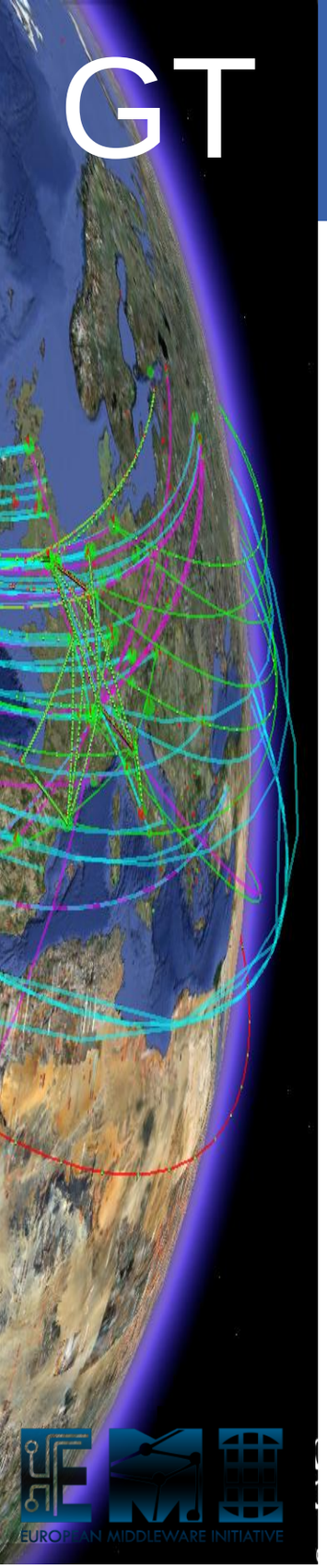


CERN IT Department

Dynamic Federations

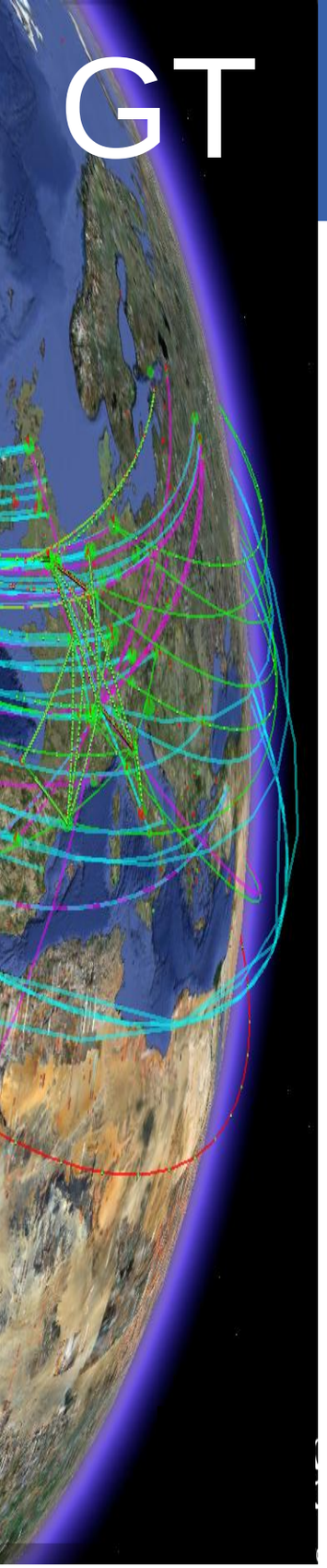
Seamless aggregation of open-protocol-based storage endpoints

Fabrizio Furano (presenter)
Patrick Fuhrmann
Paul Millar
Daniel Becker
Adrien Devresse
Oliver Keeble
Ricardo Brito da Rocha
Alejandro Alvarez

Credits to ShuTing Liao (ASGC)

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

CERN

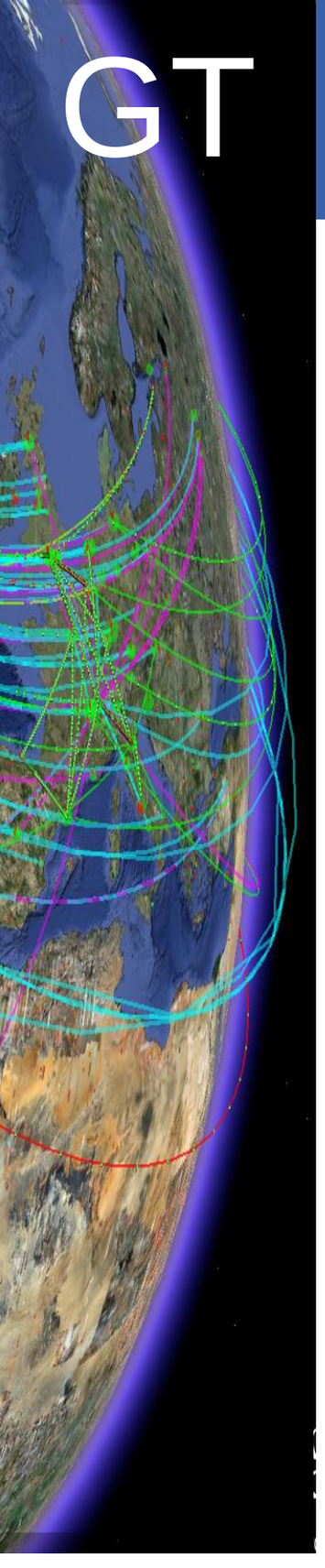


Dynamic Federations

CERN IT Department

- Federating storage means giving seamless access with thin, standard clients
  - without the need of switching communication protocol
- We know several use cases
  - Easy, direct job/user data access, WAN friendly
  - Failover is part of the concept
  - Friend sites can share storage and bandwidth
  - Easy access to shared things like conditions
  - Automatically detect if the endpoint is working
  - Federate un-indexed fast changing things, like SQUID caches
  - Federate third party outsourced HTTP/DAV servers (also clouds)
  - Federate the information of one or more experiment's DBs (e.g. two LFCs), also considering what's in the SQUID caches worldwide
    - Transparent, direct access to the official replicas AND the cached ones as well
  - Support for advanced site choices, e.g. self healing
- We like HTTP and DAV
- We want to take the concept to the next level

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



Dynamic Federations

CERN IT Department

- Technically “loosely coupled storage systems”
- Idea: a single entry point for a federation of endpoints
  - “lonely”, un-indexed storage clusters (e.g. dCache, DPM, plain HTTP servers)
    - Caches fall in this category
  - site/VO catalogues (e.g. LFCs) indexing their storage elements
- Idea: **make it dynamic**
  - The task of federating is done ***on the fly***, just communicating with the endpoints
- This entry point knows its endpoints, can redirect clients to them, it can merge and present their metadata to browsing clients

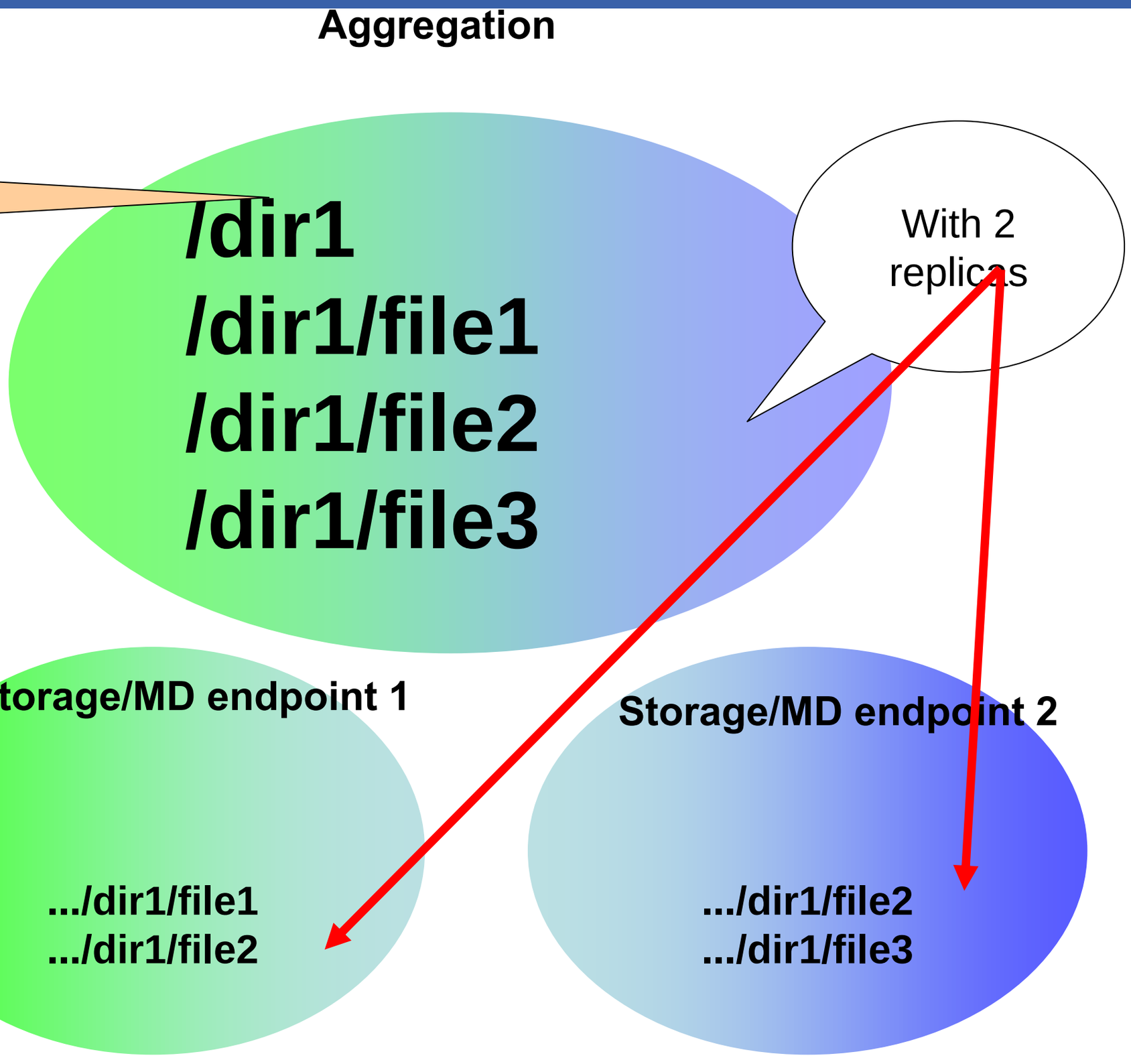
CERN IT Department
CH-1211 Geneva 23
Switzerland
www.cern.ch/it

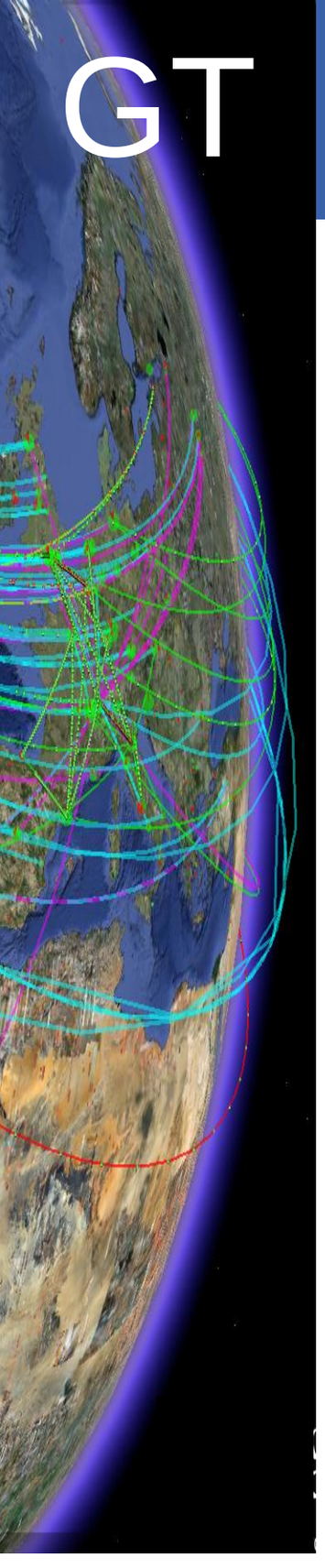
The basic idea

We see this

All the metadata interactions are hidden

NO persistency needed here, just efficiency and parallelism





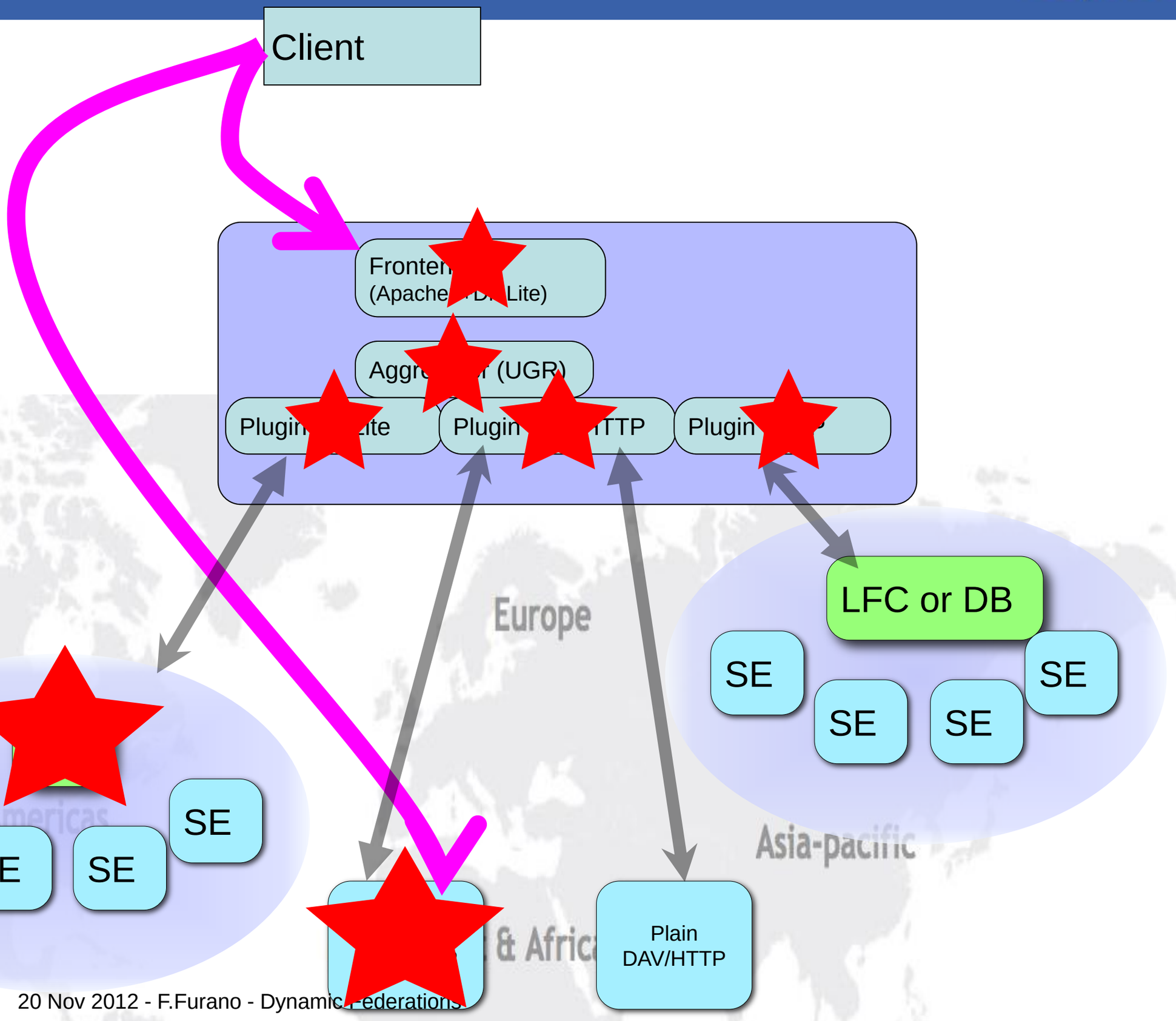
Dynamic Federations

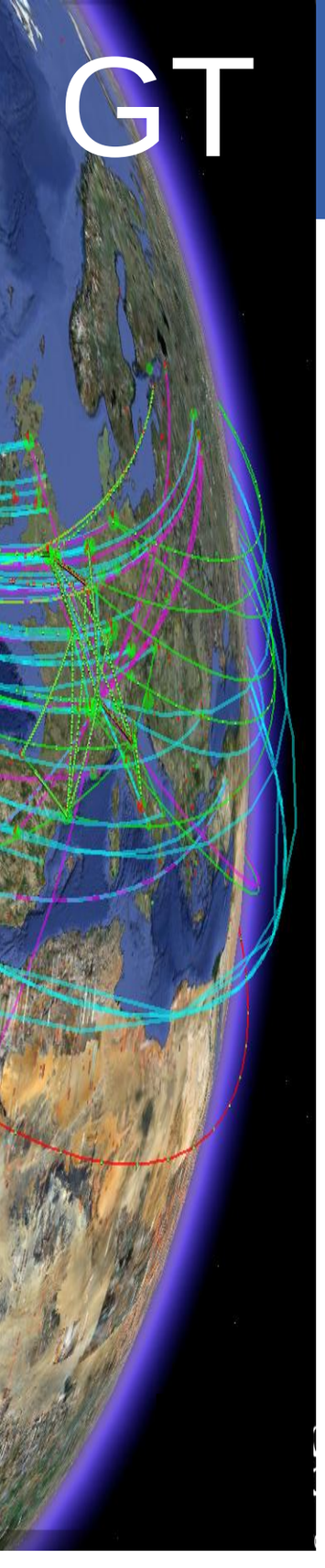
CERN IT Department

- What's the goal?
  - Make different storage clusters be seen as one
  - Make global file-based data access seamless
- How should this be done?
  - No strange APIs, everything looks "banal"
  - Participate in the HTTP HEP/HPC ecosystem that is growing now
  - Use dynamic systems that are easy to setup/maintain:
    - no complex metadata persistency
    - no DB babysitting (keep it for the experiment's metadata)
    - no replica catalogue inconsistencies, by design
  - Head to high performance
- Local SE as a preference, give the freedom to point to an efficient and reliable global federation
  - Optimize redirections based on **on-the-fly client-data proximity**
  - Avoid inconsistencies, just looking at where the files are now, and at the real status of the endpoint (working/not working)
  - Limit complexity: read only (by now), as usually writes happen to well-known, close islands

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

Example

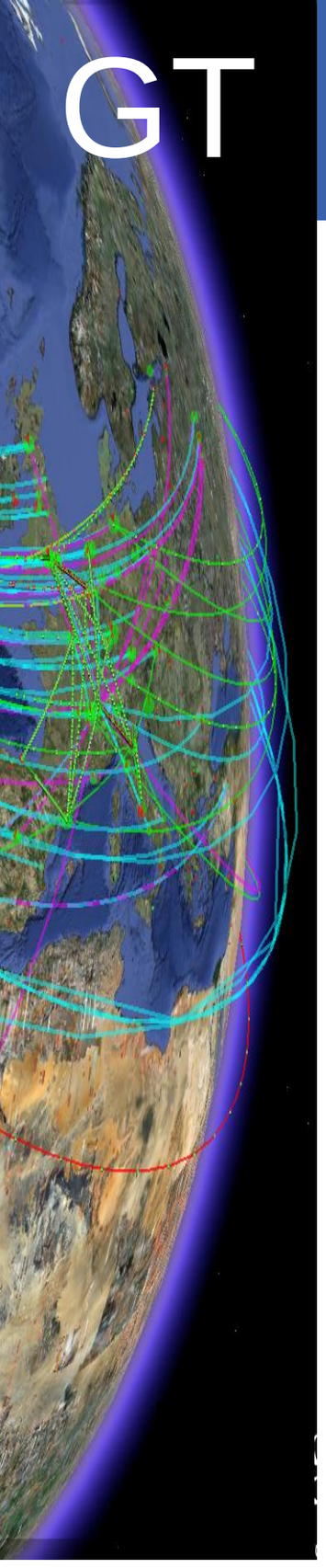

Client
Fronten
(Apache
Lite)
Aggr
(UGR)
Plugin
ite
Plugin
TTP
Plugin
LFC or DB
SE
SE
SE
SE
SE
SE
SE
SE
Plain
DAV/HTTP
Europe
Asia-pacific



Demo

- We have a stable demo testbed, using HTTP/DAV
  - Federator head node in DESY: http://federation.desy.de/myfed
  - Federating a number of endpoints. This list changes, as we agree on what to federate:
    - a DPM instance at CERN
    - a DPM instance at ASGC (Taiwan)
    - a dCache instance in DESY
    - a Cloud storage account by Deutsche Telecom
    - two endpoints in LBNL
- The feeling it gives is surprising
  - Metadata performance is in avg much higher than contacting the endpoints
- We see the directories as merged, as if it were only one system
- 10K files are interleaved in a 4-levels deep directory /myfed/dteam/ugrtest/interleaved
  - Oddly-numbered files are at CERN
  - Evenly-numbered files are at Desy
- 10K files have 2 replicas in DESY and CERN: /myfed/dteam/ugrtest/all
- One more interesting directory, with replicated files: /myfed/atlas/fabrizio/
- When a choice is possible, clients are redirected to the endpoint that is closer to them

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

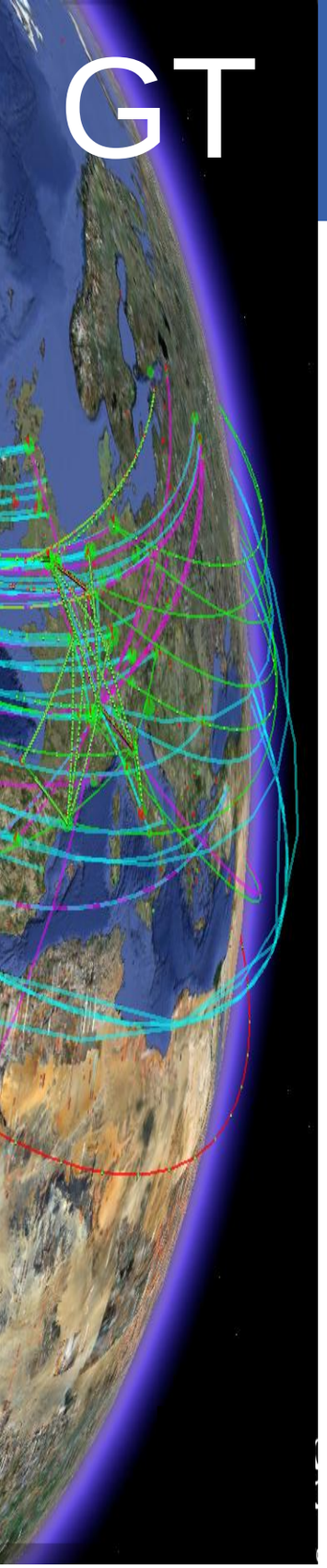


Dynamic Federations

CERN IT Department

- Currently in advanced beta, available!
- Technically TODAY we can aggregate:
  - dCache DAV/HTTP instances
  - DPM DAV/HTTP instances
  - LFC DAV/HTTP instances
  - Cloud DAV/HTTP services
  - Native LFC and DPM databases
  - Anything that can be plugged into DMLite (the new architecture for DPM/LFC)
  - **Can be extended to other metadata sources**
- The system also can load a “Geo” plugin
  - Gives a geographical location to replicas and clients
  - Allows the core to choose the replica that is closer to the client
- The Geo plugin that we wrote uses GeoIP (free)

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

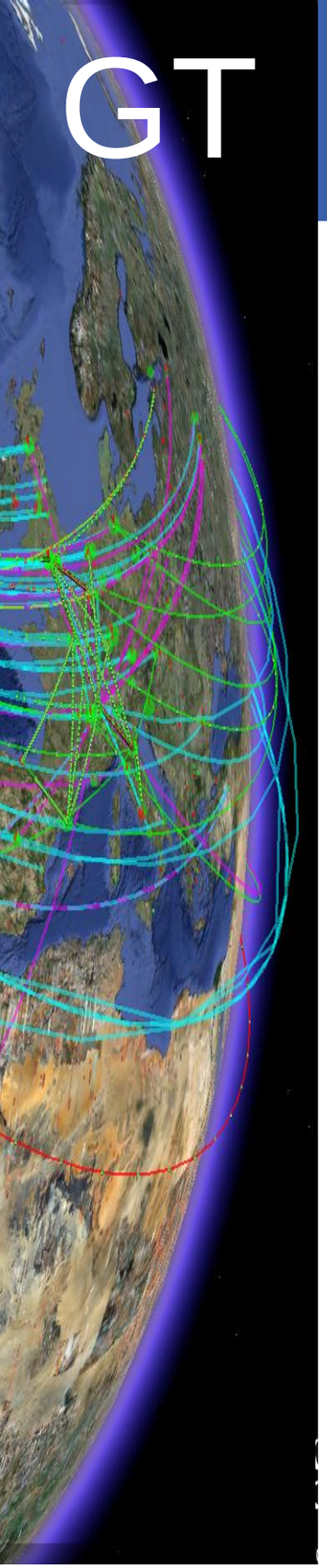


Why HTTP/DAV?

CERN IT Department

- **It's there**, whatever platform we consider
  - A very widely adopted technology
- **We (humans) like browsers**, they give an experience of **simplicity**
- Goes towards **convergence**
  - Users can use their devices to access their data easily, out of the box
  - Jobs just go straight to the data
- With direct access to data, pre-location becomes an optimization choice, not a constraint of the technology

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

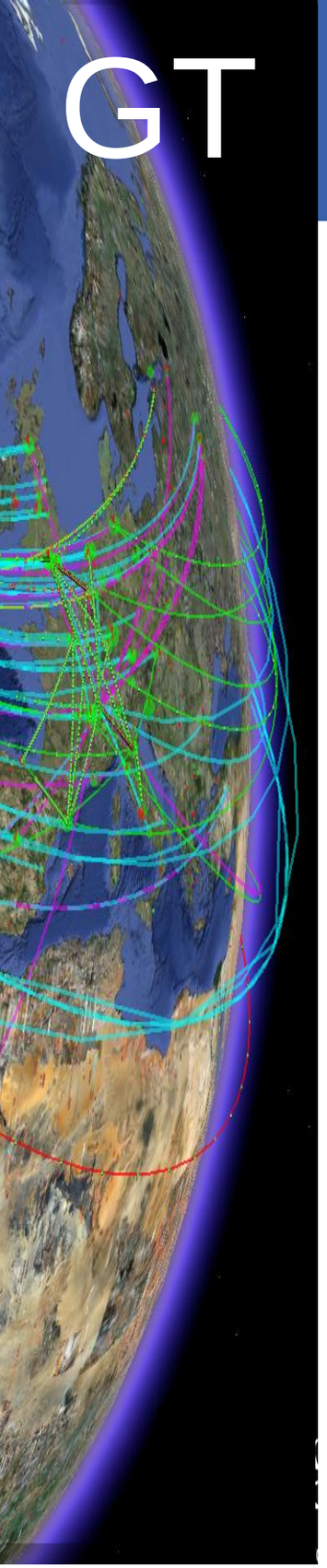


System design

CERN IT Department

- A system that only works is not sufficient
- To be usable, it must privilege speed, parallelism, scalability
- The core component is a plugin-based component called originally “Uniform Generic Redirector” (Ugr)
  - Can plug into an Apache server thanks to the DMLITE and DAV-DMLITE modules (by IT-GT)
  - Composes on the fly the aggregated metadata views by managing parallel tasks of information location
    - Never stacks up latencies!
  - Makes browsable a sparse collection of file/directory metadata
  - Able to redirect clients to replicas of hosts known to be working in that moment
  - By construction, the responses are a data structure that models a partial, volatile namespace
  - Keep them in an LRU fashion and we have a fast 1st level namespace cache
    - Peak performance is ~500K->1M hits/second per core by now

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



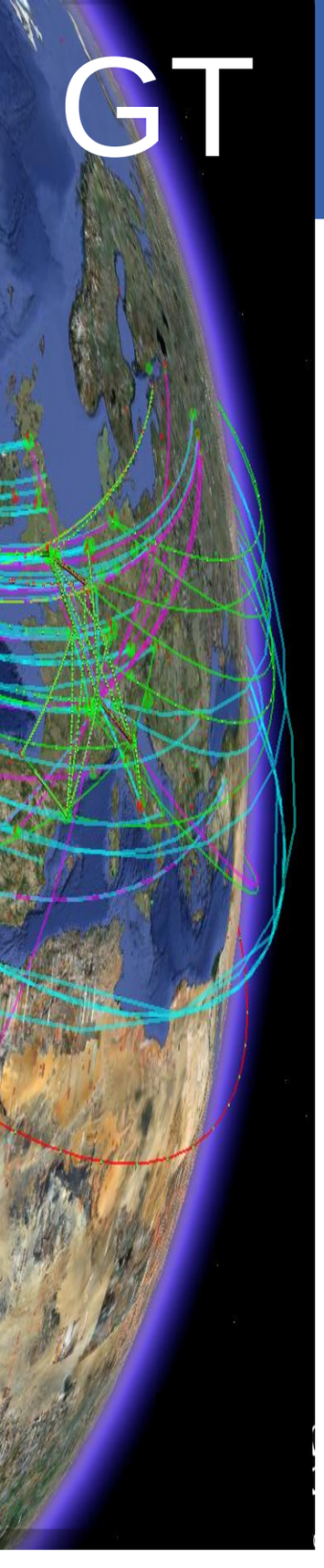
Endpoints

CERN IT Department

- **Pure DAV** endpoints give *metadata of files*
  - Listings, directories, stat, ...
- **Pure HTTP** endpoints give *replicas of files*
  - and no way to know their metadata
- **DAV+HTTP** endpoints give *both*
- Other kinds of endpoints... depends!

- Sysadmins decide the service that they want to provide
- UGR can *quickly* discover metadata and replicas in a population of many endpoints
- UGR can *quickly* discover metadata and replicas even if they are not yet shown in a cached listing

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



Focus: performance

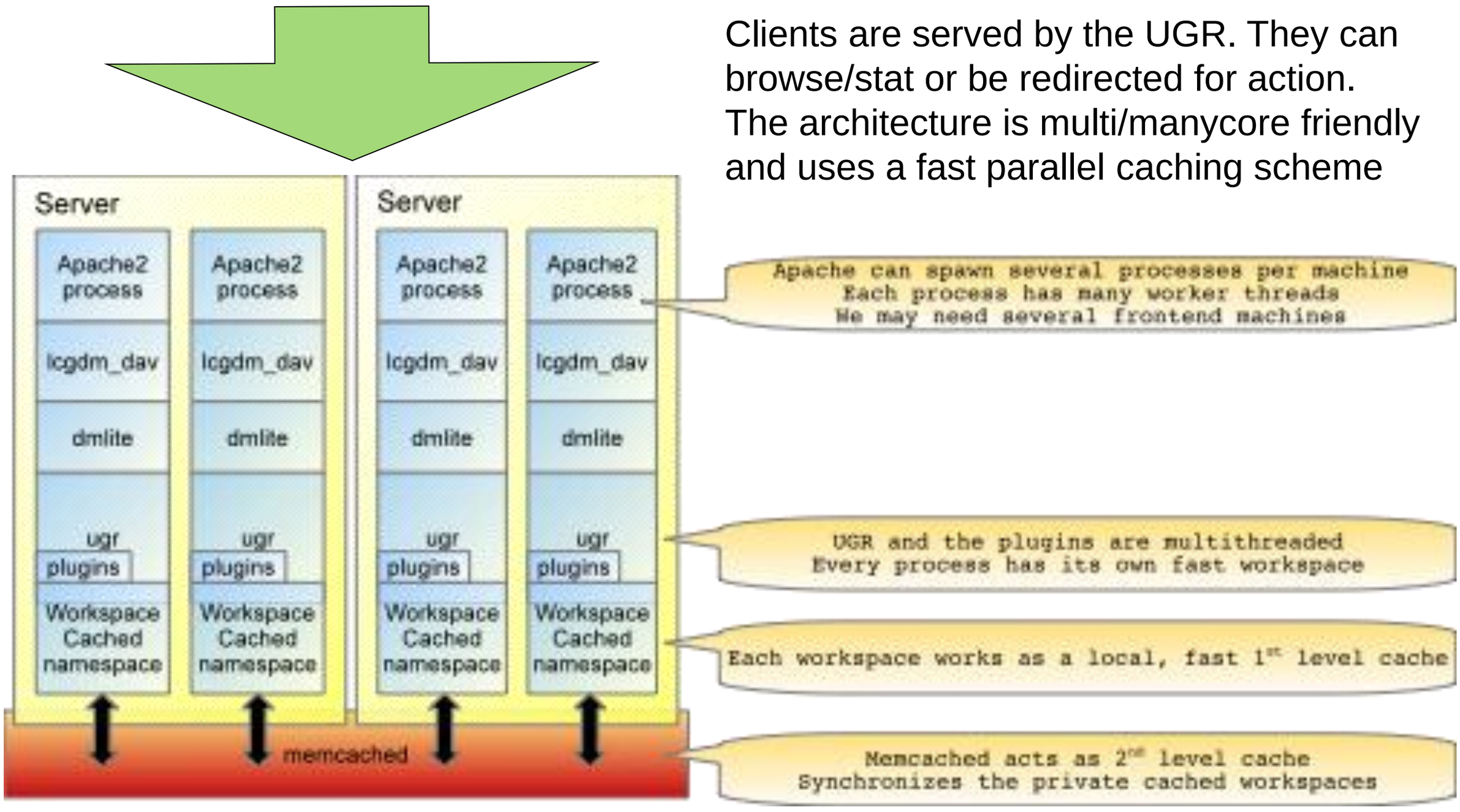
CERN IT Department

- Performance and scalability have primary importance
  - Otherwise it’s useless...
- Full parallelism
  - No limit to the number of outstanding clients/tasks
  - No global locks/serializations!
  - The endpoints are treated in a completely independent way
  - Thread pools, prod/consumer queues used extensively (e.g. to stat N items in M endpoints while X clients wait for some items)
- Aggressive metadata caching
  - A relaxed, hash-based, in-memory partial name space
  - Juggles info in order to always contain what’s needed
  - Stalls clients the minimum time that is necessary to juggle their information bits
  - Peak caching perf per CPU core: 0.5~1M stats/sec
- Spurred a high performance DAV client implementation (DAVIX)
  - Wraps DAV calls into a POSIX-like API, saves from the difficulty of composing requests/responses
  - Loaded by the core as a “location” plugin
  - Performance is privileged: uses libneon w/ sessions caching
  - Compound list/stat operations are supported

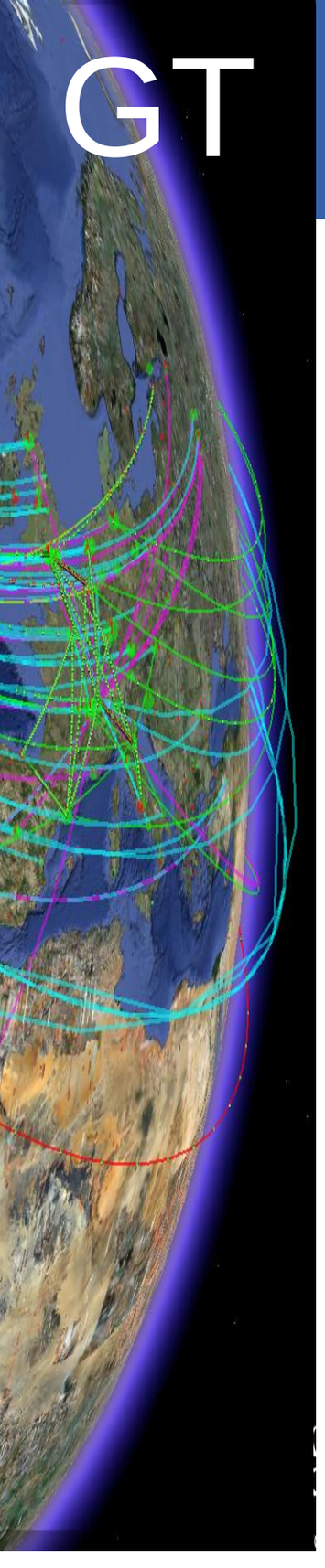
CERN IT Department
CH-1211 Geneva 23
Switzerland
www.cern.ch/it

Concurrent metadata caching

Clients come and are distributed through:
- different machines (DNS alias)
- different processes (Apache config)



Clients are served by the UGR. They can browse/stat or be redirected for action. The architecture is multi/manycore friendly and uses a fast parallel caching scheme



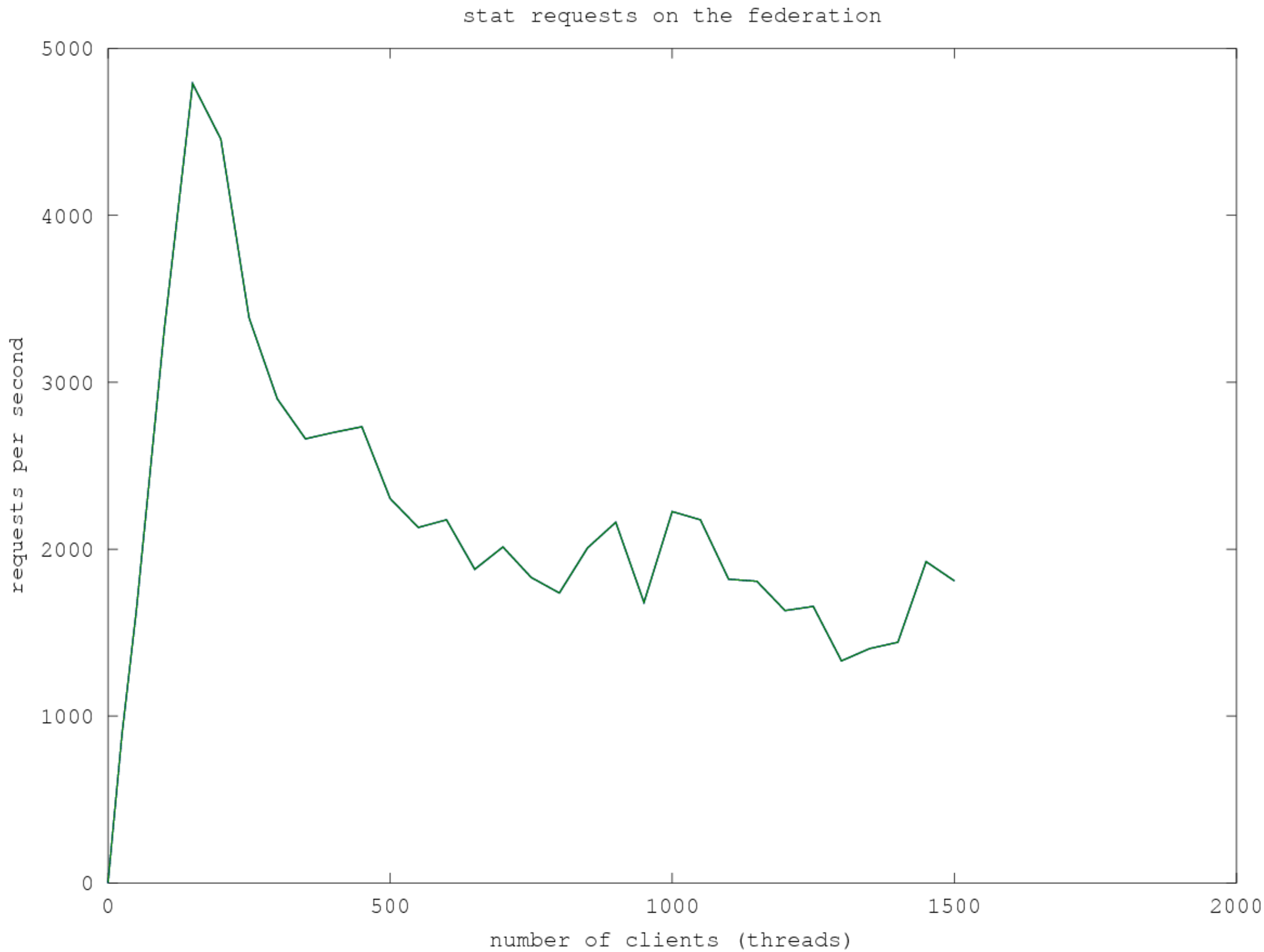
A performance test

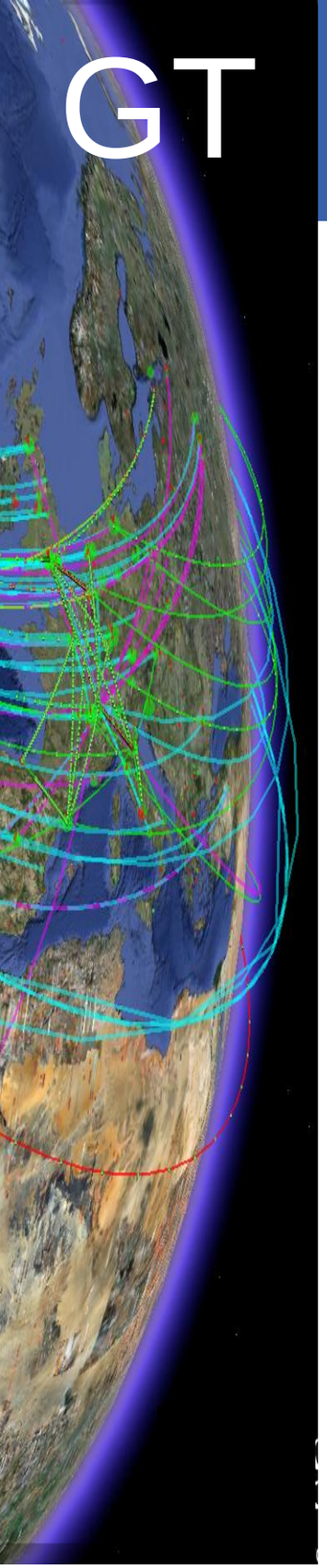
CERN IT Department

- Measuring the worst case performance becomes difficult
  - The cache caches also the intermediate actions, e.g. the directory lookups in a path
- Two endpoints: DESY and CERN (poor VM)
- One UGR in DESY (federation.desy.de)
- 10K files are interleaved in a 4-levels deep directory
  - Oddly-numbered files are at CERN
  - Evenly-numbered files are at Desy
- The test (written in C++) invokes Stat only once per file, using 100 to 1500 parallel clients doing stat() at the maximum pace from 3 machines
  - The clients run at CERN
  - Hence it's supposed to be a worst-case, measuring the full roundtrip. In practice the cache has a role anyway, as the files share the path.
  - The crude speed of 2K stats/s per frontend machine is satisfying. Seems somehow capped by Apache, as the backend UGR is up to 100 times faster

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

WAN access (CERN-DESY)

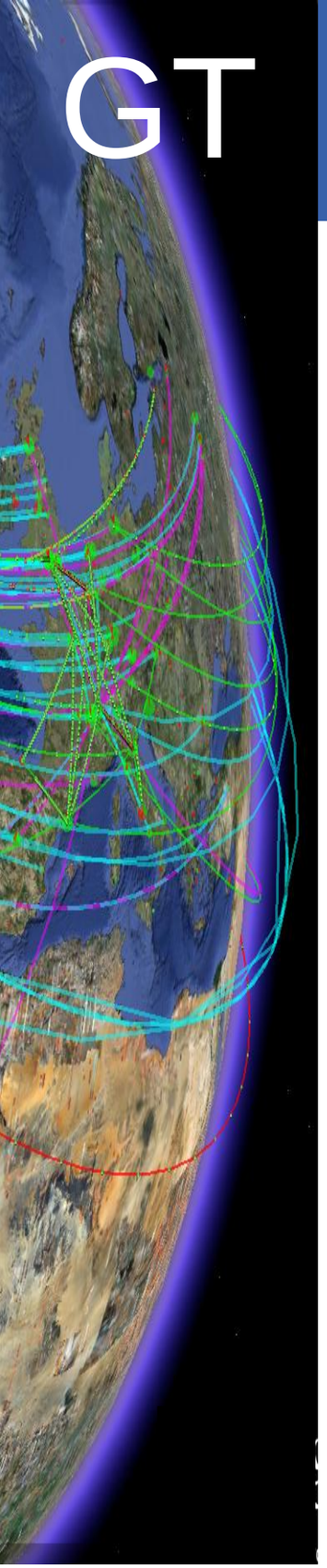




Next steps

- Release our beta, as the nightlies are good
- Improve the Wiki page, consider a website
- More massive tests, with many endpoints, possibly distant
- Immediate sensing of changes in the endpoints' content, e.g. add, delete
  - SEMsg in EMI2 SYNCAT would be the right thing in the right place
- Some more practical experience (getting used to the idea, using SQUIDs, CVMFS, EOS, clouds,... <put your item here>*)
- Power users helping in getting the best out of the system
  - Cooperation is very appreciated

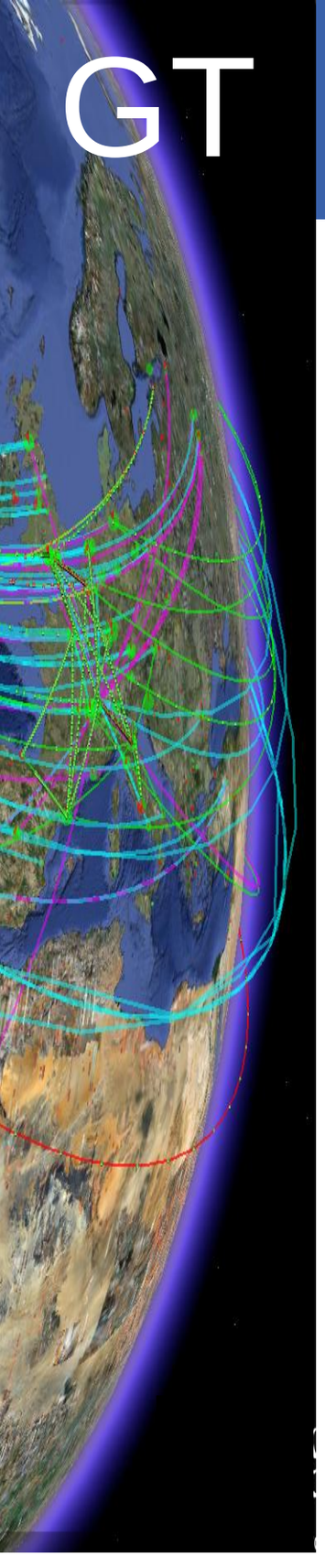
CERN IT Department
CH-1211 Geneva 23
Switzerland
www.cern.ch/it



Get started

- Get it here:
  - https://svnweb.cern.ch/trac/lcgdm/wiki/Dynafeds
- What you can do with it:
  - Easy, direct job/user data access, WAN friendly
  - Friend sites can share storage
  - Storage-less sites, storage-only sites
  - Federating catalogues
    - Combining catalogue-based and catalogue-free data

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



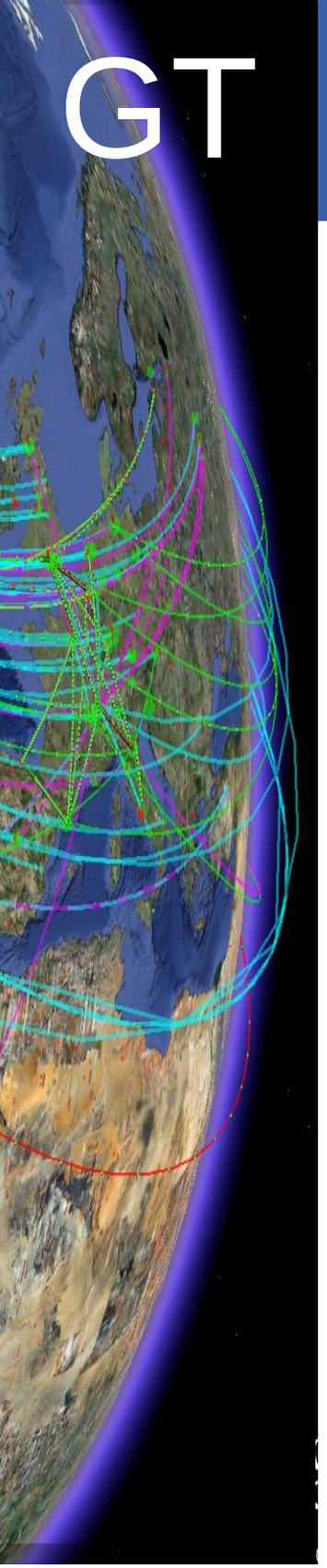
Xrootd and HTTP

CERN IT Department

- Many instances of xrootd, not only HEP
  - Generally high quality deployments of a high quality framework
  - Sometimes very important ones (e.g. CERN with EOS)
  - Advanced features: tapes, data movements, monitoring, etc.
- Many instances of http/dav compliant SEs
- Once a site/organization has chosen a framework, changing it can be very expensive
  - Also quality of service is at risk
- Risk is having islands of protocols within the same community
- Risk is having to write glue code in the client applications, to wrap the two protocols

- DPM can join an Xrootd federation and an HTTP one as well
- dCache too, STORM has GPFS behind
- How to accommodate Xrootd-based SEs in an HTTP environment?

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

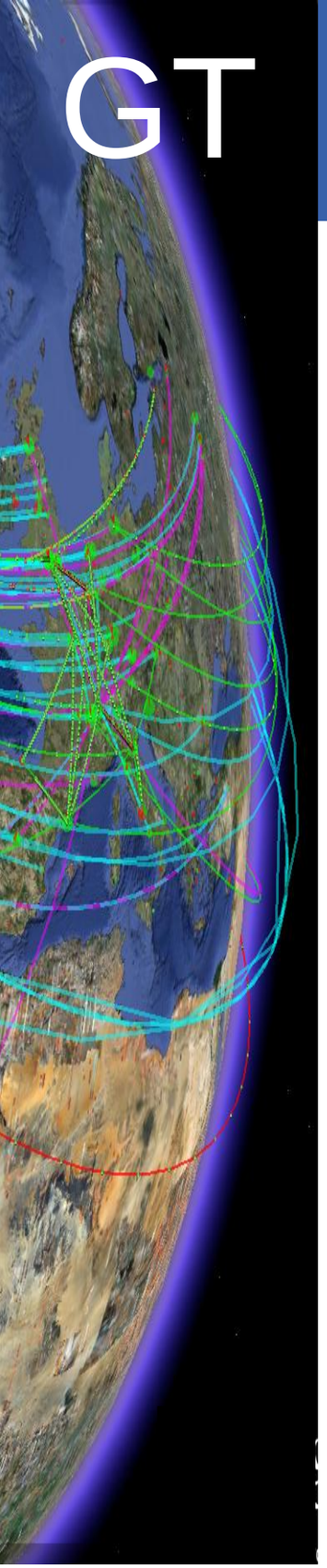


An XrdHTTP plugin

CERN IT Department

- ...writing an HTTP plugin for the Xrootd framework!
- Evaluation work started, to evaluate the needed effort:
  - support basic HTTP[S] and DAV with the DM extensions (replicas, etc.)
  - plug into the framework to use its features (e.g. tapes or monitoring) without reconfiguring the site
- A newer version of the Xrootd framework will drastically reduce the need for code duplication and the effort
- Work in progress...
- Goal is getting the best from each system or framework and respect the choices

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

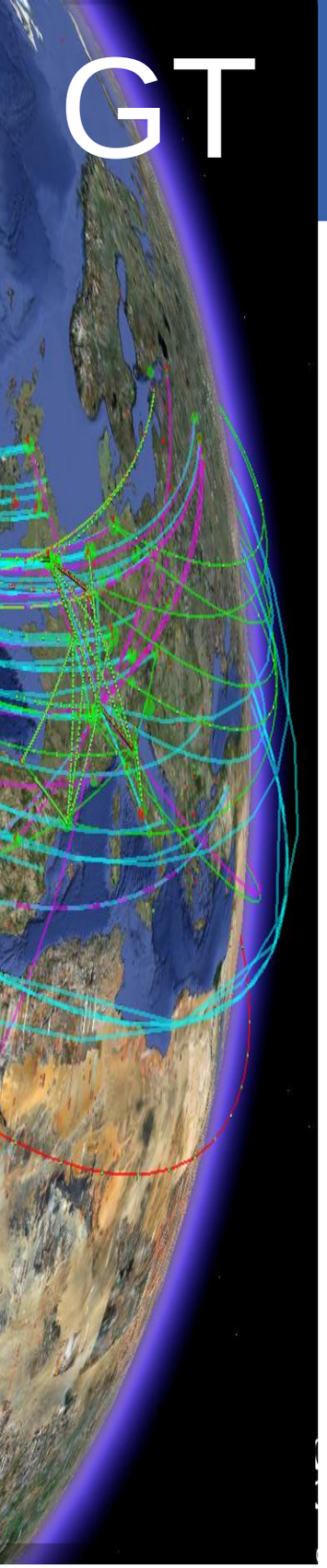


Conclusions

CERN IT Department

- Dynamic Federations: an efficient, persistency-free, easily manageable approach to federate remote storage and metadata endpoints
- Usable for fast changing caches and clouds
- Gives ways to solve some nasty Data Management problems
- Work in progress, status is very advanced, demoable, installable, documented.
- There will be an Application Area seminar on the 27th, about the whole HTTP story... do not miss it!
  - https://indico.cern.ch/conferenceDisplay.py?confId=218328

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



Thank you

CERN IT Department

- Questions?

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