Why use FTS?

3rd ESCAPE DIOS Workshop - https://indico.in2p3.fr/event/26307/

Steven Murray on behalf of the CERN FTS team

Tuesday 22nd March 2022

The FTS team and other contributors

FTS
File Transfer Service

The FTS team

- **CERN, Switzerland**
  - **Mihai Patrascoiu – Project leader**
  - **Joao Lopes – C++ / Python developer**
  - **Steven Murray – Service manager**
- **Indiana University Bloomington, USA**
  - **Edward Dambik – C++ / Python developer**

Other contributors

- **CBPF, Brazil**
  - **Eraldo Silva Junior – Python developer**

Many thanks to all of our contributors!

What is the File Transfer Service (FTS)?

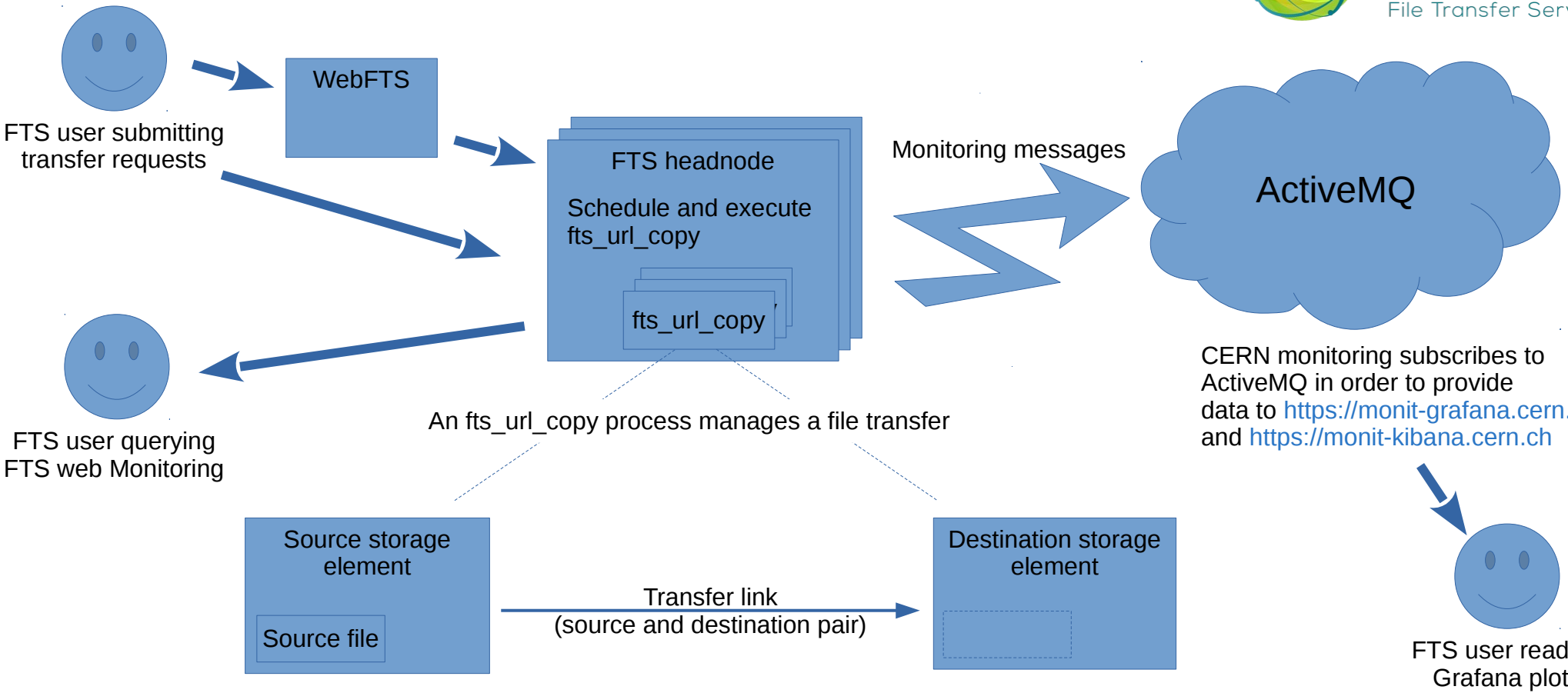
FTS
File Transfer Service

A collection of servers and clients that allow for the automated scheduling and execution of remote files transfers.

What does FTS do?

- Queues end-user transfer requests
- Schedules file transfers taking into account
  - Storage endpoint constraints
    - Inbound: maximum transfers and throughput
    - Outbound: maximum transfers and throughput
  - Link (source and destination pair) constraints
    - Minimum and maximum active transfers
  - Real-time errors
- Executes file transfers
  - 3rd party transfers are the default to avoid bottle necks
  - Transfers can fall back to “streaming” through FTS machines if needed
- Supports a variety of protocols against a variety of storage endpoints
- Supports X509 certificate authentication
- Supports OIDC token authentication - “proxy-less transfers”
  - ESCAPE, HIFIS and WLCG with EGI coming soon
- Publishes monitoring information about transfers

FTS in action



FTS clients

FTS command-line tools

- `fts-rest-transfer-submit -s https://fts3-devel.cern.ch:8446 gsiftp://source.host/file gsiftp://destination.host/file`
- `fts-rest-transfer-status -s https://fts3-devel.cern.ch:8446 c079a636-c363-11e3-b7e5-02163e009f5a`
- `fts-rest-transfer-cancel -s https://fts3-devel.cern.ch:8446 c079a636-c363-11e3-b7e5-02163e009f5a`
- `fts-rest-transfer-list -s https://fts3-devel.cern.ch:8446 -o atlas`

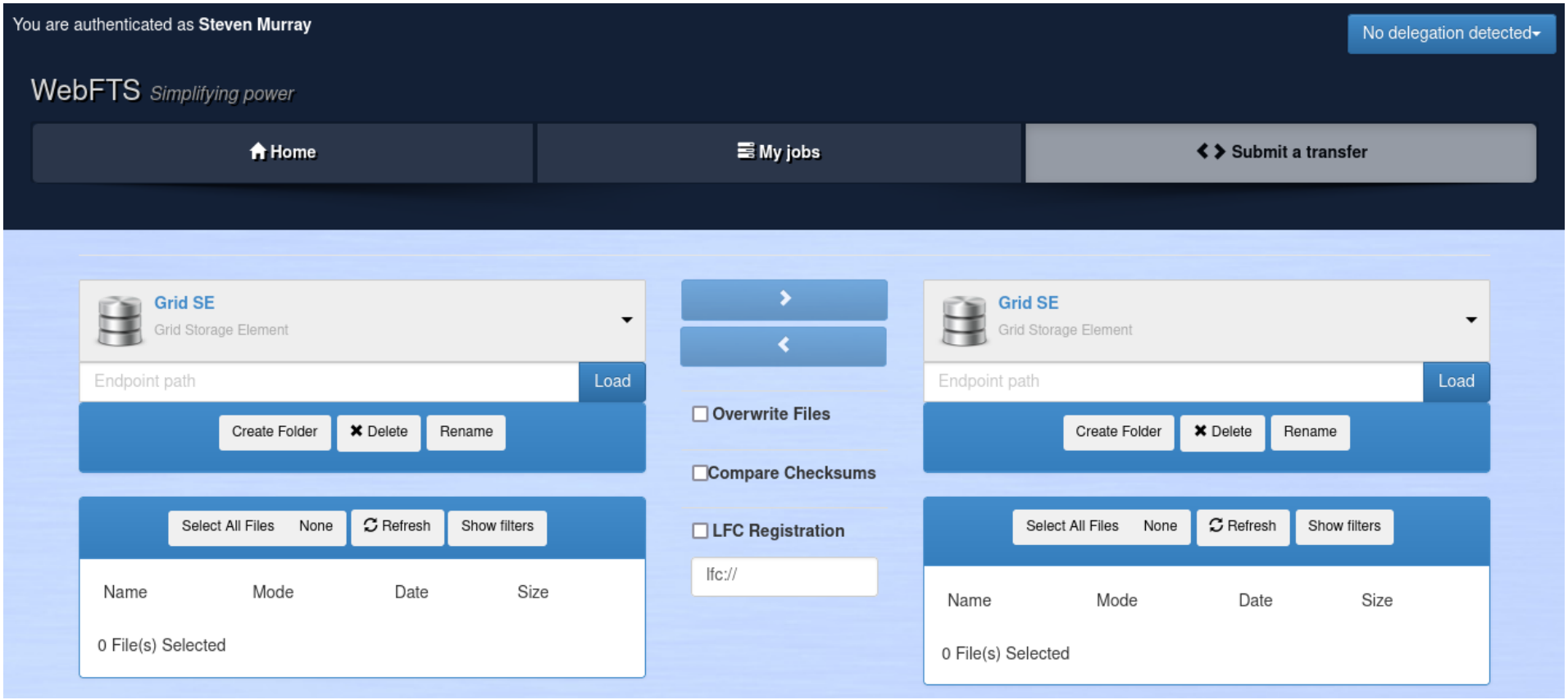
FTS python client library

- **Python 2 – to be deprecated**
- **Python 3 – the future**

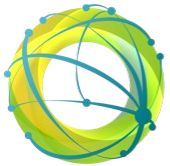
WebFTS web application

- **You favourite web browser**

WebFTS the web based GUI to FTS



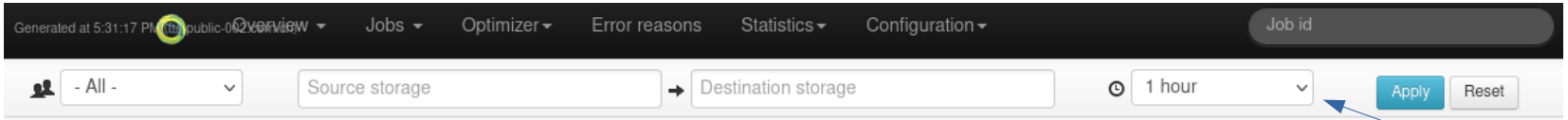
FTS uses GFAL2 to support multiple transfer protocols



FTS
File Transfer Service

- **FTS uses version 2 of the Grid File Access Library (GFAL2)**
- **GFAL2 wraps the different file access protocols of the Grid**
- **Supported protocols include:**
  - **HTTP/WebDav**
  - **GridFTP**
  - **SRM**
  - **XRootD**

FTS web monitoring – Overview page



FTS web monitoring can go back 6 hours

Click here to go to Grafana monitoring at CERN

Click here to go to the optimizer web page

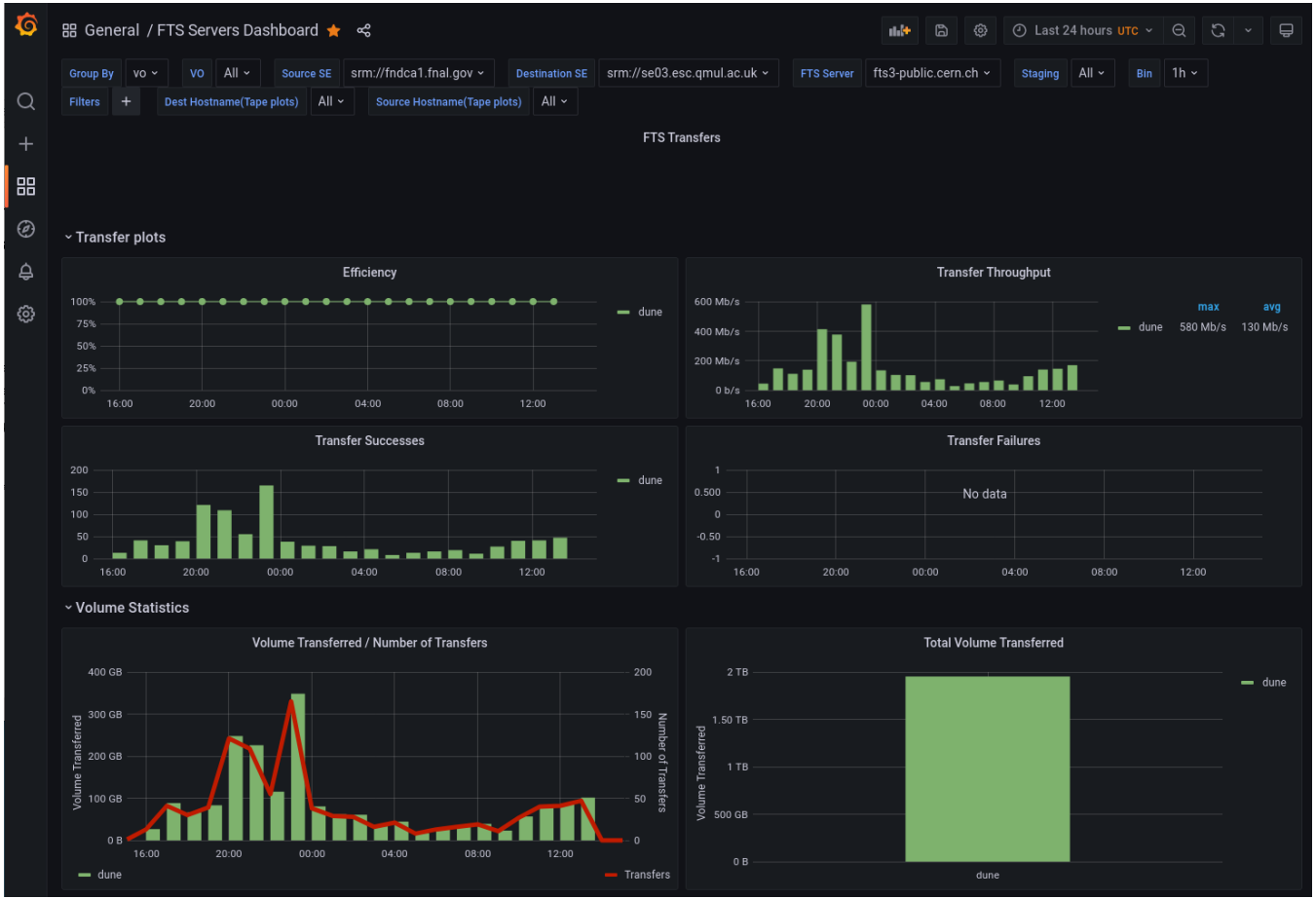
Overview

Showing 1 to 25 out of 25 from the last 1 hour

First | Previous | 1 | Next | Last

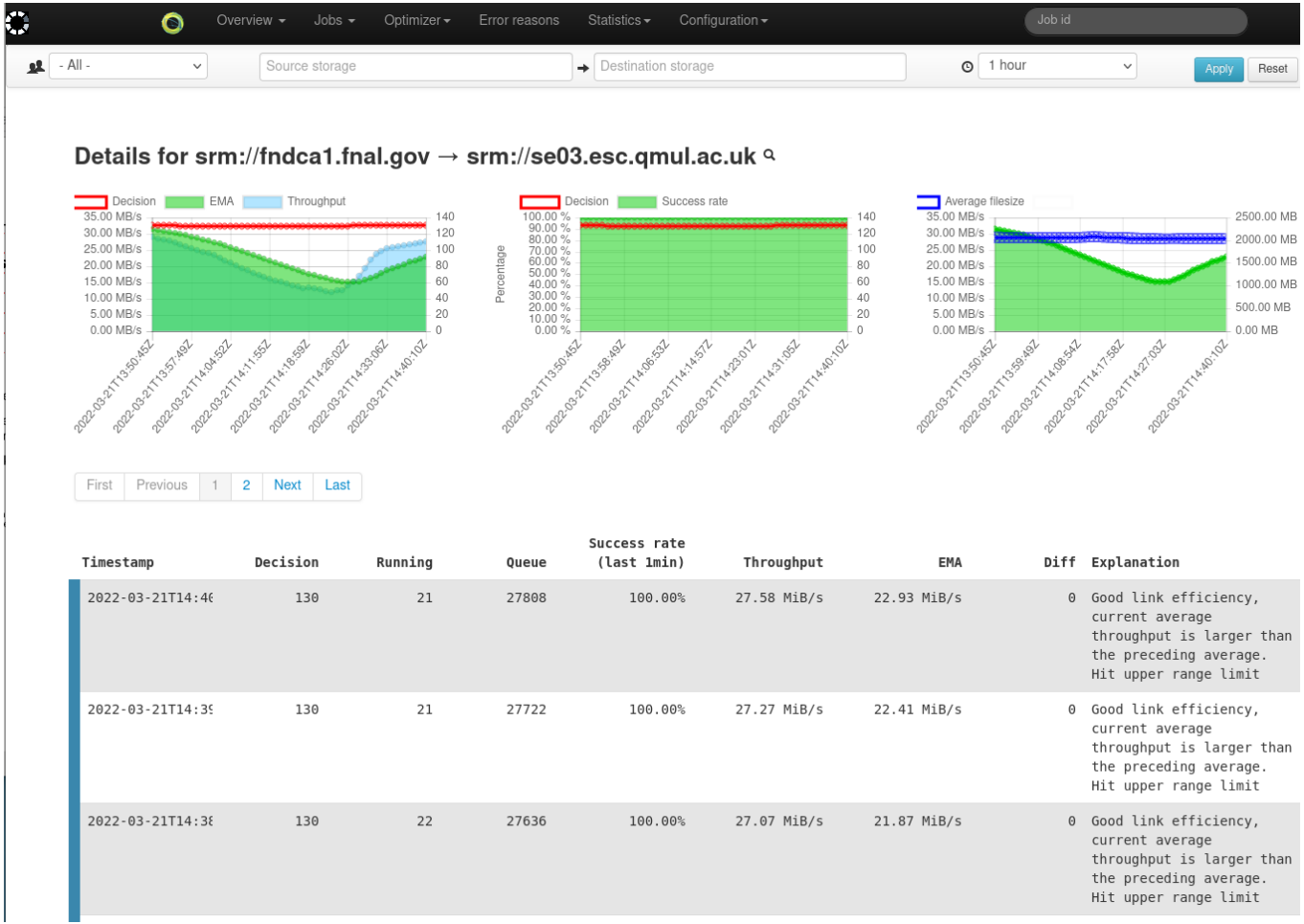
Source	Destination	VO	Submitted	Active	Staging	S.Active	Archiving	Finished	Failed	Cancel	Rate (last 1h)	Thr.
+ srm://fndca1.fna	srm://se03.esc.qm	dune	29985	28	40	16	-	52	-	7928	100.00 %	36.40 MB/s
Most frequent error Monitoring Link												
+ srm://fndca1.fna	gsiftp://dune.dca	dune	14167	35	47	16	-	306	-	247	100.00 %	183.30 MB/s
+ srm://fndca1.fna	gsiftp://bohr3226	dune	8865	102	88	55	-	750	19	2614	97.53 %	576.19 MB/s

Grafana monitoring at CERN



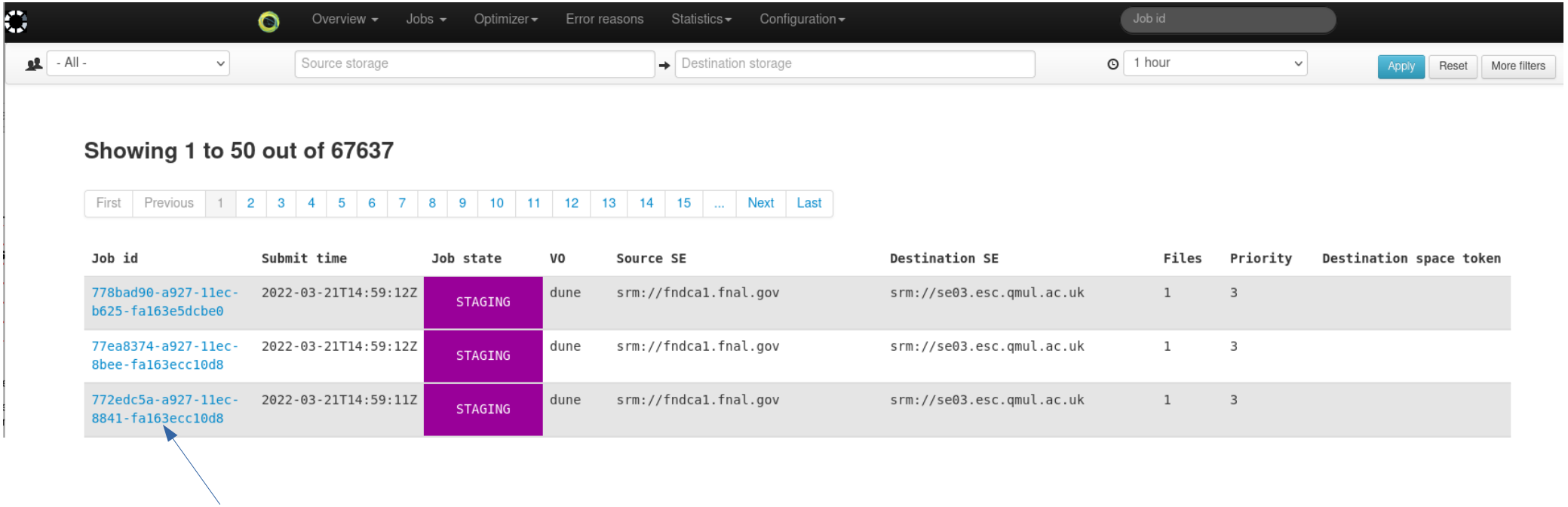
The data source of https://monit-grafana.cern.ch can go back 30 days

FTS web monitoring - optimizer



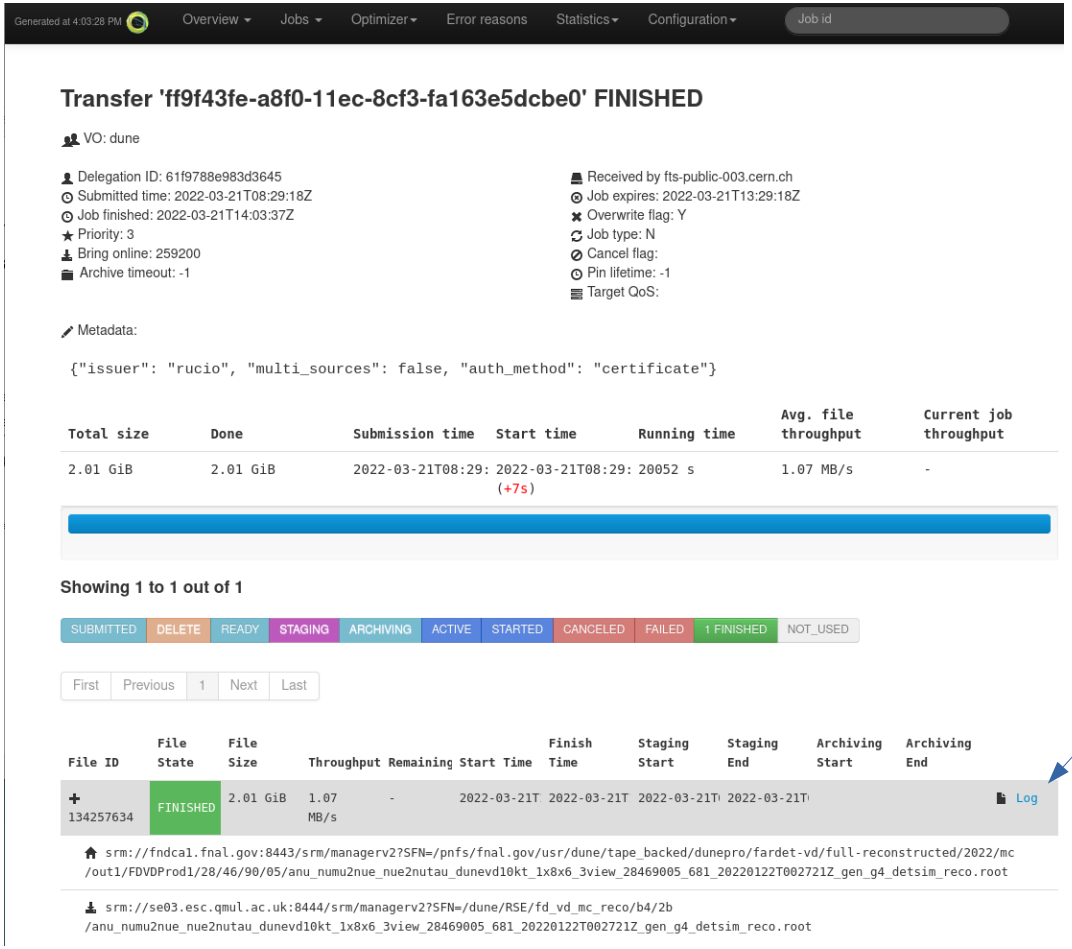
Timestamp	Decision	Running	Queue	Success rate (last 1min)	Throughput	EMA	Diff	Explanation
2022-03-21T14:4(	130	21	27808	100.00%	27.58 MiB/s	22.93 MiB/s	0	Good link efficiency, current average throughput is larger than the preceding average. Hit upper range limit
2022-03-21T14:3(	130	21	27722	100.00%	27.27 MiB/s	22.41 MiB/s	0	Good link efficiency, current average throughput is larger than the preceding average. Hit upper range limit
2022-03-21T14:3(	130	22	27636	100.00%	27.07 MiB/s	21.87 MiB/s	0	Good link efficiency, current average throughput is larger than the preceding average. Hit upper range limit

FTS web monitoring - Jobs



Click here to drill down into a specific job

FTS web monitoring – A job



Click here to get the logs of the `fts_url_copy` process

FTS in numbers

8 WLCG instances:

- BNL, CERN (4), FNAL, RAL and MIT

16 non-WLCG instances

- CERN (DAQ, Public), RAL, KEK(2), Imperial (also used by CMS), PIC, MWT2, CESNET (WebFTS + RCAuth), JINR, CNAF, SARA, SLAC, IHEP, Fermilab (containers), FENIX Research Infrastructure (Human Brain Project)

Approximately 36 Virtual Organisations (VOs)

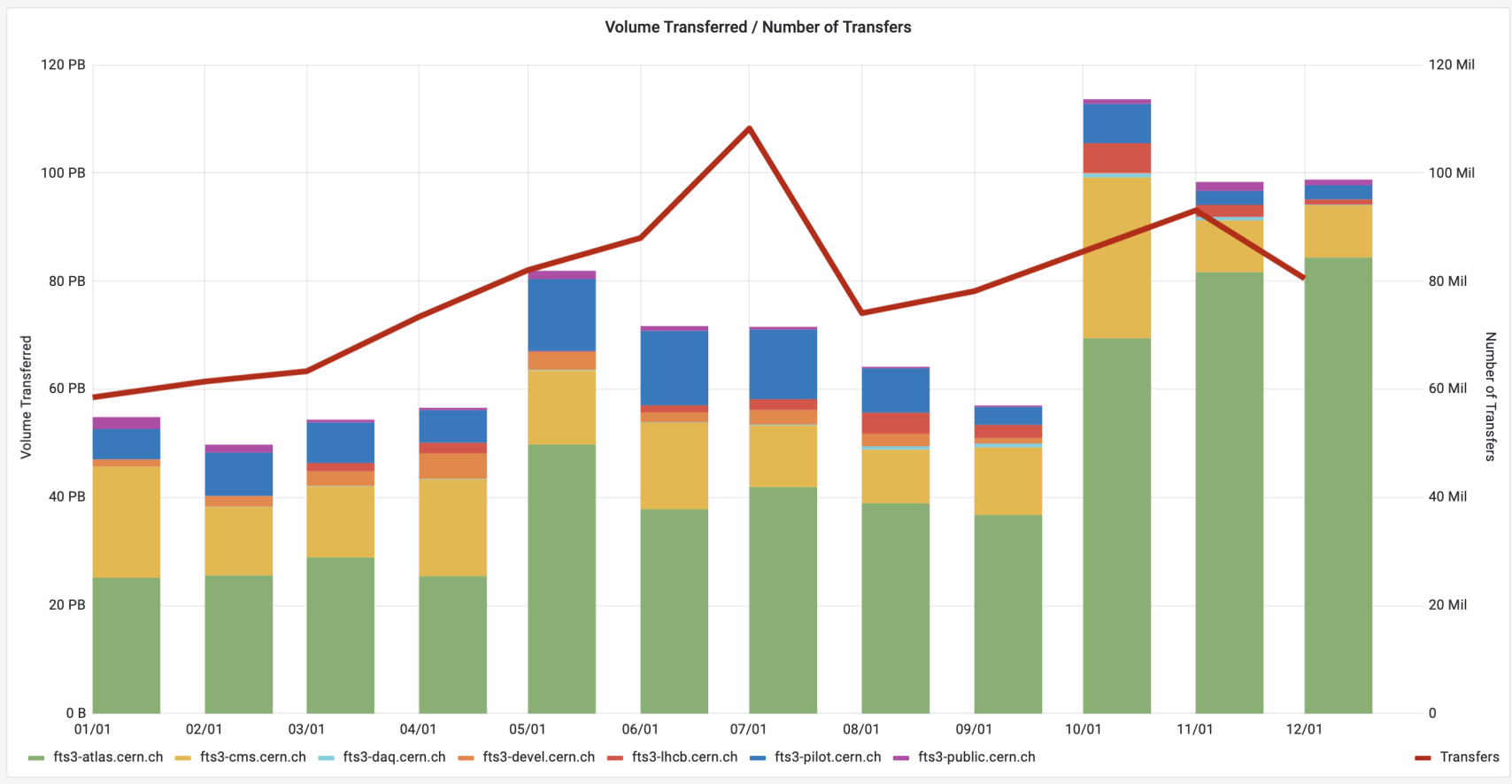
- ATLAS, CMS, LHCb, AMS, NA62, Compass, ILC, Magic, Belle, Mice, Xenon, Snoplus, Gridpp, Dune, LZ, Solidexperiment.org, SKA, Ligo, Icecube, Elixir, NP02(part of Dune), CAST, ESCAPE, Eiscat.se, Virgo, Pierre Auger Observatory, BES III, JUNO, CEPC, FENIX-RI, CTA, T2K, Project8, ICARUS, FASER, Folding@Home

Used by more than 35 experiments at CERN.

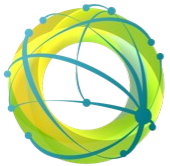
Transfers

- 2018: 848 PBs and 986M files
- 2019: 1.27 EBs and 1.08B files
- 2020: > 1.0 EBs and ~ 1B files
- 2021: > 1.0 EBs and > 1.15B files

Data moved by CERN FTS instances during 2021



Why use FTS?



Intuitive

- Simple interfaces to submit transfers
  - REST API
  - Command-line
  - Web based GUI via the WebFTS portal

Robust

- Transfers support checksumming and automatic retries

Flexible

- Support for multiple protocols: HTTPS, GridFTP, XRoot, SRM, S3 and more
- Real-time monitoring and runtime configuration

Adaptive

- Runtime optimisation: maximise throughput without overloading storage endpoints
- Transfer classification using priorities and activity types

Contacts and links

Main website: https://fts.web.cern.ch/fts/

Support questions: fts-support@cern.ch

Technical questions: fts-devel@cern.ch

CERN support ticket: https://cern.service-now.com

Operations forum: fts-ops-forum@cern.ch

FTS git repository: https://gitlab.cern.ch/fts/fts3

GFAL2 git repository: https://gitlab.cern.ch/dmc/gfal2

WebFTS git repository: https://gitlab.cern.ch/fts/webfts

Installation instructions: https://fts3-docs.web.cern.ch/fts3-docs/docs/install.html

home.cern