Dynamic Storage Federation based on open protocols

Adrien Devresse (presenter)
Fabrizio Furano
Patrick Fuhrmann
Paul Millar
Daniel Becker

Oliver Keeble
Ricardo Brito da Rocha
Alejandro Alvarez

Credits to ShuTing Liao (ASGC)

CERN IT Department
CH-1211 Genève 23
Switzerland

Motivations for Federations (1)

Current situation:

→ Data are in groups of distributed of Storage Systems at differents places

→ The location of data is managed by
- → Experiment framework
- → Meta-data catalogs

→ Jobs are (supposed to be) placed close to the data

Motivations for Federations (2)

CERN IT Department

Can be improved !

→ **What happens to your workflow if**

- → A storage system is offline ?
- → A file is missing ?
- → The meta-data catalog is overloaded ?

→ **Listing and/or browsing are expensive/slow operations**

→ **The file access pattern is clearly not optimal**

Dynamic Federation Goal ?

Simple Idea :

Federate on the fly

Grid Distributed Storage,

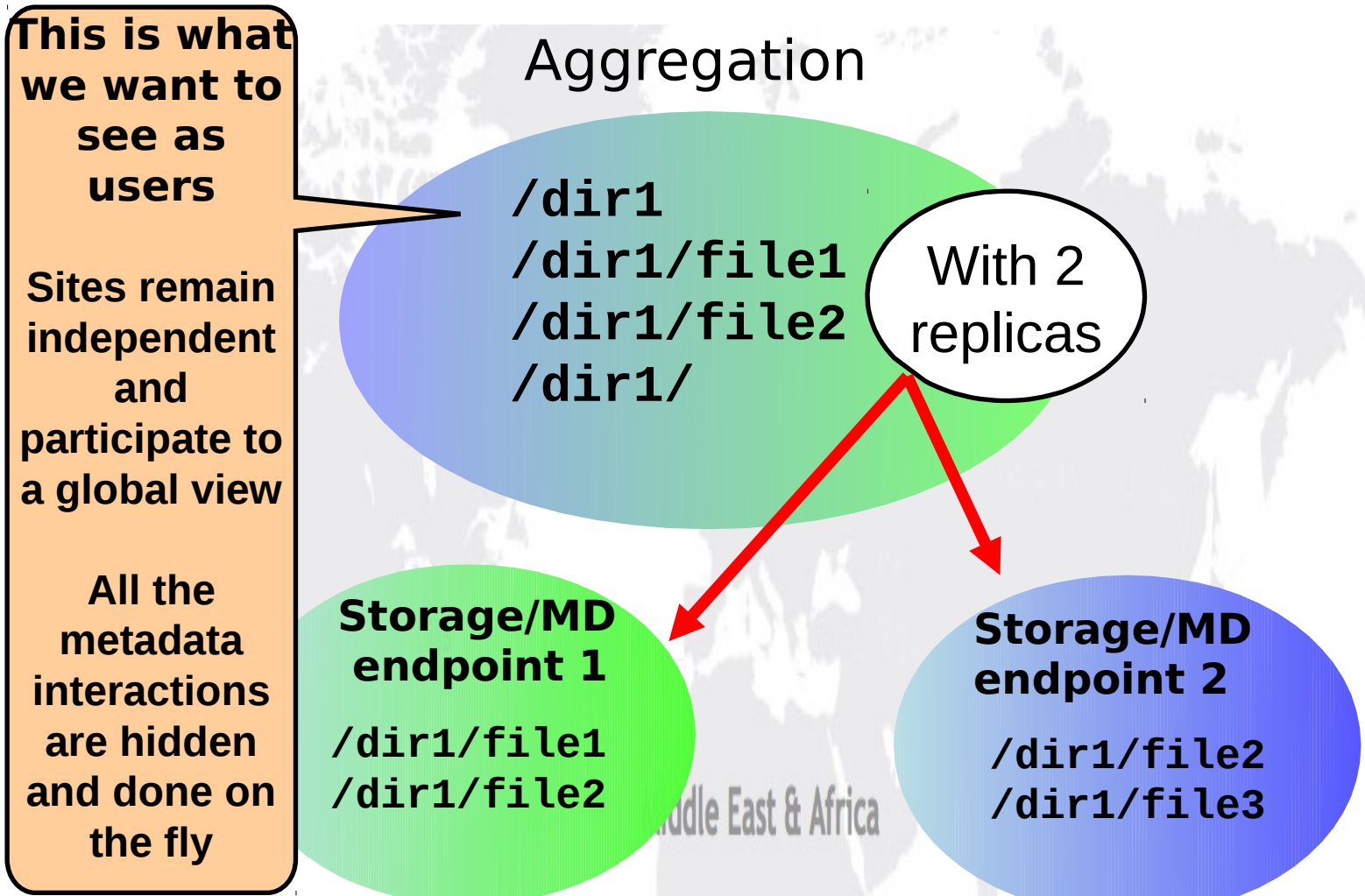
Cloud Storage,

Any Existing meta-data Catalog,

In a unique namespace

Basic Idea of the federation

CERN IT Department



New possibilities

CERN IT Department

Open a lot of new possibilities !

- → **Reliability, failover**
  - → Detect offline storage element
  - → Detect network problems
  - → World wide replica discovery
- → **Smart Redirection**
  - → Geographical redirection
  - → Network optimizations
  - → Transparent caching

New possibilities (2)

CERN IT Department

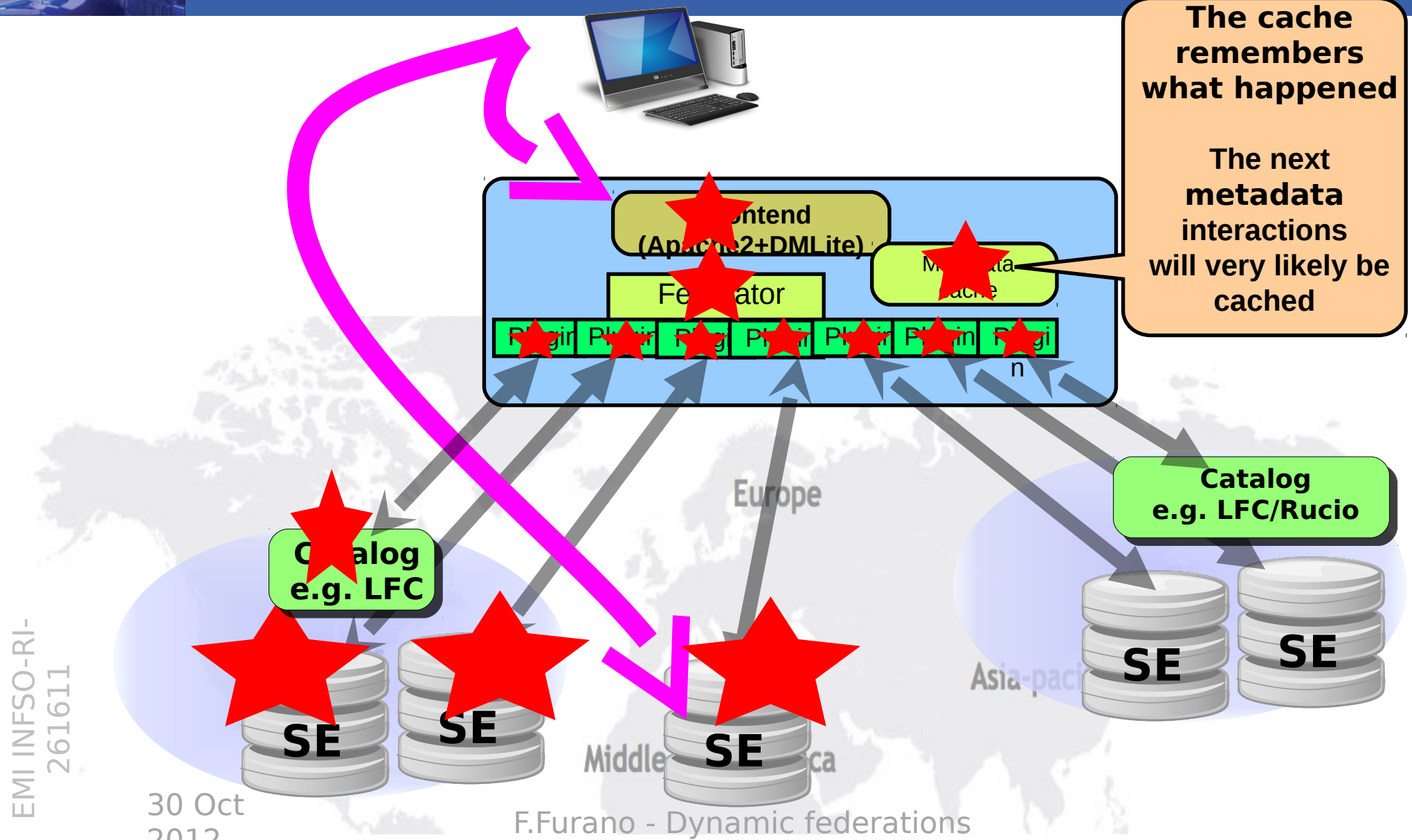
- **Easy data Migration**
  - Transparent data migration
  - Merge Grid and Cloud Storage namespace
- **Scalability**
  - Aggressive caching
  - Protection of your name servers
  - Allow horizontal deployement
  - Load balancing

New possibilities (3)

CERN IT Department

- **Performance**
  - Multi-stream download (metalink)
  - Reduce latency (GeoIP, local Fed)
  - Low response time (in memory)
- **Flexibility**
  - Algorithmic filename translation
  - Merge any type of Storage (Grid, Cloud)
  - Use / integrate your own catalog
  - Allow local federations

How does it work ?



How is it working ?

CERN IT Department

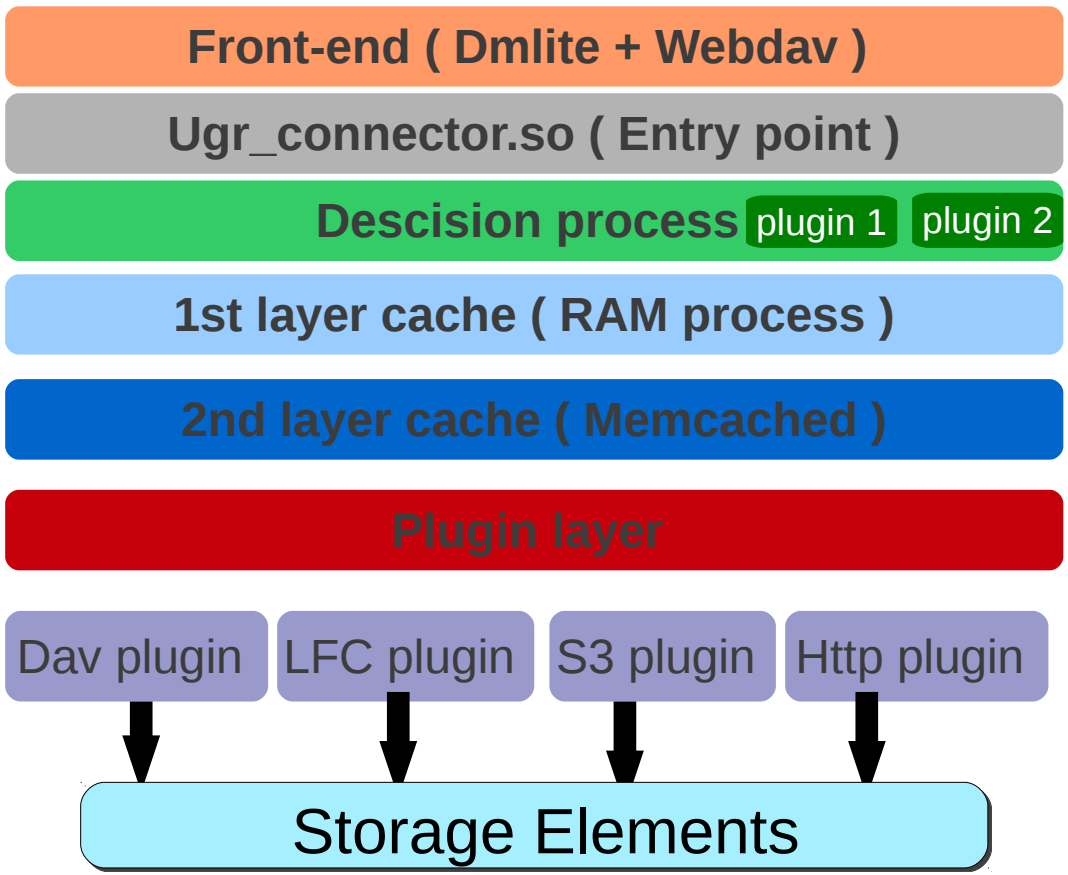
→ **No Database**

→ **Fetch the meta-data on demand from the Storage endpoints (SE)**

→ **Aggregate the meta-data on the fly**

→ **Distributed System**

Dynamic Federation Architecture



→ In Memory Only

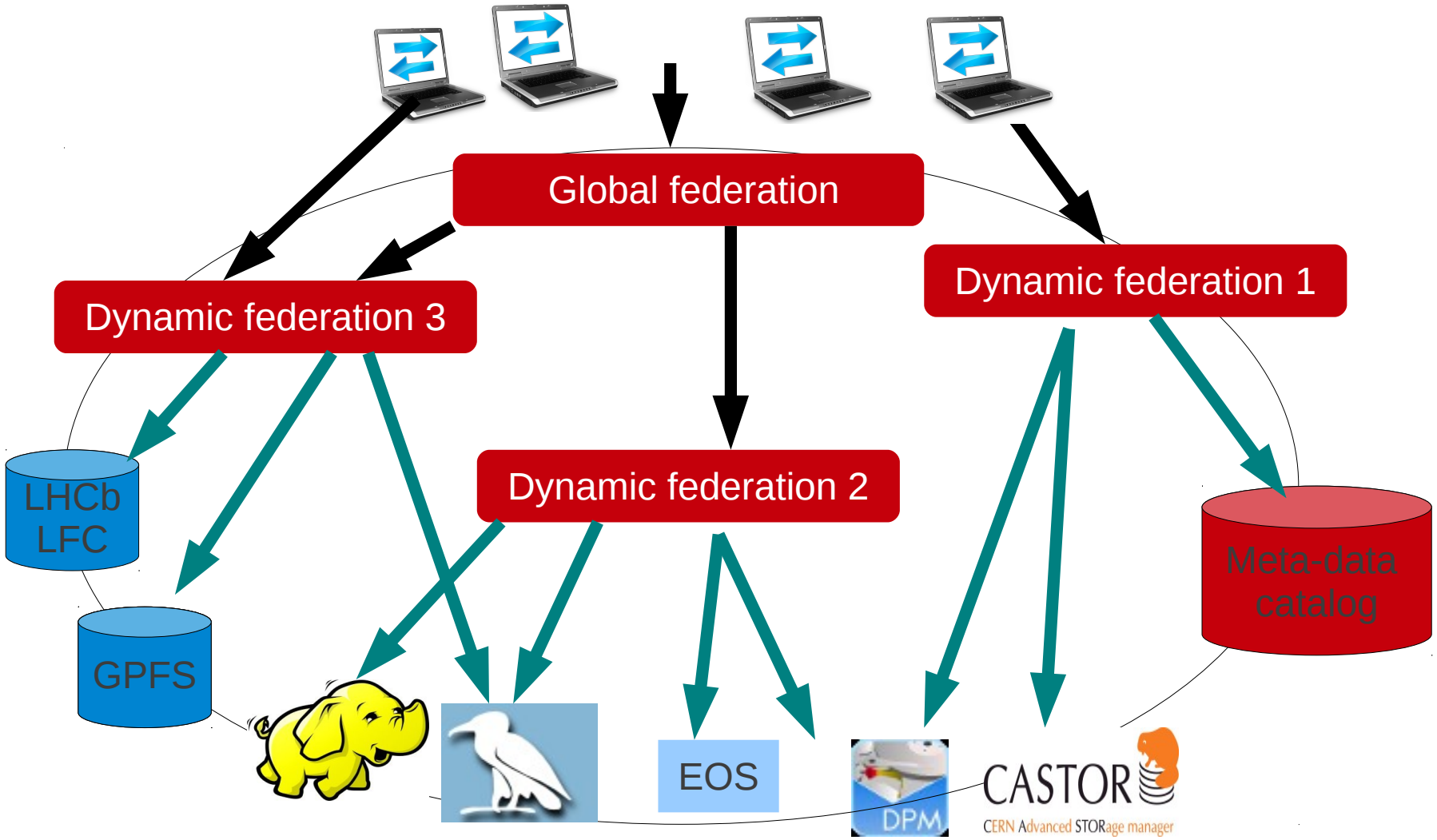
→ several layer of cache

→ No Database

→ Meta-data fetched dynamically from the Storage Elements.

Deployement possibilities

CERN IT Department



Performances test

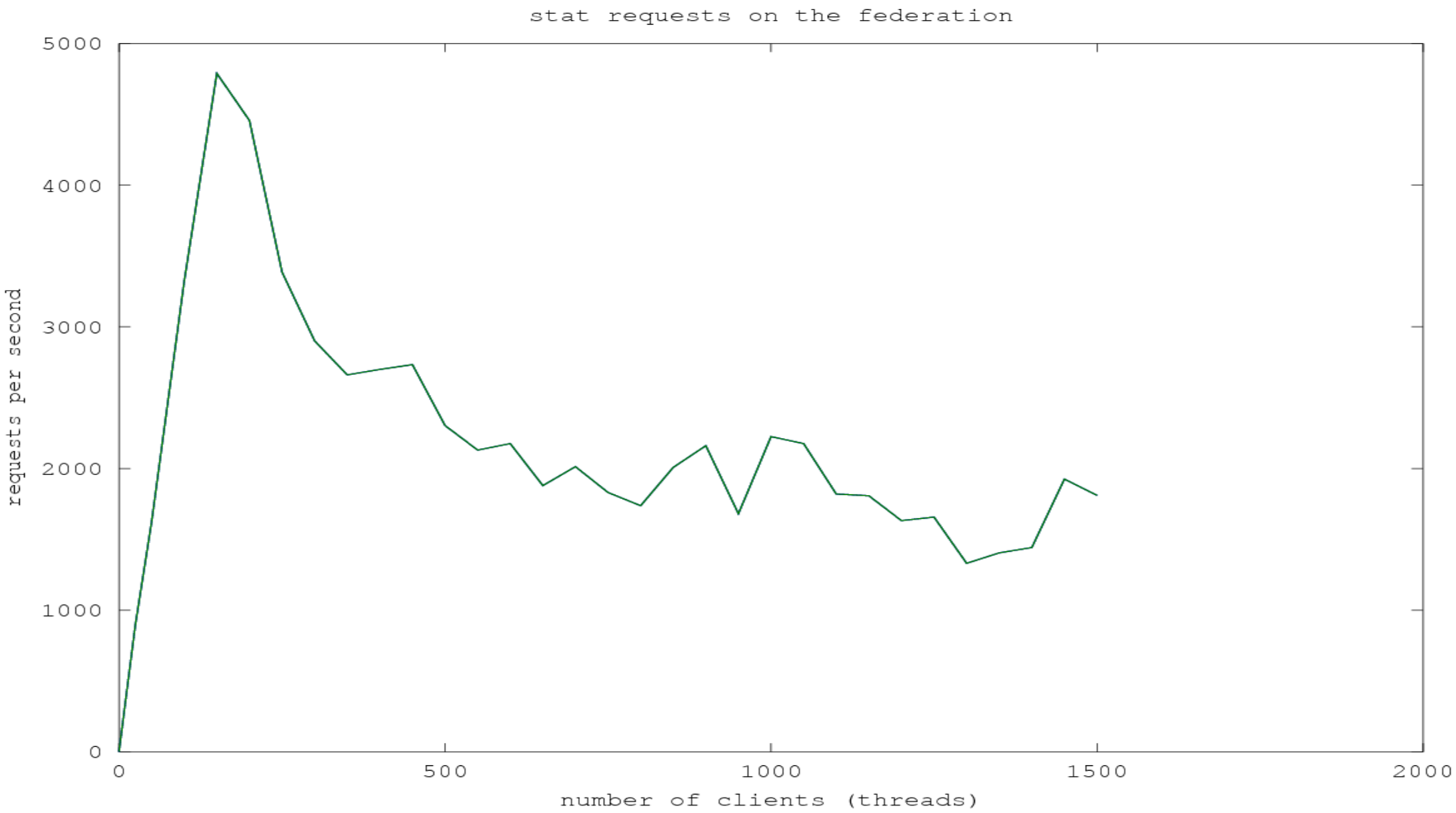
- **Two storage endpoints: DESY and CERN (poor VM)**
- **One UGR federator at DESY, clients at CERN**
- **10K files are interleaved in a 4-levels deep directory**

Oddly-numbered files are at CERN

Evenly-numbered files are at Desy

- **The test (written in C++) invokes Stat only once per file, using many parallel clients doing stat() at the maximum pace from 3 machines**

Performances results



- **Currently available!**

- **Technically TODAY we can dynamically aggregate:**
  - dCache DAV/HTTP instances
  - DPM DAV/HTTP instances
  - LFC DAV/HTTP and old Cns API instances
  - Cloud DAV/HTTP services
  - 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 one that's available uses GeoIP (free)**

We have a stable demo testbed, using HTTP/DAV

http://federation.desy.de/myfed

Dynamic Storage Federation Demo

- **It is actually 2 demos in one**
  - **An ATLAS demo, federating 8 sites, plus LFC as name translator**
    - **Note that this is not the full ATLAS repo, it's just 8 sites.**
    - **DESY, KIT, SARA, WUPPERTAL, NDGF, Muenchen, Prague, ASGC**
  - **A fully dynamic catalogue-free demo with the EMI testbed**
    - **Federating three endpoints.**
      - **a DPM instance at CERN**
      - **a dCache instance in DESY**
      - **one endpoint in LBNL**

- **Very stable, installable from the wiki**
- **Recent improvement: in the case we federate catalogues, the replicas they give can be checked o the fly**
  - **Use the catalogue as name translator**
  - **Use the catalogue as source of file listings**
  - **Check the replicas in the moment they are requested**
- **Next item: ATLAS and Rucio**
  - **We have a nice testbed, federating many ATLAS SEs**
    - **We want to federate the Rucio services and the LFC(s) seamlessly together**

Dynamic Storage Federation Links

Documentation and source code

– https://svnweb.cern.ch/trac/lcgdm/wiki/Dynafeds

Description of the demo

http://federation.desy.de/DynaFeds/The_Dynamic_Fed

Power users wanted

Helping in getting the best out of the system. Your cooperation is very appreciated.

Questions ?