Xrootd – an outlook

Fabrizio Furano

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Outline

- A big question
- What is it, how it evolved
- How it should be used
- How it was used
- What's missing (less)
- Support
- Open items

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

A big question

WHY

- Is it so special ?
- Did it survive ?
- Should I use it ?
- Will (not) it solve any problem ?

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

At the beginning

- Filling a big hole in the market
- Absence of a SCALABLE and ROBUST solution for HEP HPC
- Need to interconnect it with other HEP-related systems
- Need to deploy transparent Hierarchical Storage facilities
- Building an efficient LOCAL storage cluster was nearly impossible
  - Always the same use cases: reco, user analysis, custodial storage etc.
  - HEP has PARTICULAR requirements
    - Still partially covered by modern DFSs + tons of glue code

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

The not so basic requirements

- Data access as immediate as possible
  - Especially if the files are already where expected
- “Basic primitives” Open, rw, close, stat, prestage
  - Born for analysis-like random access and Hierarchical Storages
- “Easy” Cluster-to-cluster data movements: just a robust cp tool
- Deep-knowledged integration with ROOT
  - Fast file open, efficient random workloads, tough “intelligent” readaheads in multiuser systems… if used!
- Fast filename translations (= global namespace coherency)
  - Shield apps from the local mountpoints of sites
  - Also may create a coherent namespace ACROSS sites
- Quality
  - Always headed to the modern HEP reqs, although very generic
  - Efficient use of the available hw resources
  - Self-contained (avoids messy dependencies)
  - Super-robust communication, fault tolerance

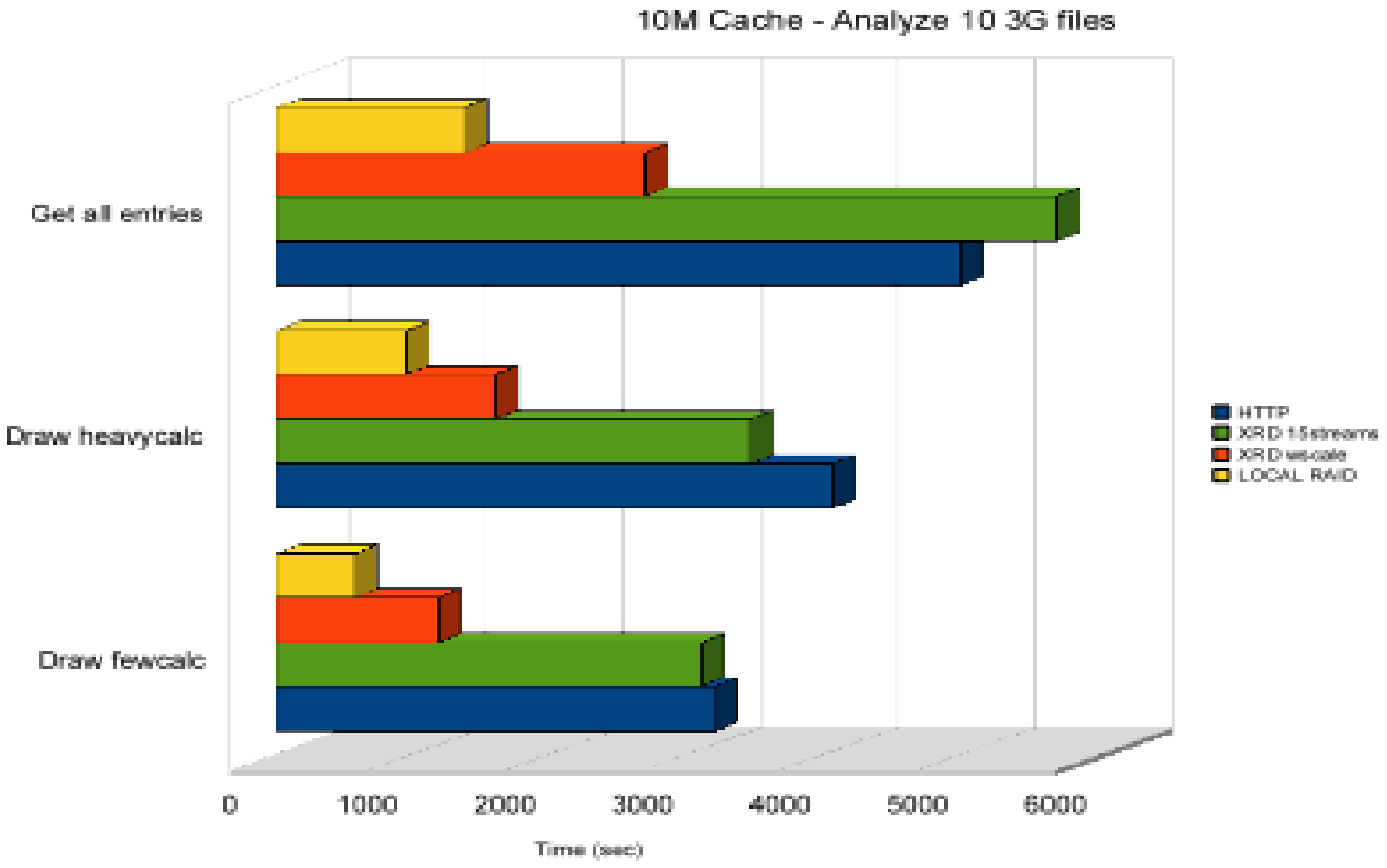
CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

What came next

- Robust communication also means that it worked via WAN, somehow
- The idea was to make it work WELL via WAN
  - Big effort started in ~2005 (async primitives and caching, multistreaming)
  - Exploited the ROOT TTreeCache as it came out (ReadV, then async reads, now async ReadV)
- Natural idea: aggregating WAN-wide storages (~2008)
- Next idea: WAN federations of sites helping each other
- Bigger outcome: possibility to simplify the glue code + get more quality

From CHEP09

CALTech 10GB/180msRTT + TTreeCache+Xrd latency hiding
More than one year ago, now things should be even better



CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

What does it do

- Hierarchical storage aggregation and access in their strictest meaning
- Plugin-based structure to override the behavior

- From tiny clusters to WAN-wide monster ones with linear scalability
  - They all look the same
- P2P-like internal file discovery, no internal catalogue
  - The files you put are the files you get
    - "You must know the files you've put and their meta-data" ™
  - No danger of misalignment, by definition

- Subset of the POSIX semantics, to support scalability
  - This subset was recently extended (Data Management Client-side functions, Jan 2010)
  - Allows the creation of replicas (replica-agnostic)
- Relies on external systems to do the right things
  - Like accessing the files which are present
  - Bad idea: doing 'find' on a PB repository
  - Another: 'ls' flooding on a PB repository to decide what to do

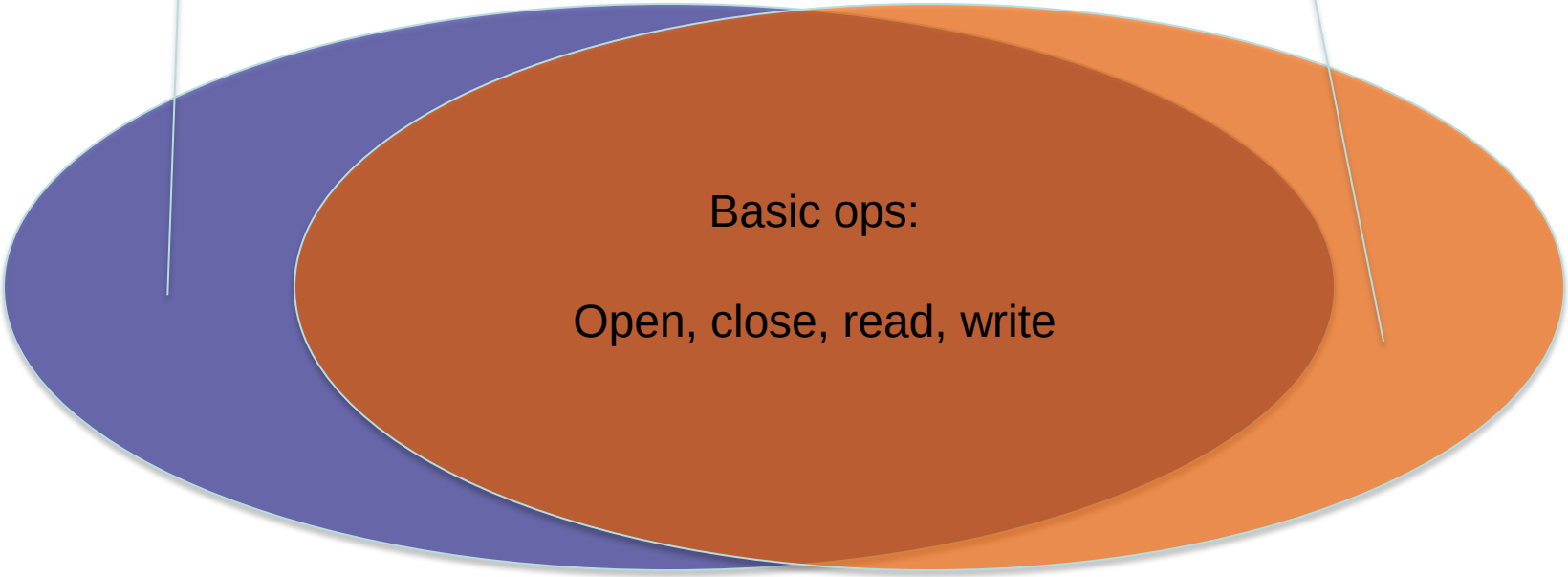
CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

POSIX vs XROOTD

What becomes possible and what does not

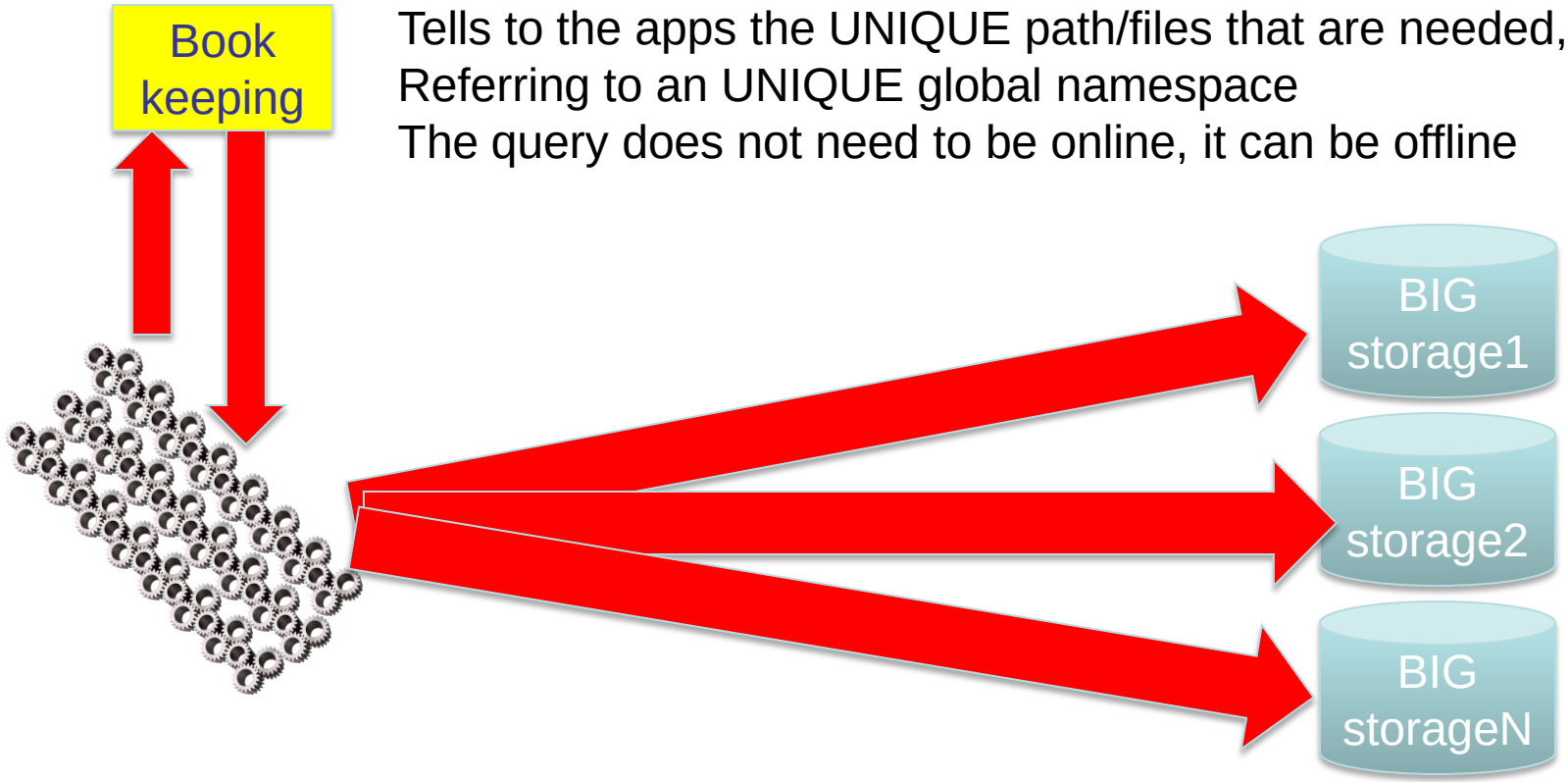
Full
File system
Semantics

Powerful async features
"No-action" server side fault tolerance
Lighweight, speed and scalability
Improved client side fault tolerance



CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

The “expected” architecture



Tells to the apps the UNIQUE path/files that are needed,
Referring to an UNIQUE global namespace
The query does not need to be online, it can be offline

Many variations are possible, sometimes they have a high price.

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

The “expected” architecture

- Design for scalability, this does not come for free.
- Enforces:
  - Separation data/metadata(app-specific)
  - Abstraction from the data location, in the local cluster or ev. In the global domain
  - performance
    - Once a dataset metadata (for an analysis) is ready, no need for the user to re-query it
    - Access the storage only when the job has settled down, gives time to the DB part (ev. Quasi-offline)
- Hides:
  - SE type: tapes/no tapes etc. all look the same
  - They only have different characteristics
    - Fast, slow, far, close, big, small
  - Tapes can be used transparently. The difference is in knowing how to do it.
- Allows direct access to several SEs at once, via LAN and WAN

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Core software process

- HEP-adapted AGILE process
- Small and committed core team
- The basic requirements are well known
  - Leave abundant space for extensions
- Get user's feedbacks as FAST as possible
- Respond as FAST as possible
  - The next beta is now, tagging takes 10 minutes
  - If you don't try it out it means that you are not interested
  - The next stable is "very soon", when the tests succeed
    - But, by definition, the subsequent tag will be better
- Filter requirements and requests
  - Virtuous ones get immediate priority

Security: Authentication

- XrdSec by Gerri Ganis (~2005): one of the best pieces of software around
- Authentication support for virtually any protocol
- The protocols were added one at a time, often as contributions
  - UNIX
  - Simple Shared Secret
  - GSI
  - Kerberos
  - VOMS extensions

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Security: Authorization

- Simple plugin-based infrastructure, called XrdAcc
- At least 2 plugins around
  - The default one: simple but effective (and static)
  - AliceTokenAcc: the very efficient ALICE schema

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Distributions

- The ‘vanilla’ one, from the SLAC CVS
  - Requires considerable knowledge and top quality support
  - Full access to any kind of weird configuration/platform
- The ‘xrd-installer’, from the good old ARDA repository
  - Stays to the ‘vanilla’ one like Ubuntu stays to the Linux kernel
  - Supports multiple ‘bundle flavors’
  - Refurbished several times, very simple and stable. Supports data globalization, Virtual MSS and ML monitoring.
  - Pre-configured, reduces support requests to a minimum
  - Historically used by the ALICE sites (~60 SEs)
- The VDT setup
  - Recently under heavy refurbishment after evaluating the ‘xrd-installer’
  - Seems headed towards a middle point between the other two
  - AFAIK Mainly used in the US, mostly ATLAS-driven

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Support

- Historical top-level: the usual contributing experts (e.g. Andy, Gerri, Fabrizio, Wilko, Jacek, Andreas, Derek, Artem etc.)
  - Whoever contributes a part typically gives support for that part
  - Often on a best-effort basis but well done so far
- Internally-grown ALICE knowledge, good for first level support
- AFAIK, OSG is setting up a more formal structure

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

The millionth test

- We have seen so many, sometimes with strange contradictory results
- Large scale tests can be precious
  - Difficult to do
  - Can generate useful feedback, thus improving quality
    - A “secret test” (but publishing the results) does not make life better
    - It declares the unwillingness to have a system which performs well
    - In other words: is your test performing bad? We are the first who want to understand why and make it better.
    - Moreover, can a test of very old stuff give indications for the future? I doubt.
- Not many such tests around now, probably a good sign

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Every feature you like

- Xrootd is a fully plugin-based system
  - Developing new plugins can give whatever feature
  - E.g. PROOF, or XrdCASTOR
- Full POSIX-like namespace compliance+additional HEP reqs+unique hexabyte storage is an open research problem
- Weakened push towards reasonable usage patterns and architectures?
  - E.g. ‘ls’ floodings, namespace-unaware systems, useless file aliasings, apps wasting file resources, …
  - Difficult to tell.

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Open items (1/2)

- Hot: putting clients into servers
  - Or: different criteria to fully differentiate clients in the same app
  - E.g. How to instantiate together:
    - a client tuned for WAN TTreeCache-based random access
    - one optimized for blasting non-TTreeCache LAN traffic ?
    - one optimized for large files xfers
- The Extreme Copy (beta since ~1 year): Torrent-like dynamic multiserver file fetching
  - Needs the previous item to be really strong
- Components for site cooperation
  - Proxies and caching proxies (nothing more than proofs of concept right now)
  - Bandwidth/queuing manager (early alpha)
- 'Personal' persistent caching proxy (or TXNetFile/XrdClient extensions)
- A full-featured command line interface
  - The current one is a quite rough tool

Open items (2/2)

- Client-side data management funcs (e.g. ‘recursive ls’ or ‘df’): good level but incomplete by now
- WAN performance: huge breakthrough, still to gain
  - Both for file xfer and data access (TTreeCache and not)
- A robust and complete server-to-server file copy
- ROOT integration: very good quality, but still to gain
  - Only partially asynchronous (XrdClient can be fully async instead)
  - Will be more evident with the parallelization of the computing, I/O will likely become the bottleneck again
- “Intelligent” readaheads. After the ATLAS AOD story (Feb) I got no feedbacks…
  - The backport to ROOT 5.22 was done AFAIK
  - Andrei Maslennikov was the only feedback “well, you could try this…” Very partial but still something.
  - Interesting topic, many things could be done.
- An homogeneous, top-class support structure
  - Quality helps support, but vice-versa is true as well

Thank you

Questions?

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it