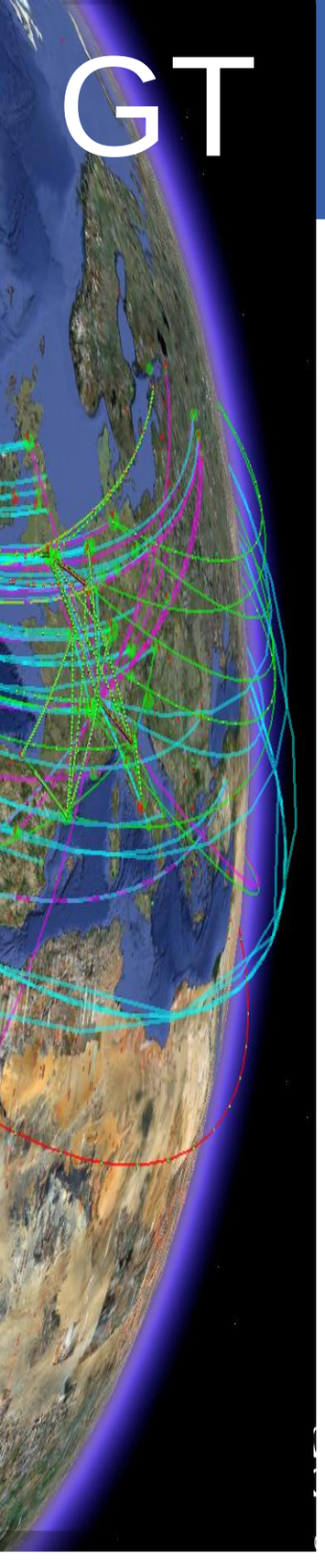


CERN IT Department

Dynamic Federations

A demo with dCache, DPM and a cloud storage provider

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

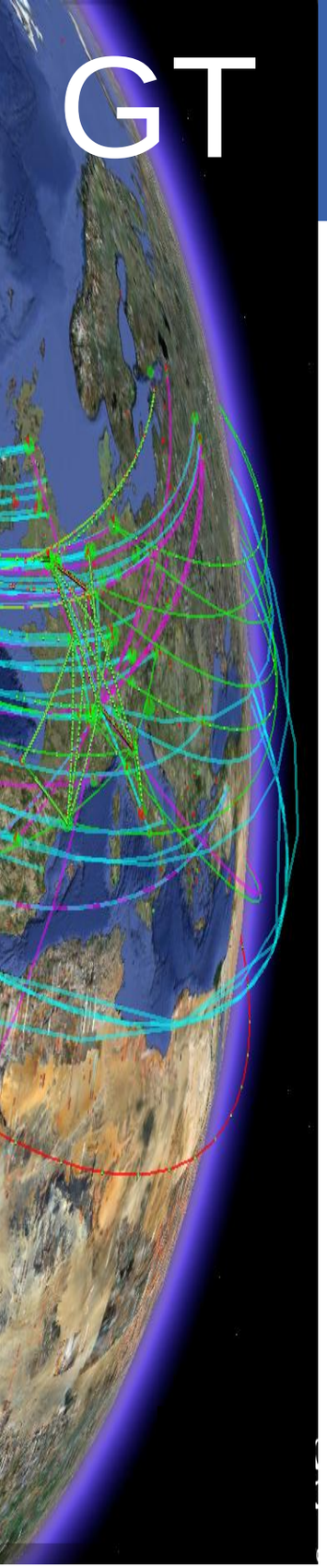
Credits to ShuTing Liao (ASGC)

Partially funded by EUROPEAN MIDDLEWARE INITIATIVE *and* dCache

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

EMI is partially funded by the European Commission under Grant Agreement INFSO-RI-261611

CERN

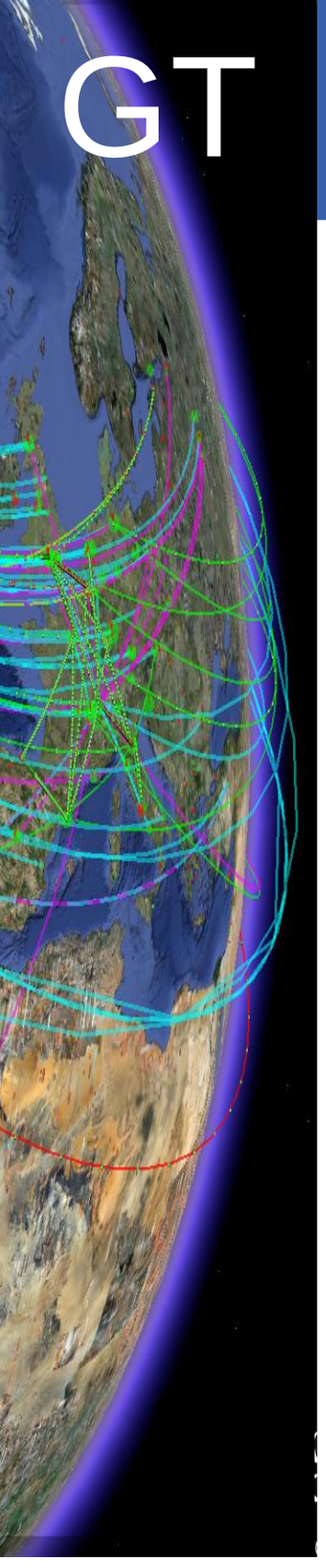


Storage Federations: Motivations

CERN IT Department

- Currently data lives on islands of storage
  - catalogues are the maps
  - FTS/gridFTP are the delivery companies
  - Experiment frameworks populate the island
  - Jobs are directed to places where the needed data is
    - or should be .....
  - Almost all data lives on more than one island
  - Assumption:
    - perfect storage (unlikely to impossible)
    - perfect experiment workflow and catalogues (unlikely)
- Strict locality has some limitations
  - a single missing file can derail the whole job
    - or series of jobs
- Failover to data on another island could help
    - → Federation

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

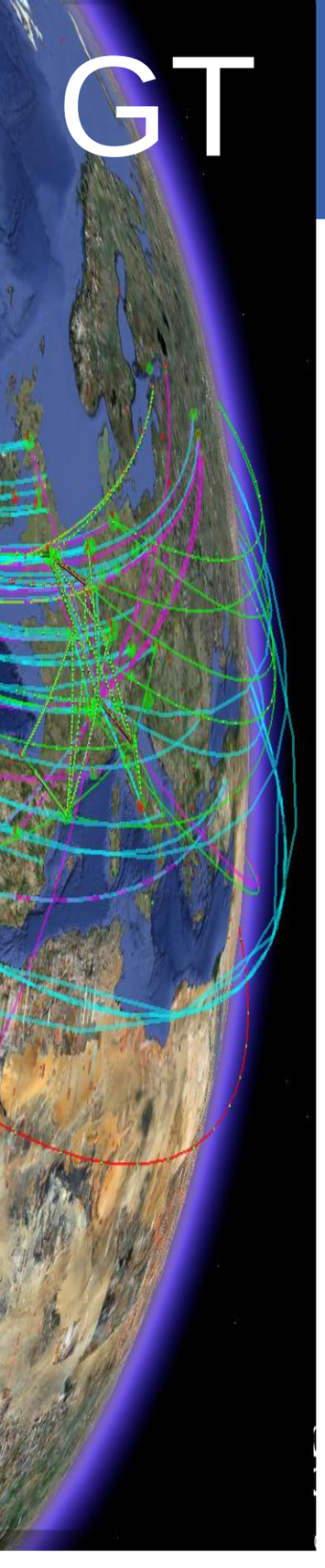


Storage federations

CERN IT Department

- What has to be done?
  - 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”
  - Use dynamic systems that are **easy to setup/maintain:**
    - no complex metadata persistency
    - no DB babysitting (keep it for the experiment’s metadata)
    - no central catalogue inconsistencies, by design
  - Use systems that exhibit **high performance**
    - Goal is hundreds of Ks interactions per second (peak)
  - Try to optimize by using proximity etc.

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

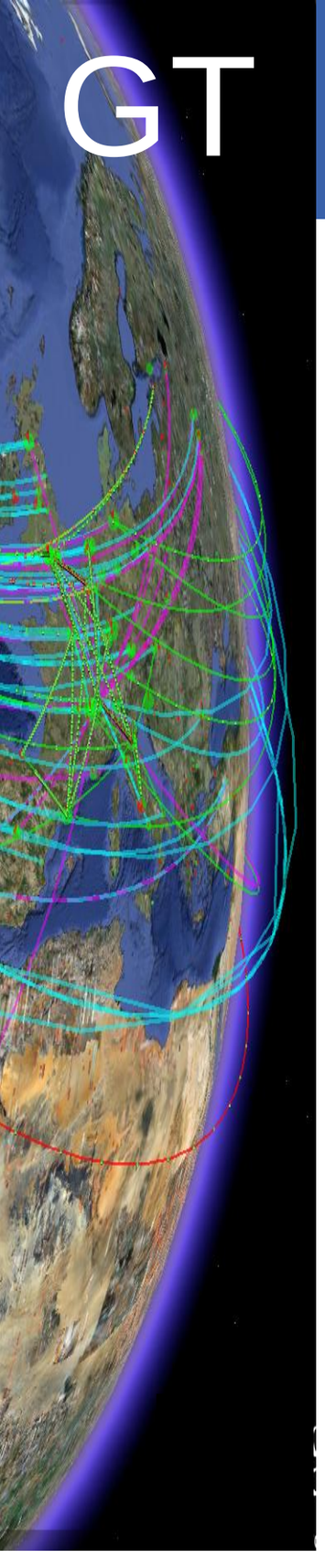


Vision: Everything Everywhere

CERN IT Department

- Transparent access to remote data
  - should feel natural
  - This includes interactive browsing of files and directories
- Seamless storage federations of:
  - Official Storage Elements, LFCs, catalogues...
  - Cached data (i.e. SQUID-like things, not registered in any catalogue)
  - HTTP/DAV-based servers
  - Cloud storage services
  - HTTP-enabled XROOTD/EOS clusters, sharing the data.
- Base everything on open ‘just works’ technologies
- 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.
- Limit complexity: read only
  - Usually writes happen to well-known, close islands

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

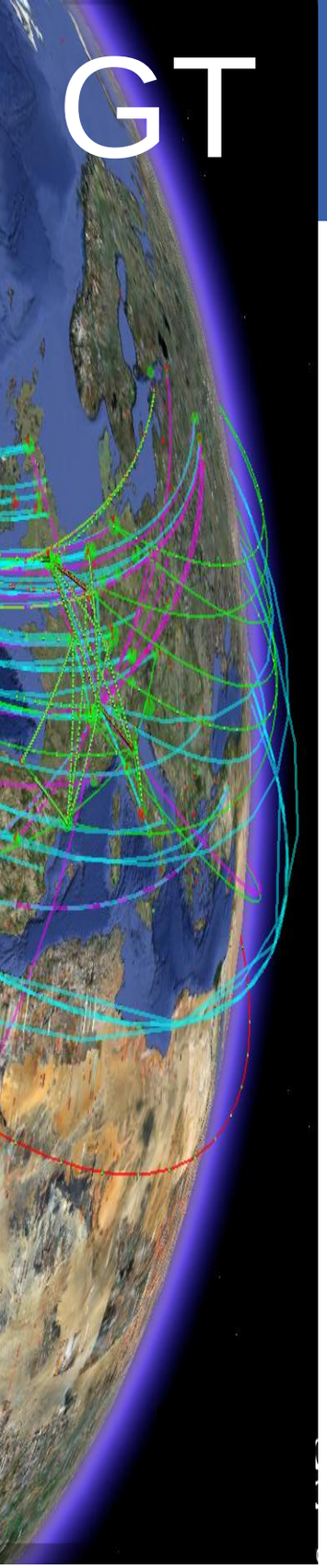


Dynamic Federations

CERN IT Department

- We federate (meta)data repositories that are ‘compatible’
  - Name space (modulo simple prefixes)
  - Permissions (they don’t contradict across sites)
  - Content (same key or filename means same file)
- Dynamically and transparently discovering metadata
  - looks like a unique, very fast file metadata system
  - properly presenting the aggregated metadata views
  - redirecting clients to the geographically closest endpoint
- As a plus: can be used by client tools that **everybody knows**
  - **focus on HTTP/DAV**, we can use
  - it by ROOT via Browswer, OS’es to smartphones

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

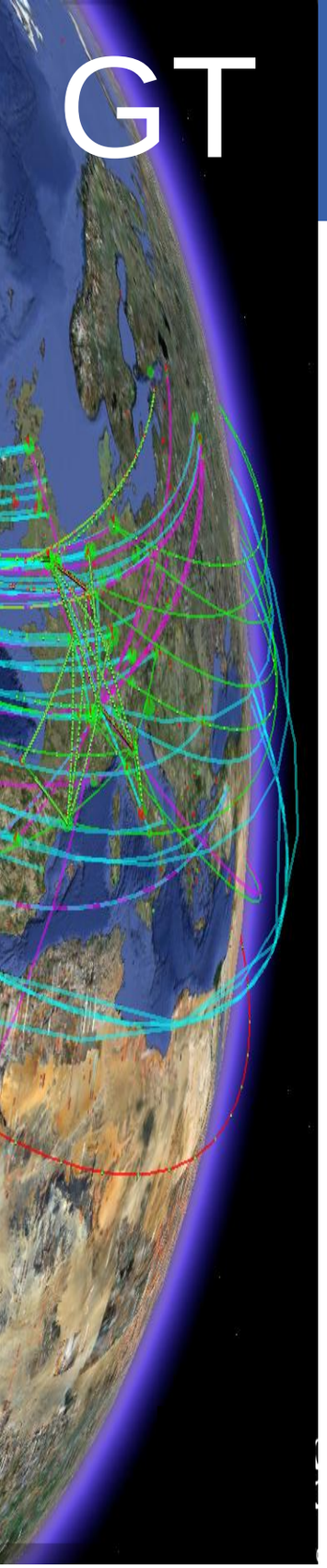


Dynamic Federations

CERN IT Department

- 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 (through DMLite used as a client)
  - **Can be extended to other 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 one that’s available uses GeoIP (free)

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



Demo

- The demo federates three storages via HTTP/DAV:
  - a DPM instance at ASGC (Taiwan)
  - a dCache instance in DESY
  - a Cloud storage account by Deutsche Telecom

- The feeling it gives is surprising
  - performance is in avg higher than contacting the endpoints

- I put one test file in 3 sites, i.e. 3 replicas.
  - /myfed/atlas/fabrizio/hand-shake.JPG
  - **Clients in EU get the one from DESY/DT**
  - **Clients in Asia get the one from ASGC**

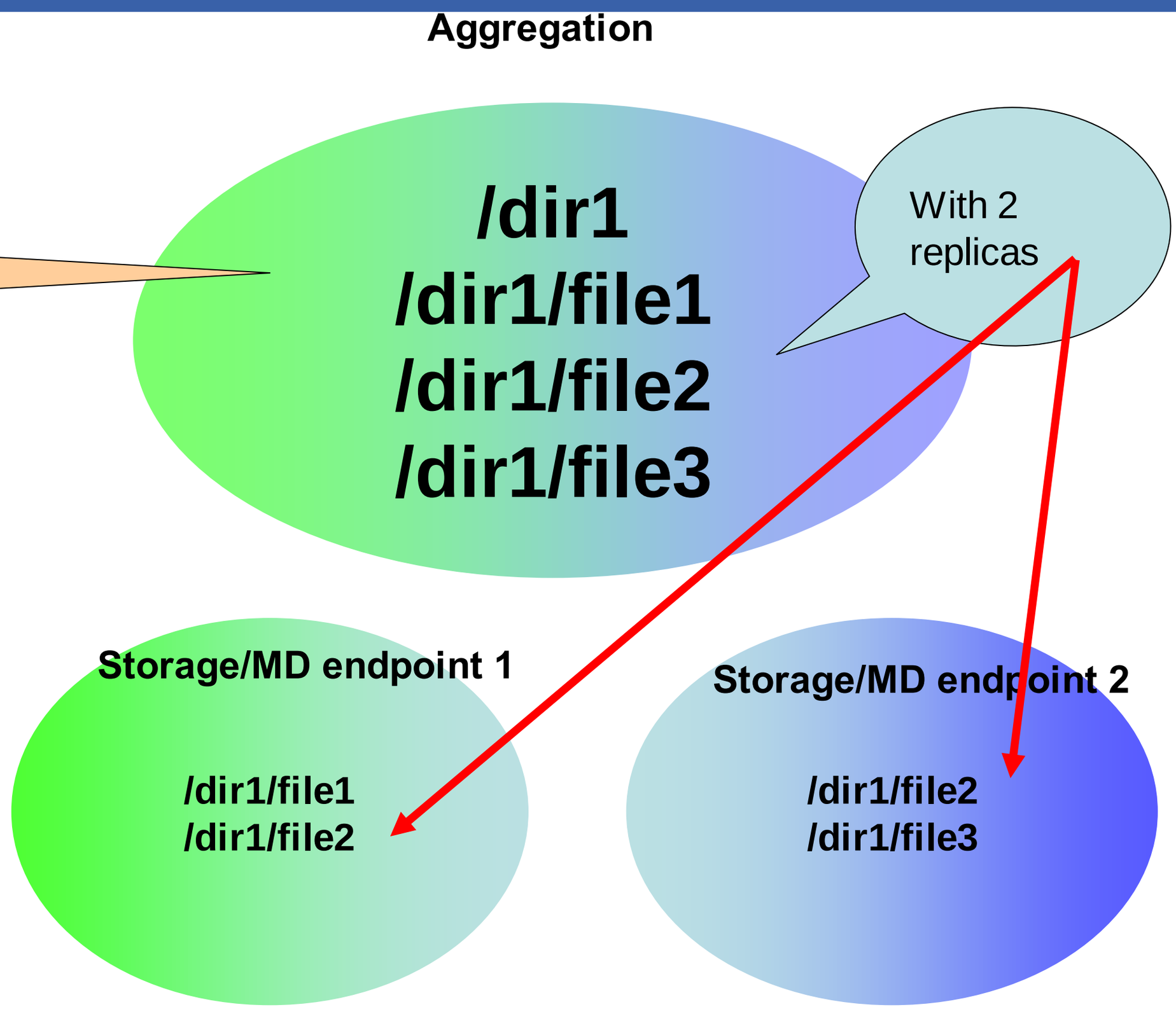
CERN IT Department
CH-1211 Geneva 23
Switzerland
www.cern.ch/it

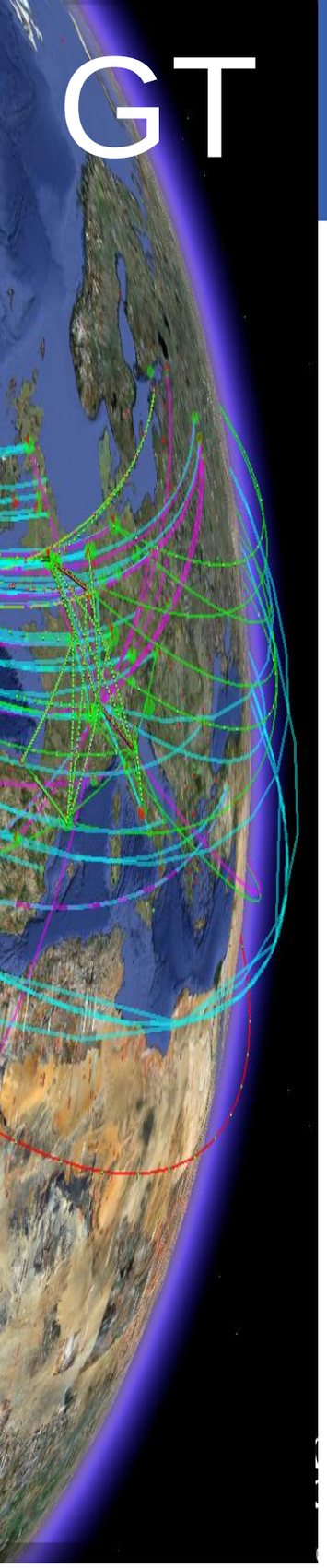
The basic idea

We see this

All the metadata interactions are hidden

NO persistency needed here, just efficiency and parallelism





Dynamic Federations

CERN IT Department

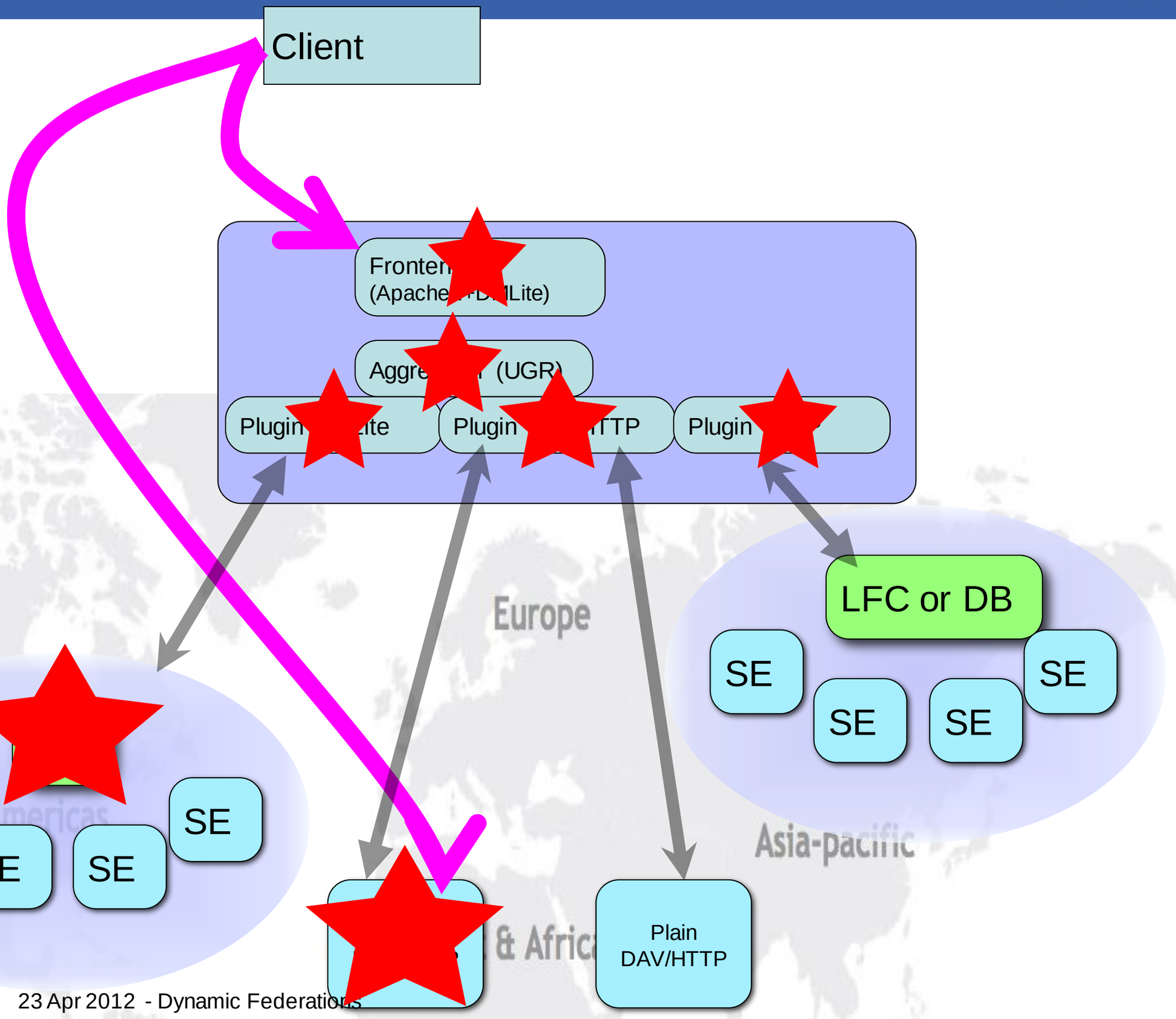
- Technically “loosely coupled storage systems”
- Idea: a single entry point for a federation of ***endpoints***
  - “lonely” storage clusters (e.g. dCache, DPM, plain HTTP servers)
  - site/VO catalogues (e.g. LFCs) pointing to storage elements
- This entry point knows its ***endpoints***, can ***redirect*** clients to them, it can present their metadata to clients
- Many interesting possibilities
  - Federate third party outsourced HTTP/DAV servers (also clouds)
  - Federate the content of SQUID caches
  - Federate them together with the information of some experiment’s DB
  - When requesting a file we would get it from an ***endpoint*** that is good for us, it could be a cache or a non-cache one.
  - See as one 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

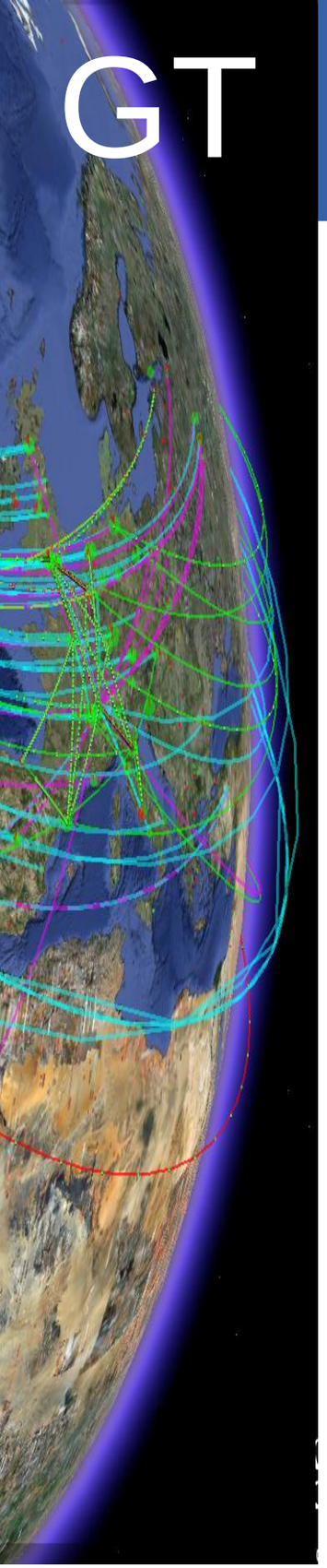
CERN IT Department
CH-1211 Geneva 23
Switzerland
www.cern.ch/it

Dynamic Federations

- The ***endpoints*** are a federation, hence they are homogeneous
  - Same access protocol (e.g. HTTP/DAV)
  - Same name space (not necessarily same content!)
  - The same file/replica has the same (or compatible) path/name
  - They grant access to the same groups of users
- This entry point learns dynamically, automatically about their metadata content
  - As clients contact it to get access to files
  - It can ask the ***endpoints*** for information on the fly
- This entry point redirects each client to the proper ***endpoint***
  - Ev. applying some smart criteria, e.g. geographical proximity
- In principle it would work for any data access protocol that
  - works over WAN
  - supports redirections
- The system core is fully protocol-agnostic
- Our focus is towards HTTP/DAV
  - DPM and dCache are releasing support for it (StoRM is on it's way)
- Work in progress, priority is read access
  - As, in general, write access is done in the local site

Example


Client
Plugin
Plugin
Plugin
LFC or DB
SE
SE
SE
SE
SE
SE
SE
Plain
DAV/HTTP
Europe
Asia-pacific

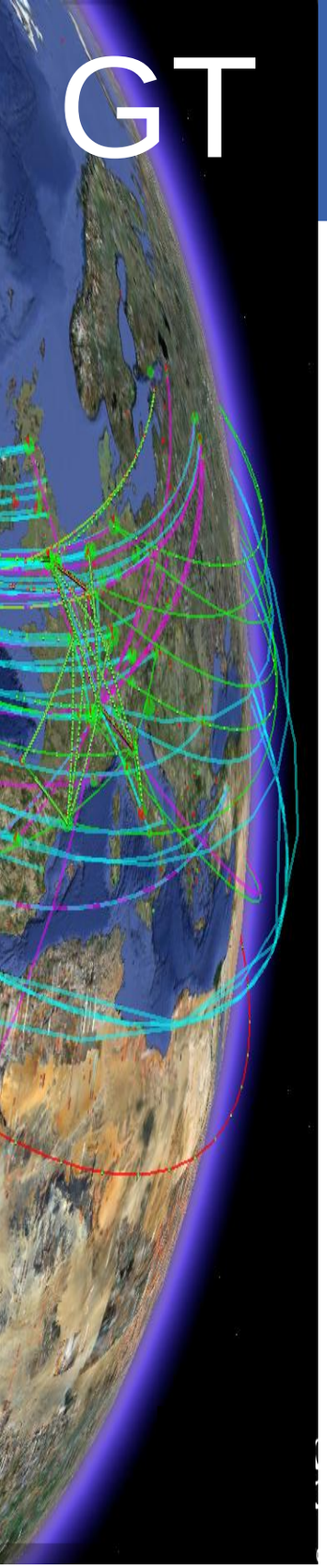


Why HTTP/DAV?

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

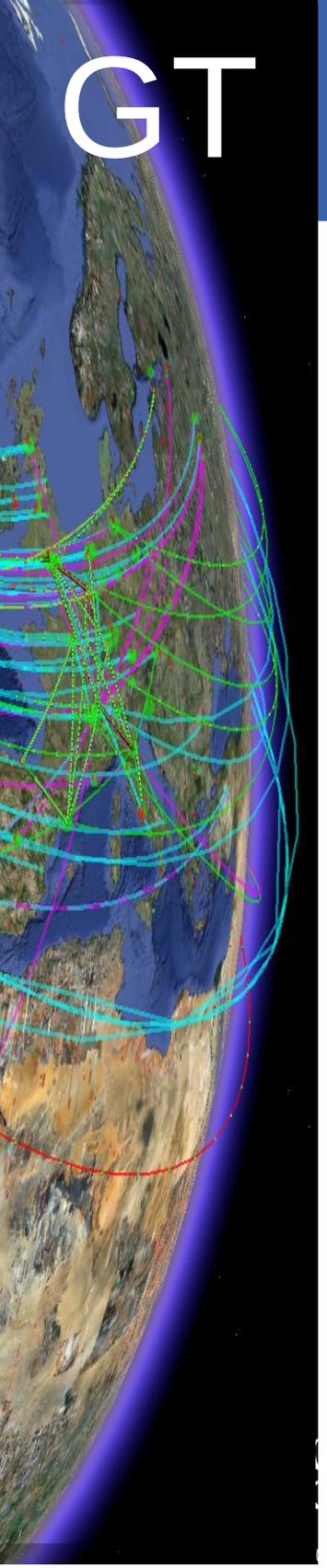
CERN IT Department
CH-1211 Geneva 23
Switzerland
www.cern.ch/it



System design

- 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!
  - Able to redirect clients to replicas
  - 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

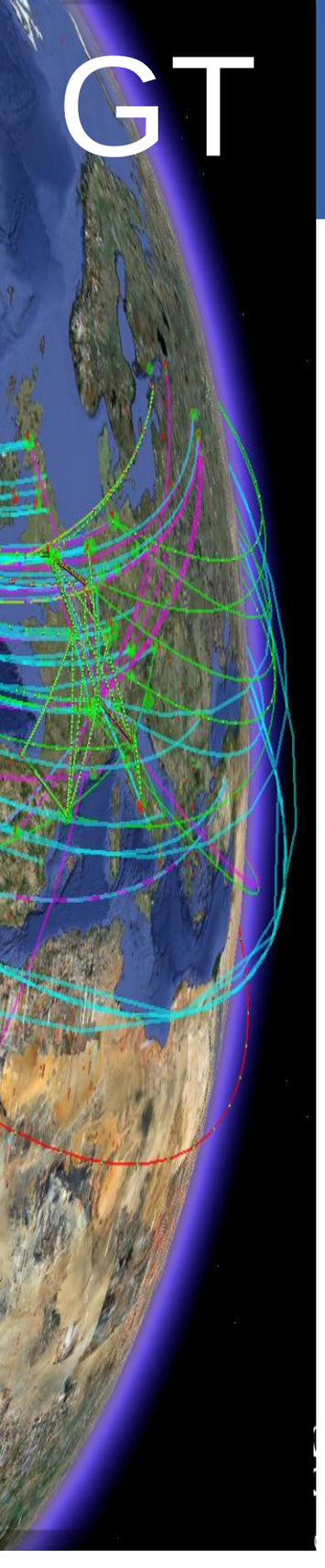


Focus: performance

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 perf per CPU core: 0.5~1M stats/sec
- High performance DAV client implementation (DAVIX)
  - Loaded by the core as a "location" plugin
  - Uses libneon w/ sessions caching
  - Compound list/stat operations

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



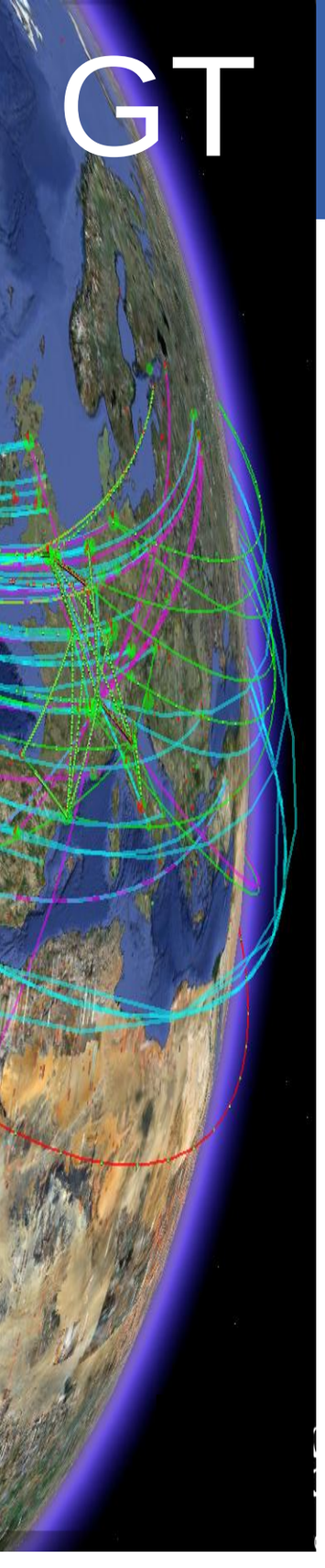
Dynamic Feds vs XROOTD feds

CERN IT Department

- XROOTD federations are focused on the “redirection” concept
  - Very light at the meta-manager, just redirect clients away as soon as possible
    - If not possible, the penalty is 5 seconds per jump
  - Global listing is implemented in the client, slow, a bit more difficult, less robust
  - Some details do not match yet very well with geography-aware redirections

- Dynamic Federations support both the “redirection” concept and the “browsing” concept by design
  - Cache metadata for the clients, in-memory
  - Designed for scalability, performance and features
  - Extendable plugin architecture, geography-aware redirection
  - Can speak any protocol, our focus in on **http**-based things

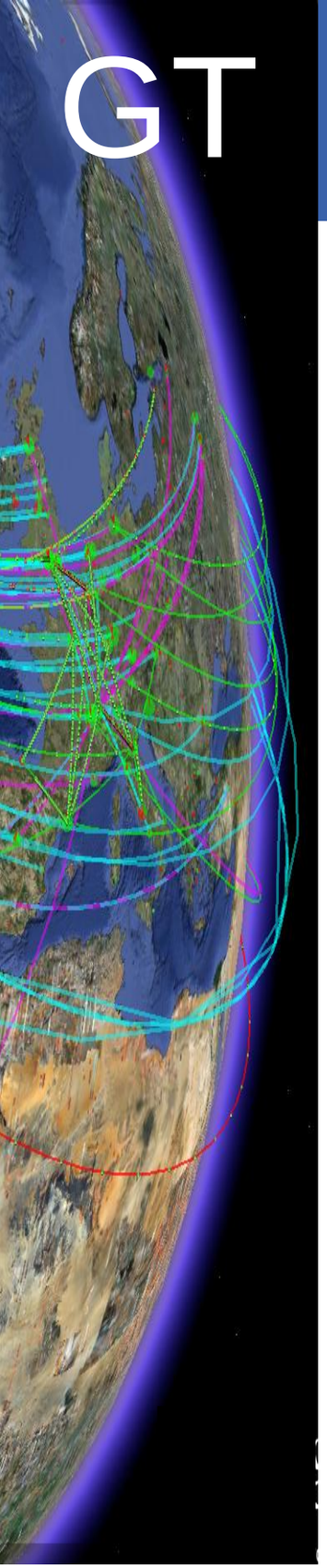
CERN IT Department
CH-1211 Geneva 23
Switzerland
www.cern.ch/it



Next steps

- Implement the missing bits of security, without hurting scalability and performance.
- More massive tests, with many endpoints, possibly distant
- Precise performance measurements
- Handle gracefully the ‘death’ of the endpoints
- Understanding the impact of the production workloads
  - Does the machinery need a 2nd level of caching? Which characteristics?
- Immediate sensing of changes in the endpoints’ content, e.g. add, delete
  - SEMsg in EMI2 already has some useful notifications (default off)
- 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

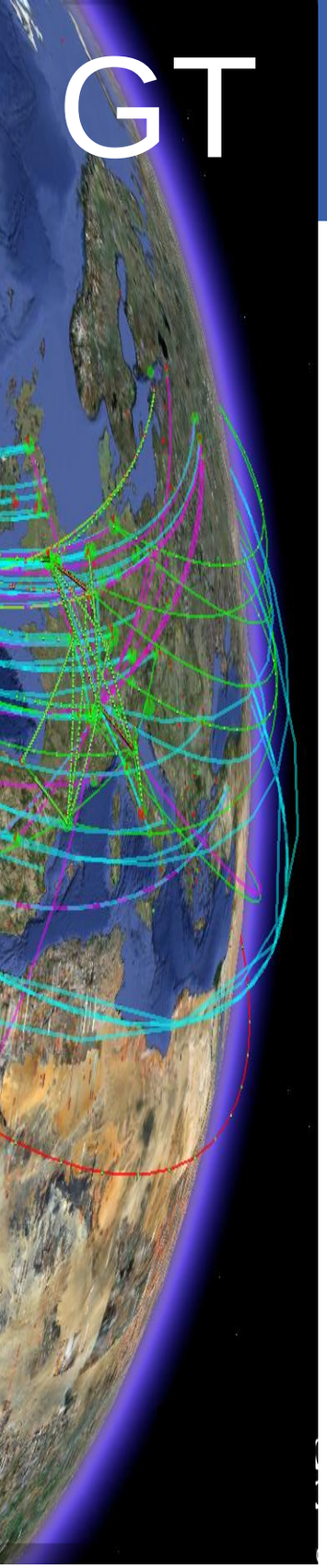
CERN IT Department
CH-1211 Geneva 23
Switzerland
www.cern.ch/it



Conclusions

- Dynamic Federations: an efficient, persistency-free, easily manageable approach to federate remote storage endpoints
- Usable for fast changing caches and clouds
- Gives ways to solve some nasty Data Management problems
- Peak performance is very high: $O(10^5)$ hits/s
- Opens our Data world to a large variety of already available clients, by using standard protocols.
- Work in progress, first milestone was now.

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



Thank you

Questions?

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

Partially funded by EUROPEAN MIDDLEWARE INITIATIVE

EMI is partially funded by the European Commission under Grant Agreement INFSO-RI-261611