data files on local worker pools
    - by direct upload
    - by staging out from a mass storage (e.g. CASTOR)
- Query results
  - classification and handling tools
    - retrieve, archive
- Packages
  - optimized upload of additional libraries needed by the analysis

GUI controller

Allows full *on-click* control on everything



- define a new session
- submit a query, execute a command
- query editor
  - execute macro to define or pick up a **`TChain`**
  - browse directories with selectors
- online monitoring of feedback histograms
- browse folders with results of query
- retrieve, delete, archive functionality
- start viewer for fast **`TChain`** browsing

Outline

- Reminder about PROOF
- Recent developments and status
- Near future
- Testing at CAF
- Quick Demo

Near Future Plans

- Data access (see next)
- Multi-user scheduling (see next)

- Packetizer optimizations
  - re-assignment of being-processed packets to fast idle workers
- Dynamic cluster configuration
  - come-and-go functionality for worker nodes
  - *olbd* network to get info about the load on the cluster

- Improve handling of error conditions
  - identify cases hanging the system, improve error logging, …
  - exploit *olbd* control network for better overview of the cluster
- Testing and consolidation

- Monitoring of cluster behaviour
  - MonAlisa: allows definition of *ad hoc* parameters, e.g. I/O / node / query

PROOF: data access issues

- **Low latency in data access is essential**

- File opening overhead
  - minimized using asynchronous open techniques
- Data retrieval
  - caching, asynchronous pre-fetching of data segments to be analyzed

- Asynchronous features supported by XROOTD / XrdClient

- Plan of work to fully exploit this in TFile / TSelector using the knowledge available in TTree (L. Franco)

PROOF: multi-user scheduling issues

- New *scheduler* component being developed to control the use of available resources in multi-user environments (J. Iwaszkiewicz)

- Decisions taken on per query base following a metric based on:
  - load of the cluster
  - resources need by the query
  - user history and priorities
  - …

- Requires support for dynamic re-configuration of worker assigments (prototype being tested)

- Generic interface to external schedulers planned
  - Condor, LSF, …

Outline

- Reminder about PROOF
- Recent developments and status
- Near future
- Testing at CAF
- Quick Demo

Testing at CAF

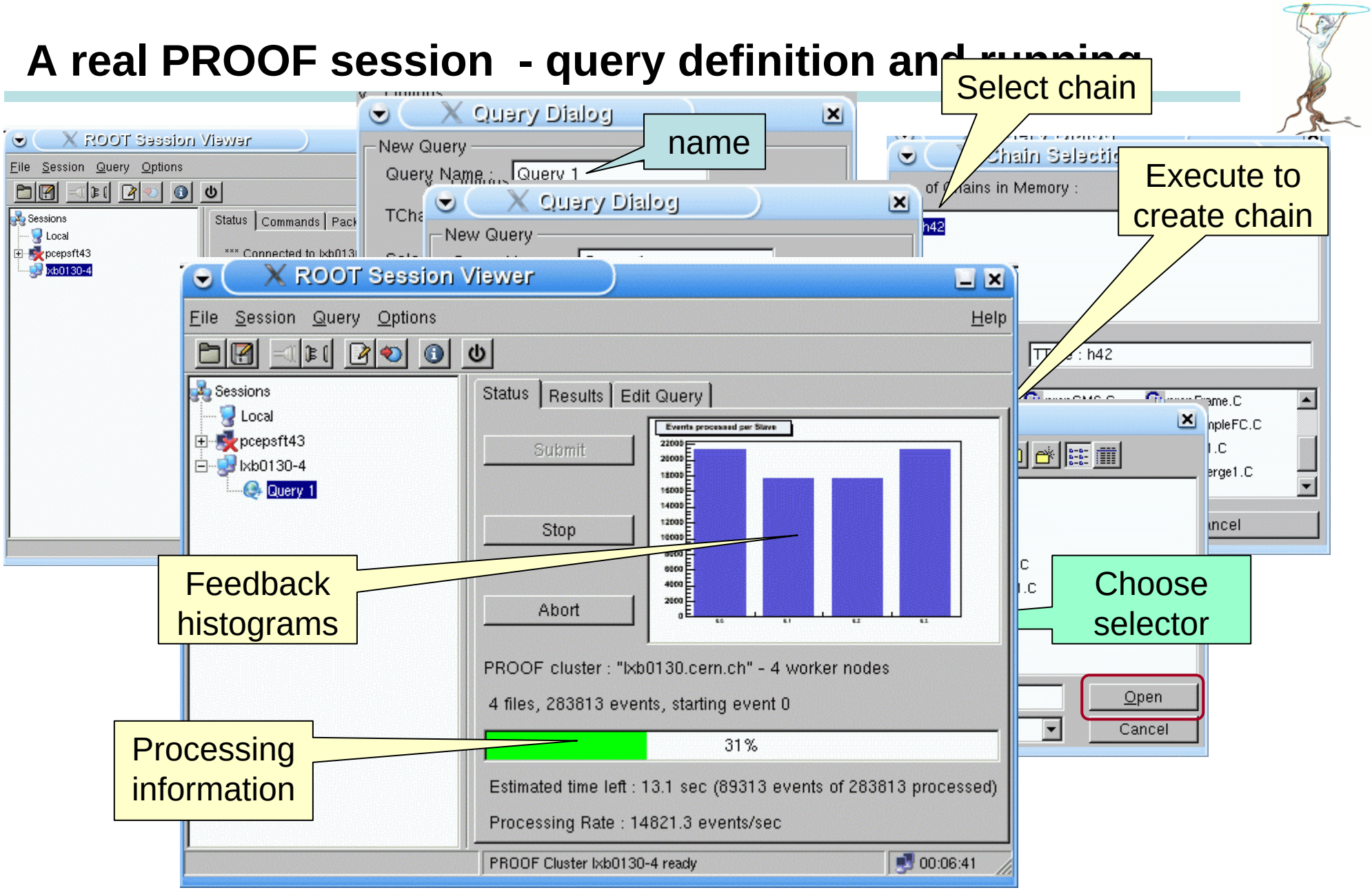
- CAF
  - 40 dual Xeon 2.8 GHz machines, 4 GB RAM, GB/s Ethernet
  - 215 GB disk pool
  - SLC4
- lxcafdev: 5 machines
  - testing developments
- lxcaf: 35 machine tested by ALICE (J.F. Grosse-Oetringhaus)
  - stress testing functionality
  - performance tests

Quick demo

- CMS test analysis selector
- 290 files (115 MB) distributed on lxcafdev

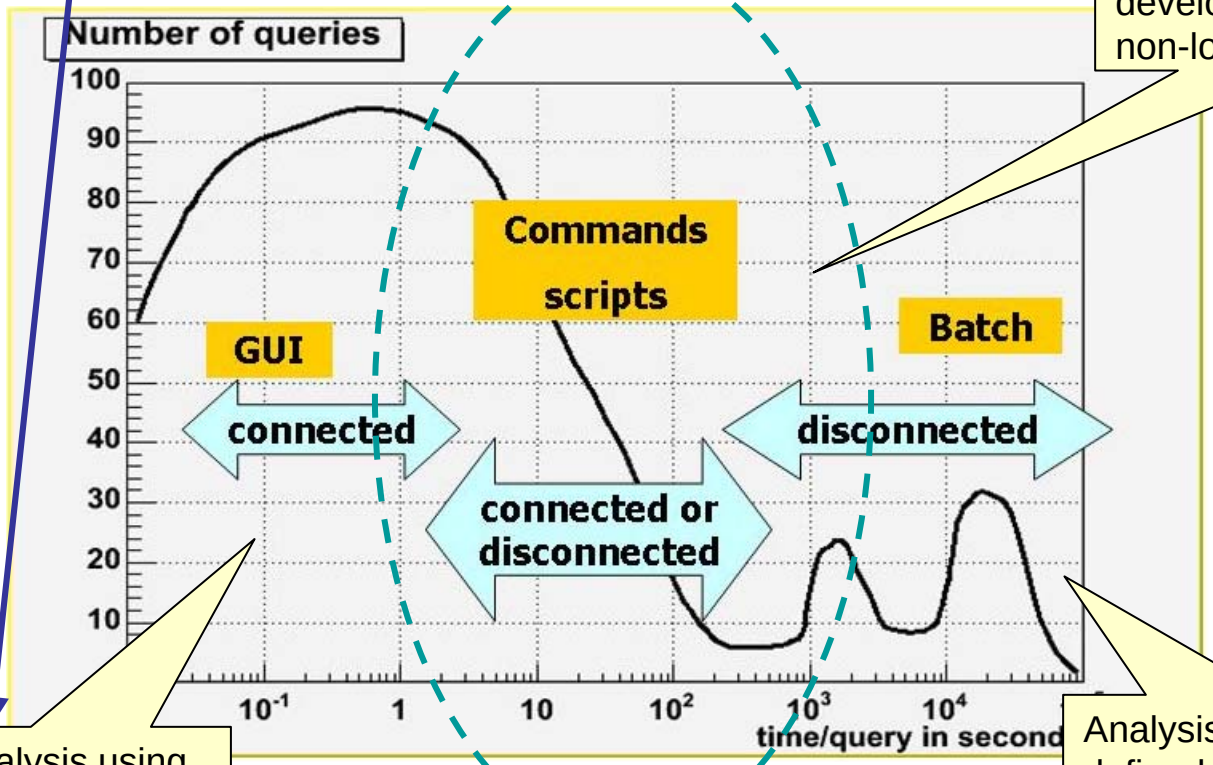
- Local run accessing files from lxcafdev
- PROOF run with 8 workers

A real PROOF session - query definition and running



Typical end-user job-length distribution

Goal: bring these to the same level of perception

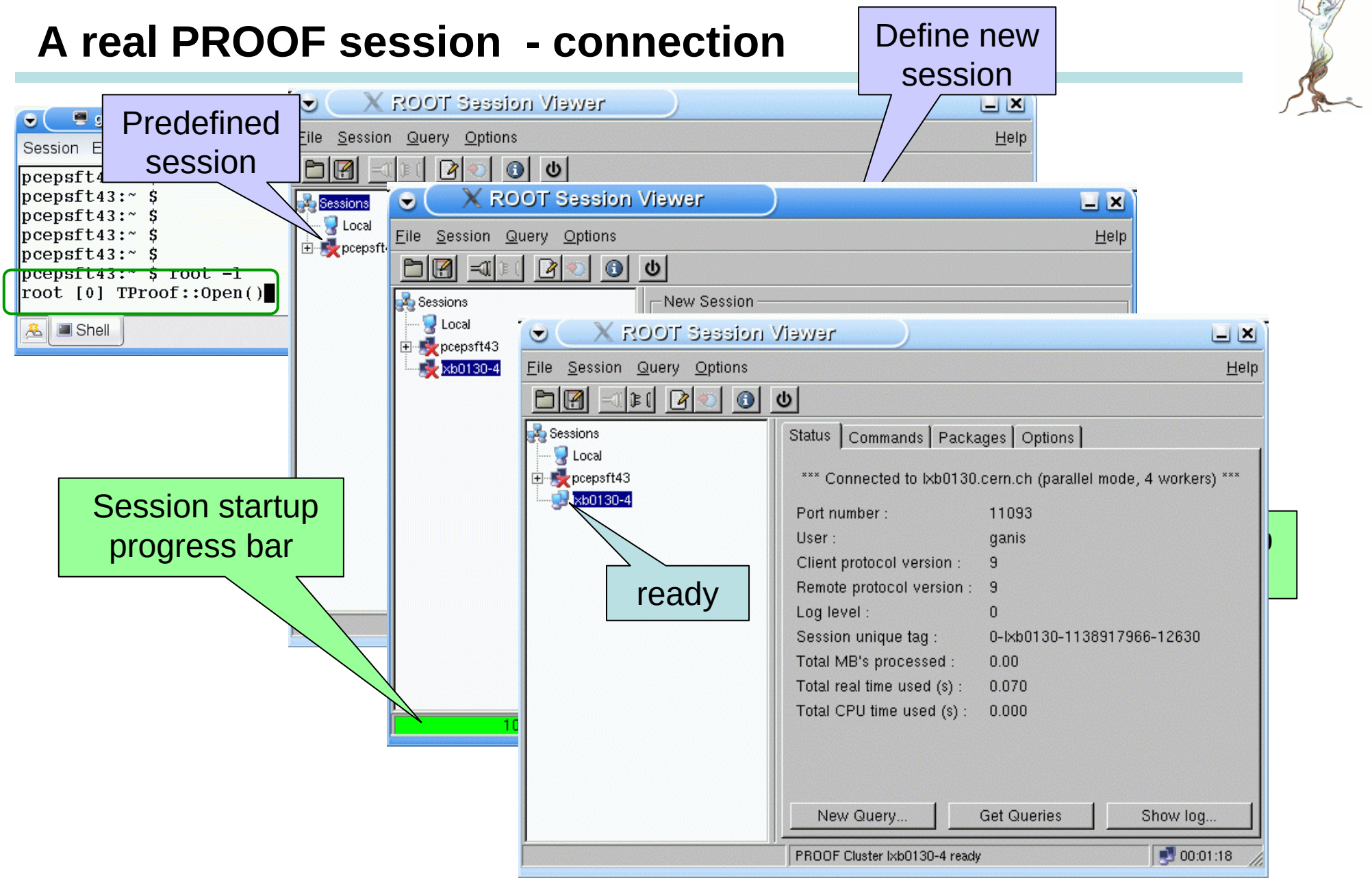


Medium term jobs, e.g. analysis design and development using also non-local resources

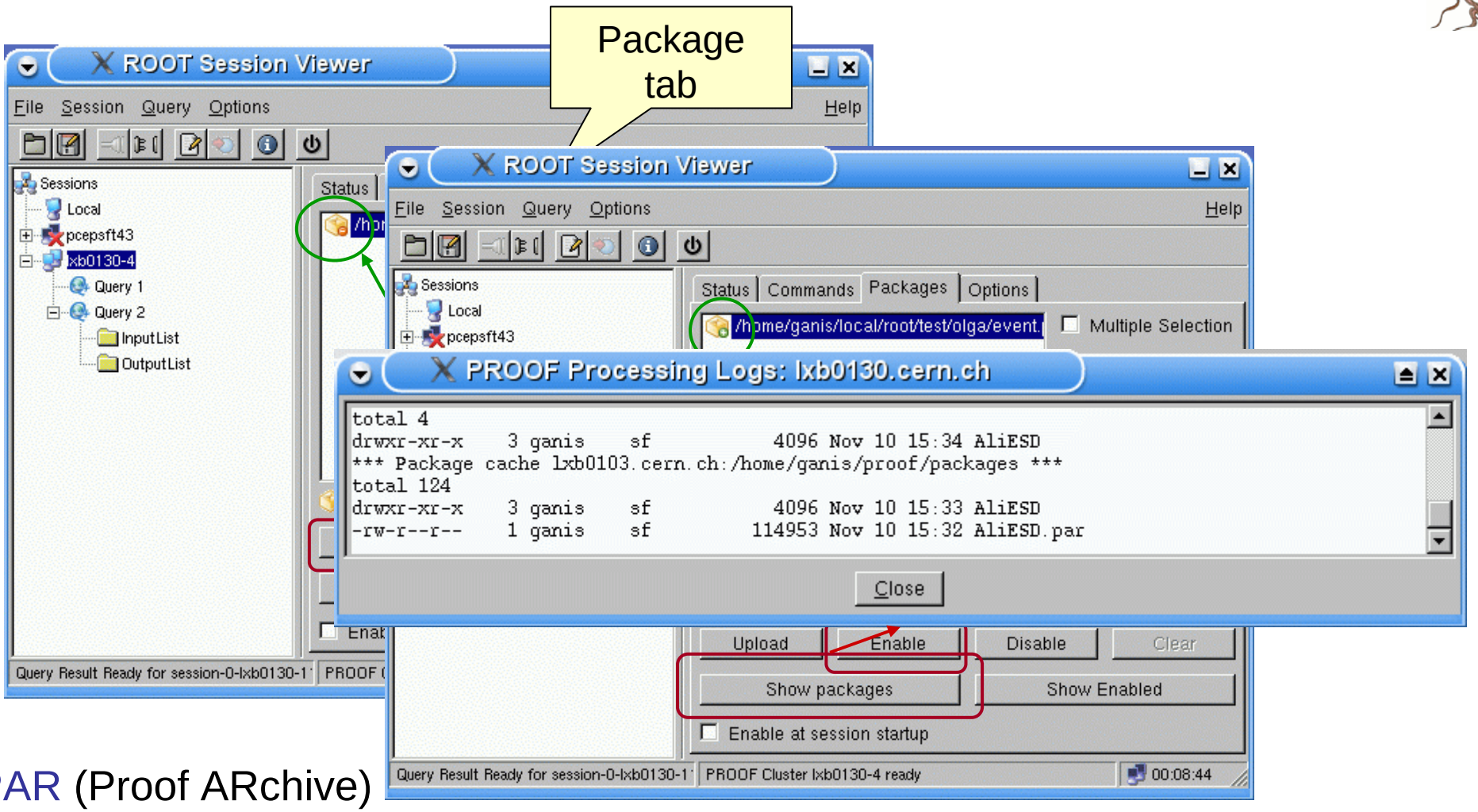
Interactive analysis using local resources, e.g.
- end-analysis calculations
- visualization

Analysis jobs with well defined algorithms (e.g. production of personal trees)

A real PROOF session - connection



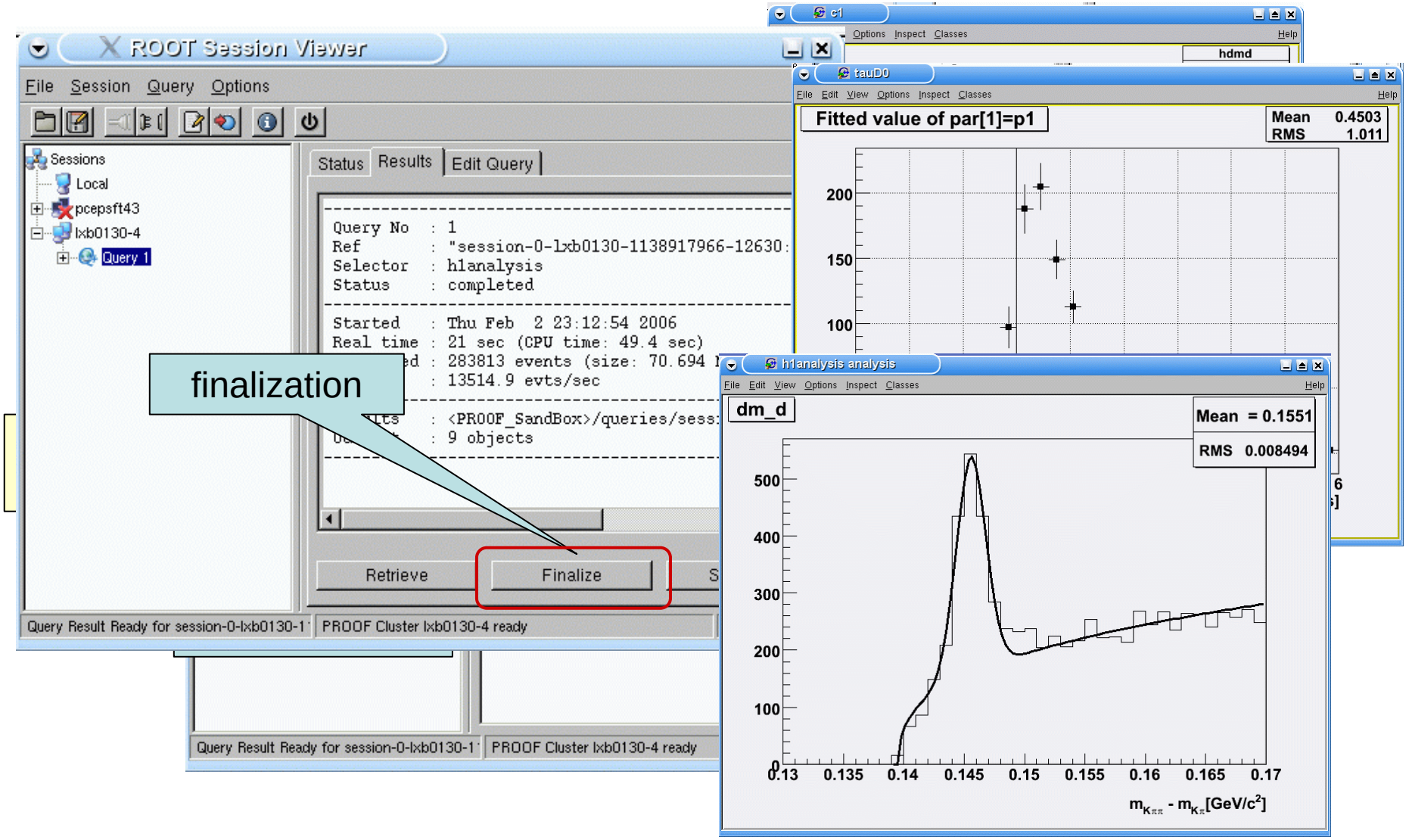
A real PROOF session - package manager



PAR (Proof ARchive)

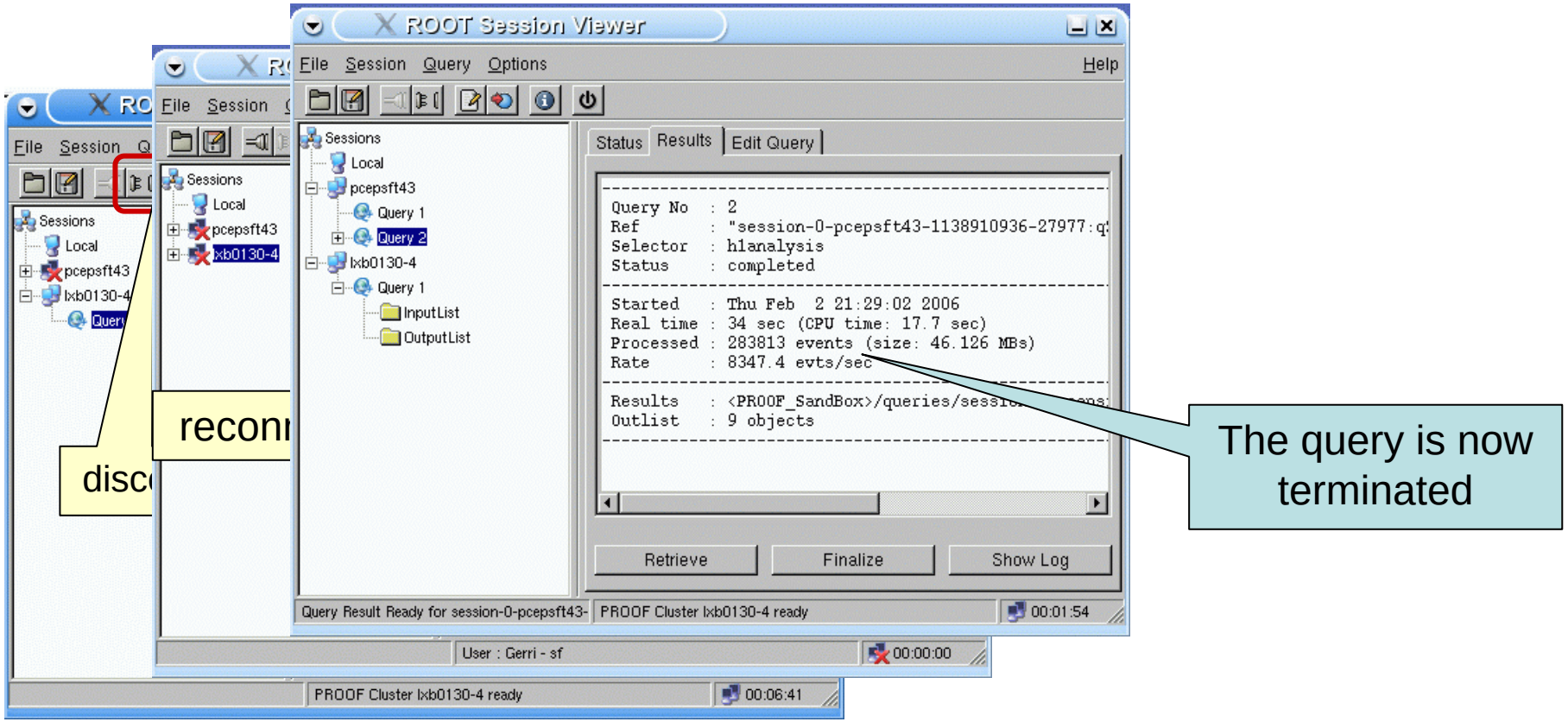
- ROOT-INF directory, BUILD.sh, SETUP.C
  - Control setup of each worker

A real PROOF session: query browsing and finalization

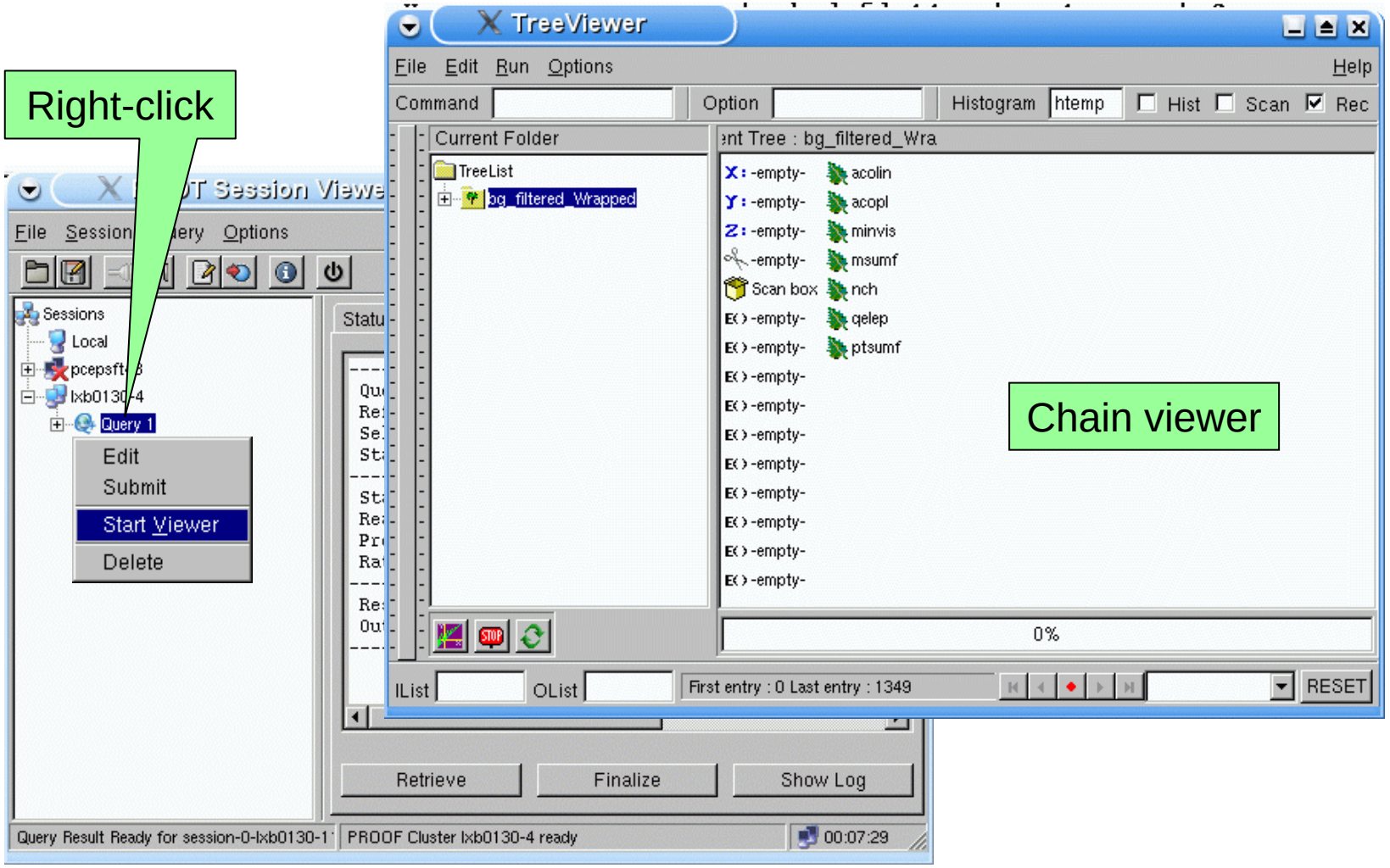


A real PROOF session: disconnection / reconnection

- Running sessions kept alive by server side coordinator
- Reconnection is much faster: no process to fork

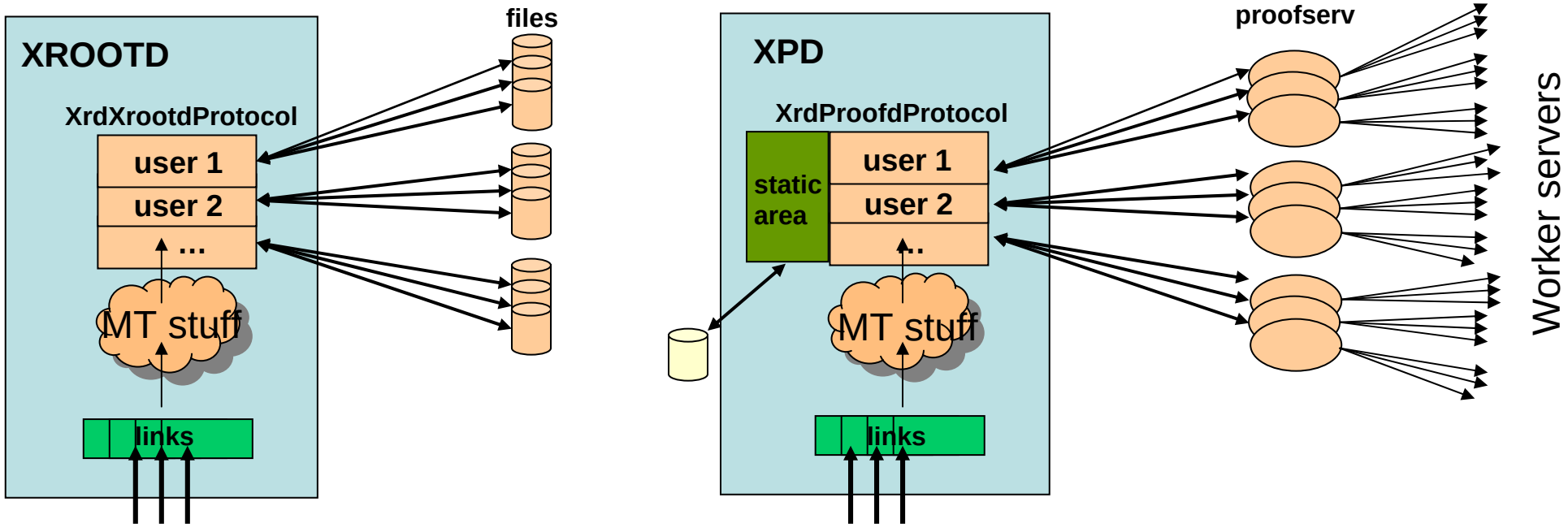


A real PROOF session: chain viewer



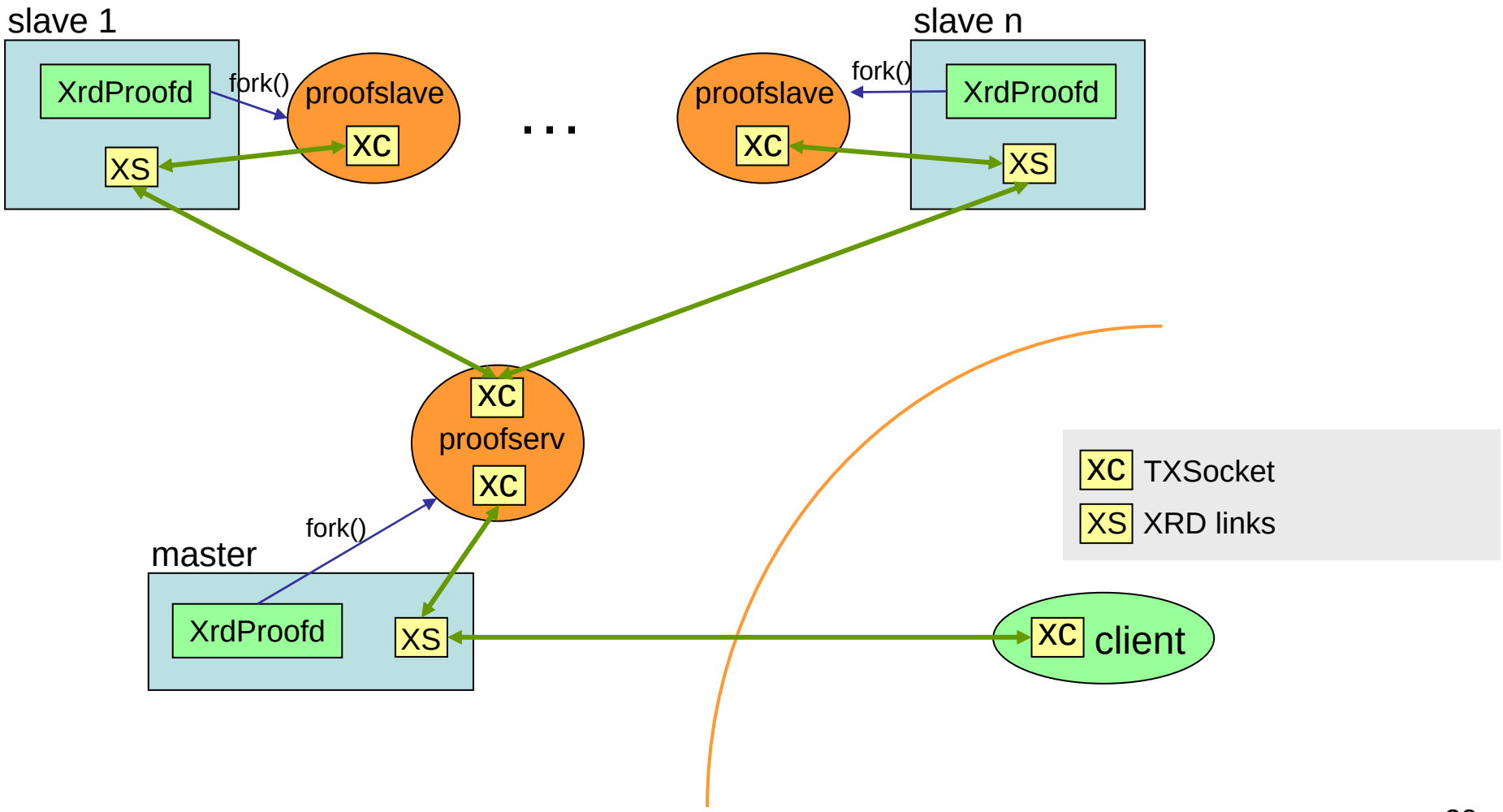
XrdProofd basics

- Prototype based on XROOTD

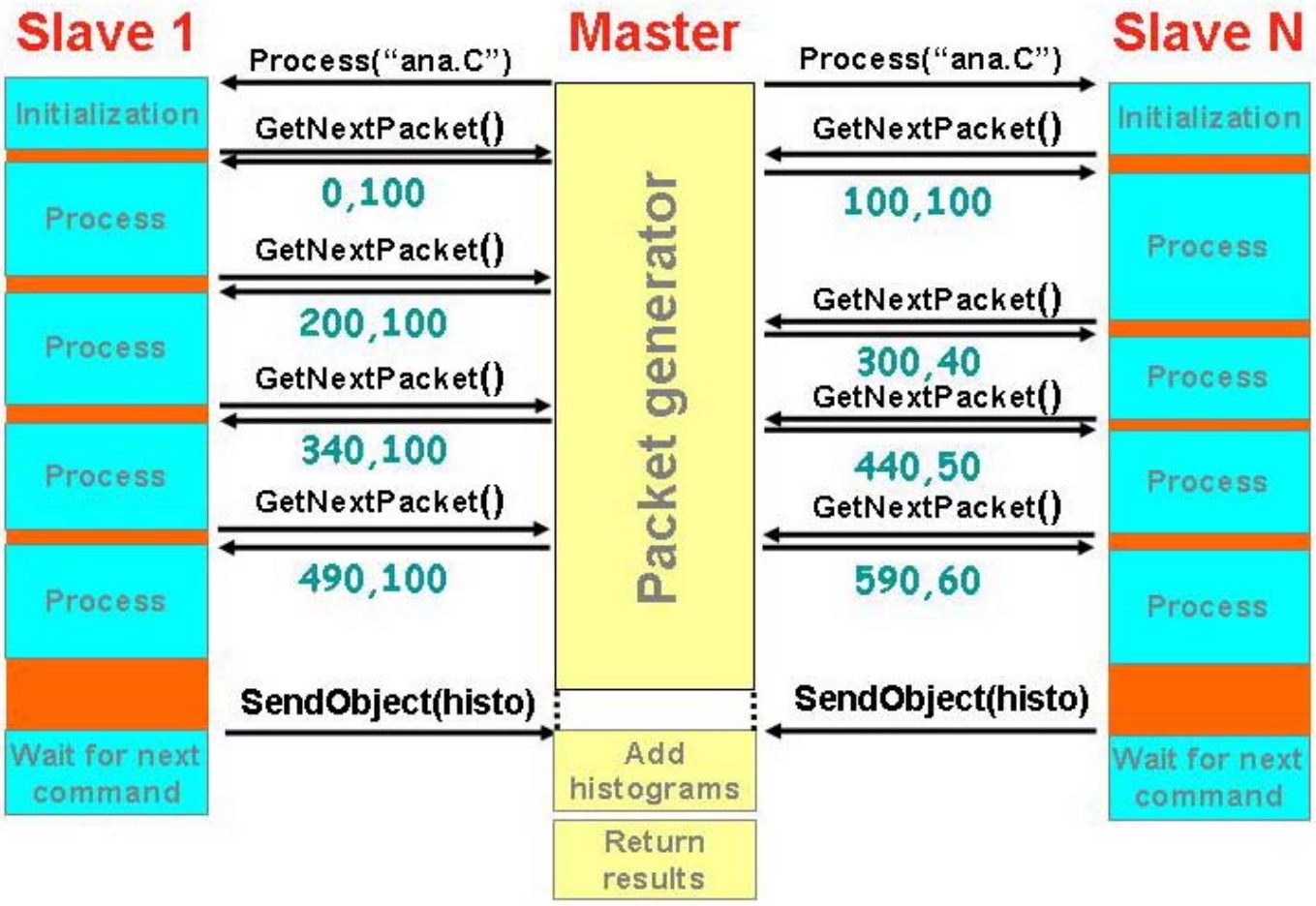


- XrdProofdProtocol: client gateway to proofserv
- static area for all client information and its activities

XrdProofd communication layer



Workflow: Pull Architecture



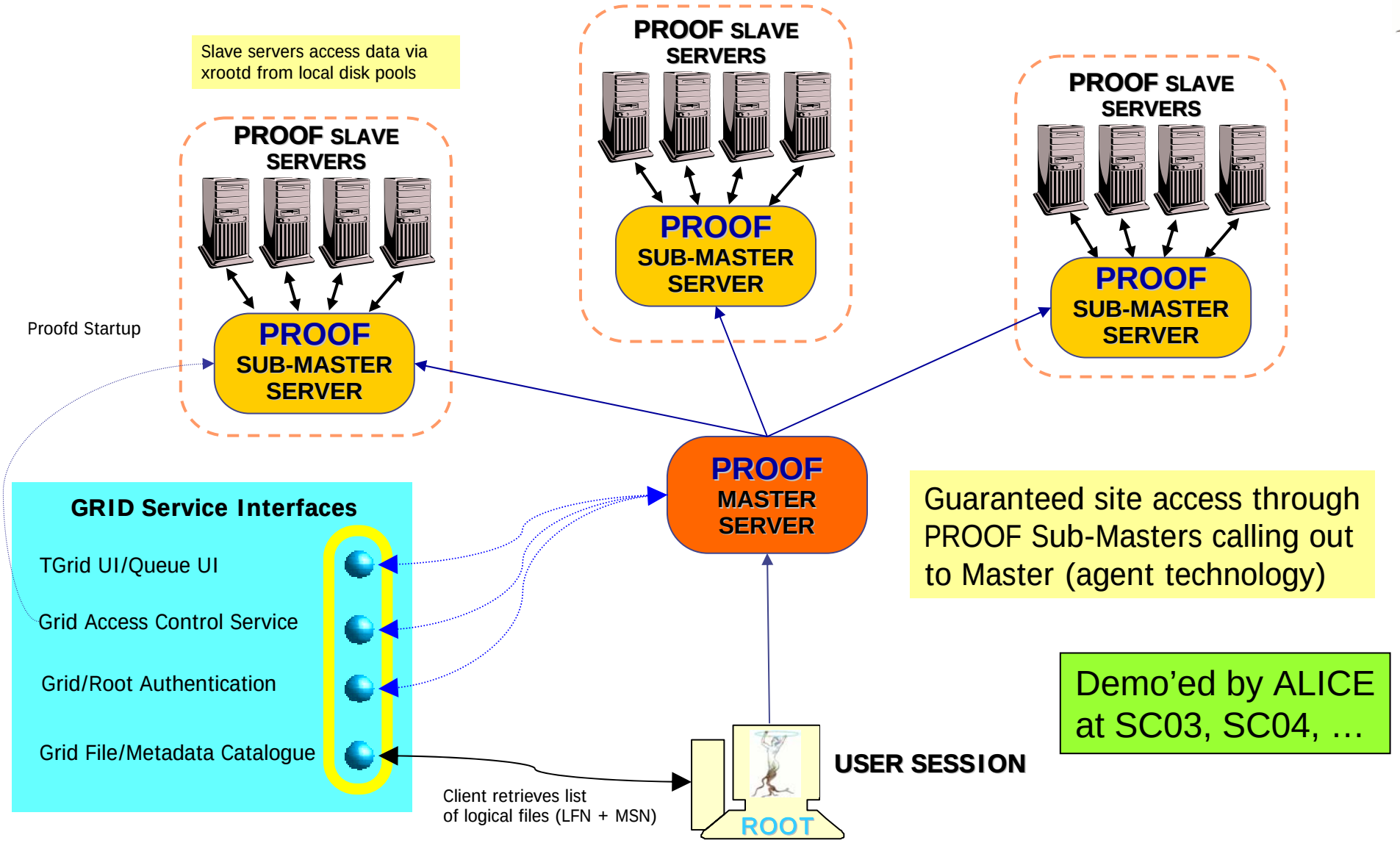
dynamic load balancing naturally achieved

PROOF @ AliEn: command-line session

- **TGrid**: abstract interface for all services

```
// Connect
TGrid *alien = TGrid::Connect("alien://");

// Query
TString path= "/alice/cern.ch/user/p/peters/analysis/miniesd/";
TGridResult *res = alien->Query(path, "*.root");

// Create chain from list of files
TChain chain("Events", "session", res->GetFileInfoList());

// Open a PROOF session
TProof *proof = TProof::Open("proofmaster");

// Process your query
chain.Process("selector.C");
```

PROOF @ GRID


Slave servers access data via xrootd from local disk pools
PROOF SLAVE SERVERS
PROOF SUB-MASTER SERVER
PROOF SLAVE SERVERS
PROOF SUB-MASTER SERVER
PROOF SLAVE SERVERS
PROOF SUB-MASTER SERVER
Proofd Startup
PROOF MASTER SERVER
GRID Service Interfaces
TGrid UI/Queue UI
Grid Access Control Service
Grid/Root Authentication
Grid File/Metadata Catalogue
Client retrieves list of logical files (LFN + MSN)
ROOT
USER SESSION
Guaranteed site access through PROOF Sub-Masters calling out to Master (agent technology)
Demo'ed by ALICE at SC03, SC04, …