ATLAS & FTS: Reflections and ideas

XRootD and FTS Workshop @ STFC UK

2023-06-09

Alessandra Forti
on behalf of
ATLAS Distributed Computing

Part I

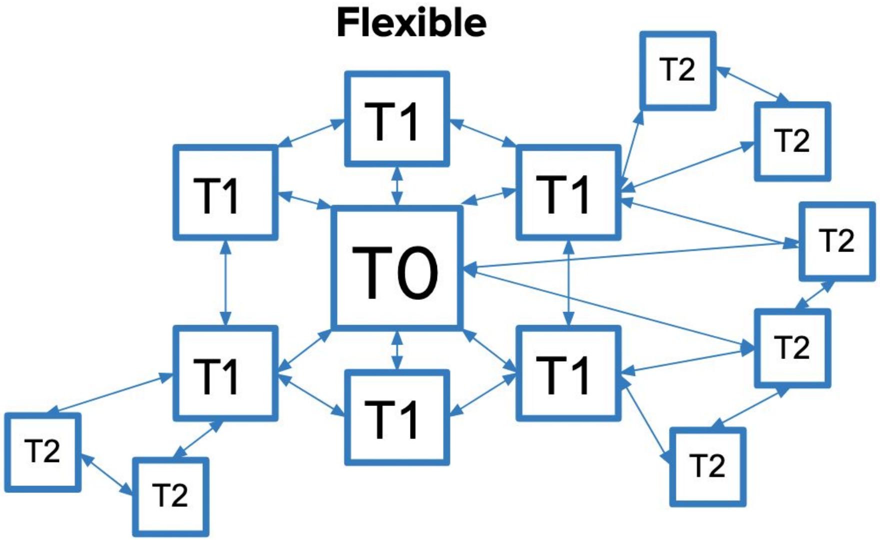
Retrospective

Introduction

- ATLAS transfers topology is a fully connected mesh
  - **All sites** transfer to **all other sites**
  - Some sites have **special roles** (output aggregation, merging, long-term archive, HPC, cloud, …)
  - 1 **Tier-0**, 9 **Tier-1**s, 56 **Tier-2**s, 68 **Tier-3**s
- Large range of file sizes due to many different ATLAS activities
  - O(10) kB - O(10)GB
  - ATLAS activities != Rucio/FTS activities
  - File sizes within one ATLAS activity tend to be consistent
- Two major levels of storage quality
  - Tape and disk
- 3 FTS instances

↓

- **Large combination of different types of transfer**

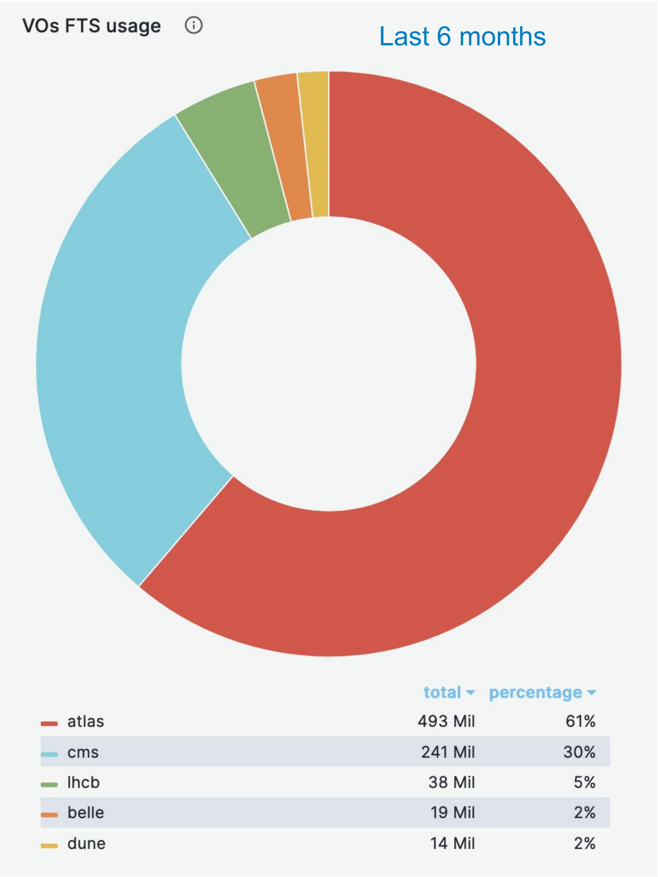


FTS

- FTS is a core service for ATLAS
  - Currently 60% of all FTS transfers are ATLAS
  - Only small storageless sites do remote access

- 3 FTS instances
  - CERN
    - Handles majority of sites and transfers
  - BNL
    - Manages transfers with US sites as destination
    - Can serve as a failover in case of CERN FTS issues
  - CERN Pilot
    - Used for testing new features, e.g. tokens

- Challenges in keeping the three instances aligned with the same configuration
  - Channel settings, HTTP libraries, plugins, etc.



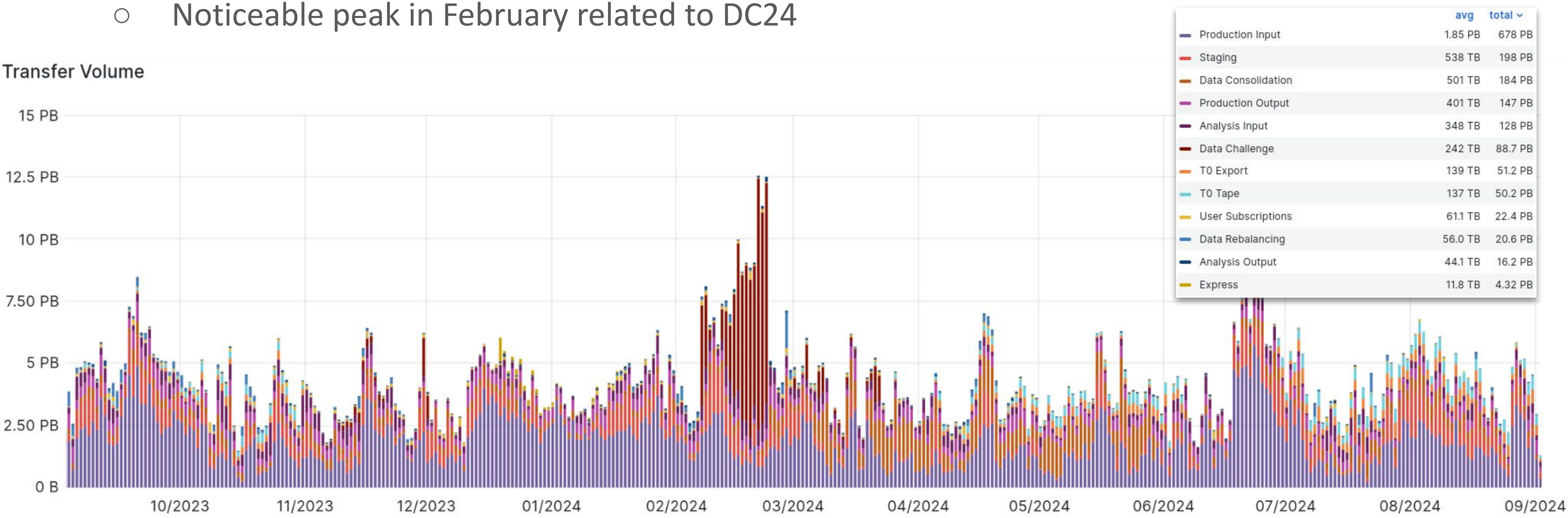
VOs FTS usage

Last 6 months

	total	percentage
🟥 atlas	493 Mil	61%
🟦 cms	241 Mil	30%
🟩 lhcb	38 Mil	5%
🟧 belle	19 Mil	2%
🟨 dune	14 Mil	2%

One year usage :: Data transfer volume

- More than half of our transfer volume due to production activity
  - We have to move this data
  - There is neither enough CPU nor storage for the textbook "job to data" case
- The remainder is a healthy mix of various experiment activities
  - Tier-0 Export, Consolidation of job outputs, rebalancing of data between sites, …
  - Noticeable peak in February related to DC24



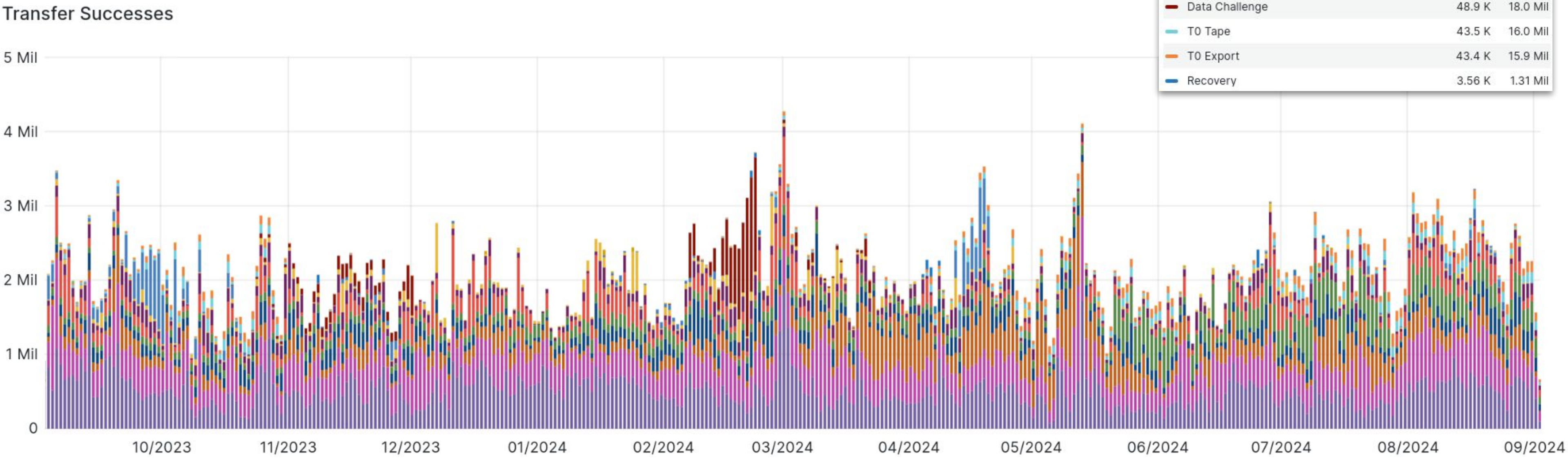
Transfer Volume

	avg	total
Production Input	1.85 PB	678 PB
Staging	538 TB	198 PB
Data Consolidation	501 TB	184 PB
Production Output	401 TB	147 PB
Analysis Input	348 TB	128 PB
Data Challenge	242 TB	88.7 PB
T0 Export	139 TB	51.2 PB
T0 Tape	137 TB	50.2 PB
User Subscriptions	61.1 TB	22.4 PB
Data Rebalancing	56.0 TB	20.6 PB
Analysis Output	44.1 TB	16.2 PB
Express	11.8 TB	4.32 PB

One year usage :: Data transfer numbers

- Number of files transferred is a relevant metric for ATLAS
  - However, we need to have complete datasets at the destination
    - Doesn't help if we can move a lot of files, if they don't belong together
  - Average size of our datasets are in the order of 100 files
  - Typical input sizes for processing are tens of datasets
  - Usually 500-700'000 jobs concurrently in the system at any time

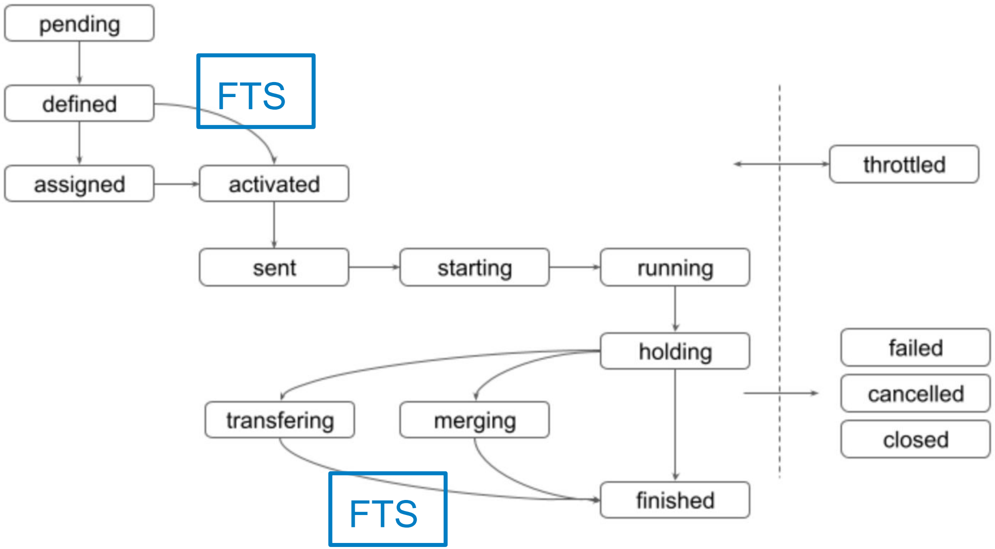
Transfer Successes



	avg	total ⌄
Production Input	514 K	189 Mil
Production Output	418 K	154 Mil
Data Consolidation	262 K	96.2 Mil
Analysis Output	179 K	65.6 Mil
Functional Test	178 K	65.4 Mil
Staging	164 K	60.2 Mil
Analysis Input	129 K	47.2 Mil
User Subscriptions	54.3 K	19.9 Mil
Data Rebalancing	52.3 K	19.2 Mil
Data Challenge	48.9 K	18.0 Mil
T0 Tape	43.5 K	16.0 Mil
T0 Export	43.4 K	15.9 Mil
Recovery	3.56 K	1.31 Mil

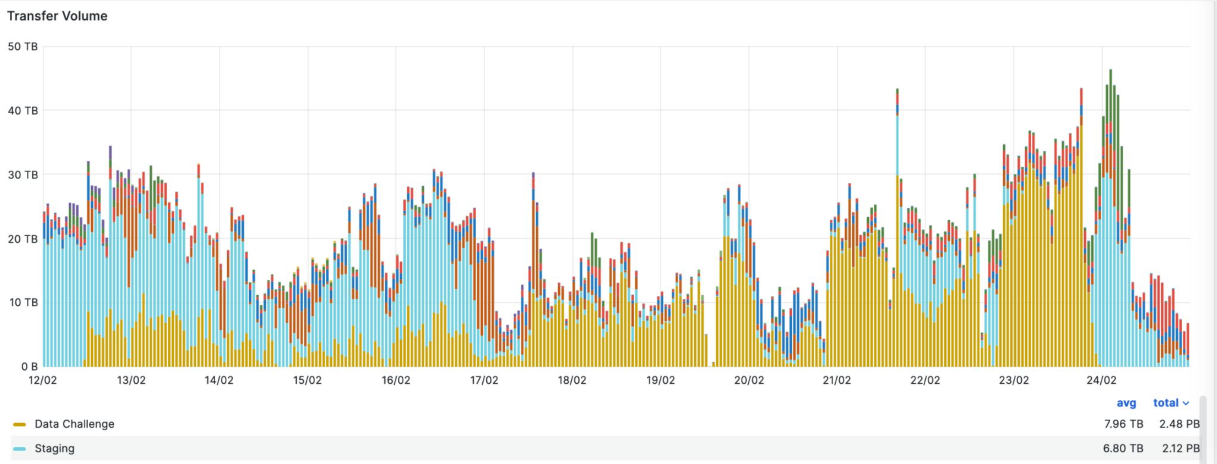
ATLAS Workflow management system (WFMS) and FTS

- FTS transfers are core to the production system
  - Pending, defined, assigned job states
    - Depend on FTS data transfers
  - Finished or failed job states
    - Depend on FTS data transfers
  - Job uploads to local storage
    - FTS replicates to remote storages
- Workflow management system
  - Continuous adjustments
  - Interaction with FTS
    - Very slow to change anything
- Experiment computing
  - **Not compatible** with traditional requirements engineering
    - Year long processes
    - Collection, design, code
  - Need to design well, modular
  - Be able to react quickly



Tape highlights

- Tapes are not just an archiving system in ATLAS
  - Dynamically used by the production system in Data Carousel mode at the same level as disk
  - Even users jobs can recall files from tape
- Transfers involving tape end points can be both local and remote
  - FTS makes no distinction, it is just another transfer
  - During DC24 Data Consolidation and Data Carousel starved DC24 transfers bandwidth and slots
    - Caused either by the experiment prioritization choice or by activity fairshares not working properly → needs more testing
- All tape systems behave differently
  - Getting a uniform behaviour may require coordinated development with different storages



	avg	total
Data Challenge	7.96 TB	2.48 PB
Staging	6.80 TB	2.12 PB

Example of staging interfering with DC24

Commercial clouds

- Google
  - ATLAS Google Project: Total Cost of Ownership & Evaluation of a commercial cloud
    - Multi-year project, touching everything from batch compute, to burst scaling, to PB-scale storage
  - SWT2 is now deploying part of their pledge using Google Cloud
    - ATLAS development of new URL signature mechanism inside Davix for seamless cloud storage integration
    - Looking forward to trying it out at scale via FTS

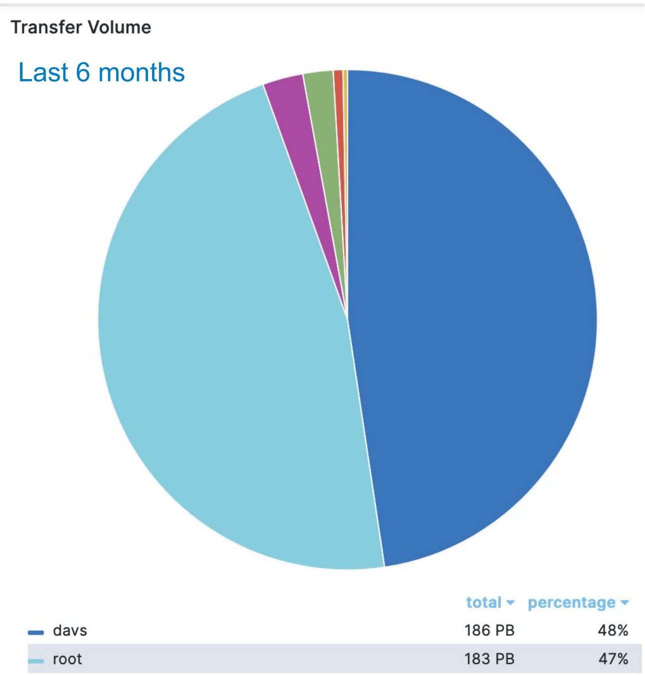
- SEAL
  - Commercial cloud cloud startup developing cryptographically verifiable storage
    - Provides ATLAS with 10PB of storage with no egress fees
    - Using standard S3-based frontends, so our tools just work™
  - Mostly network-related challenges as they are neither on LHCOPN nor LHCONE
    - Caused WLCG sites to route traffic through the public internet, circumvented with Rucio multi-hops
    - They are now deploying ESnet-peered routers so they can join LHCONE

root://

- The root:// protocol for third-party-copy is not relevant for ATLAS

- However, if we sum up all the activities
  - root:// and davs:// have a similar amount of usage
  - ATLAS uses them for completely different activities
    - davs:// for third-party-copy
    - root:// for jobs which read from local storage and XCaches

- Shoveler monitoring still important
  - The volume read via directIO/caches is overestimated
  - Current monitoring uses the full file size
    - Instead of the fraction of data actually accessed



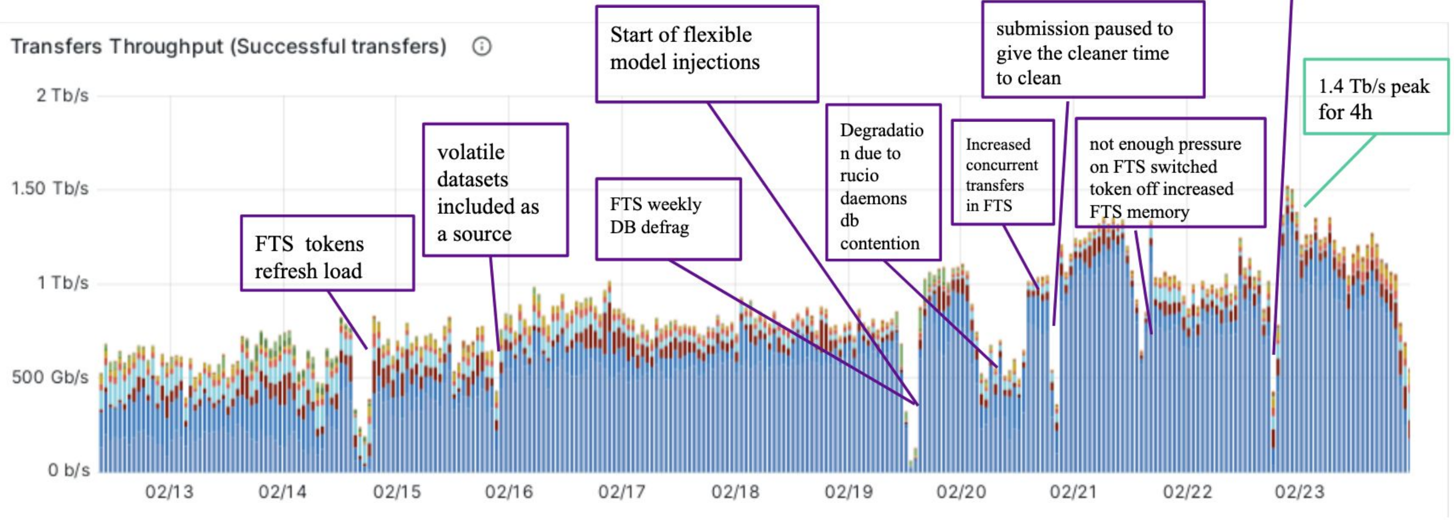
Transfer Volume

Last 6 months

	total	percentage
davs	186 PB	48%
root	183 PB	47%

Zooming in on DC24

- DC24 was a success in finding bottlenecks, not achieving rates
  - It required a lot of babysitting, constant monitoring and interventions
    - Configuration, hardware, number of instances, daily ops….
  - Almost daily manual tuning
    - Particularly for concurrent transfers values
    - We calculated and configured what we expected but…
    - … then we had to add a factor 5-10 to barely reach the goal



FTS tokens refresh load

volatile datasets included as a source

FTS weekly DB defrag

Start of flexible model injections

Degradation due to rucio daemons db contention

Increased concurrent transfers in FTS

submission paused to give the cleaner time to clean

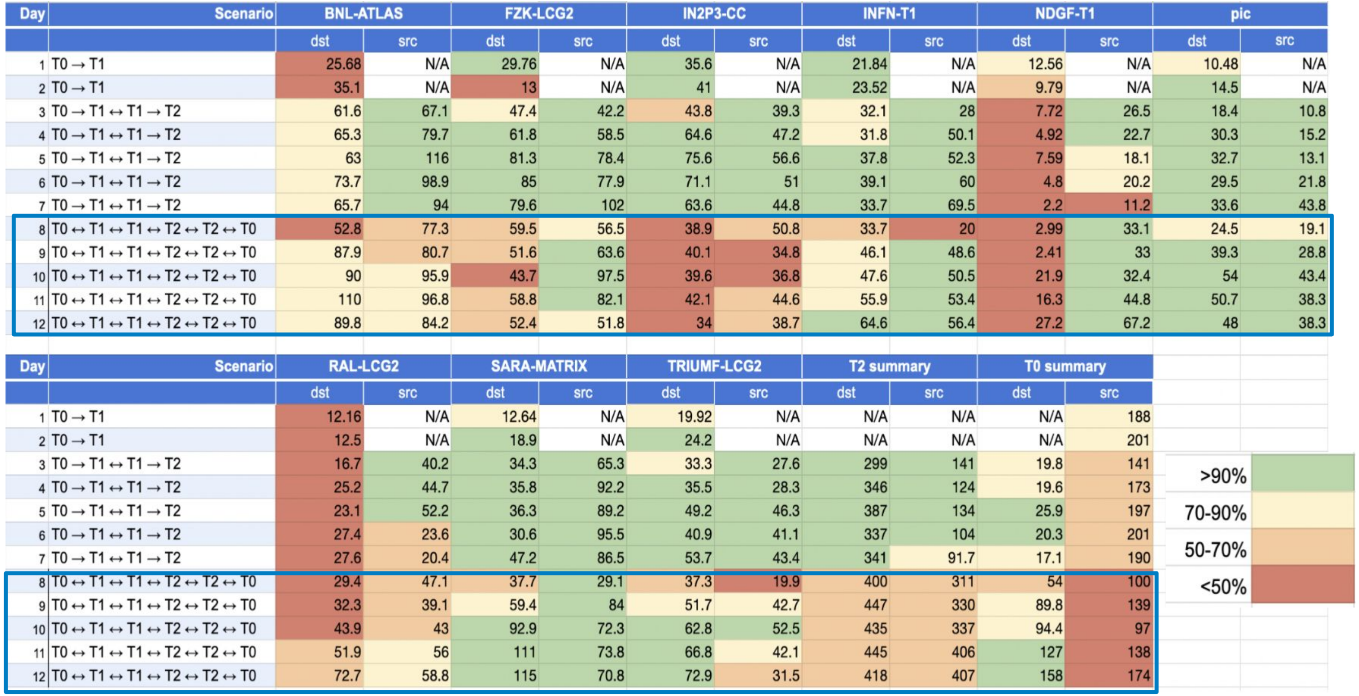
not enough pressure on FTS switched token off increased FTS memory

stopped submissions installed second high memory FTS instance for T2s. Cleanup 3M cancelled transfers

1.4 Tb/s peak for 4h

Zooming in on DC24

- Too many unprioritized transfers in the system affected heavily the rates
  - It worked well in the dress rehearsal and the first week
    - We had way less transfers *and* the rates were above target
  - There was a clear degradation in the second week when we injected for the full topology



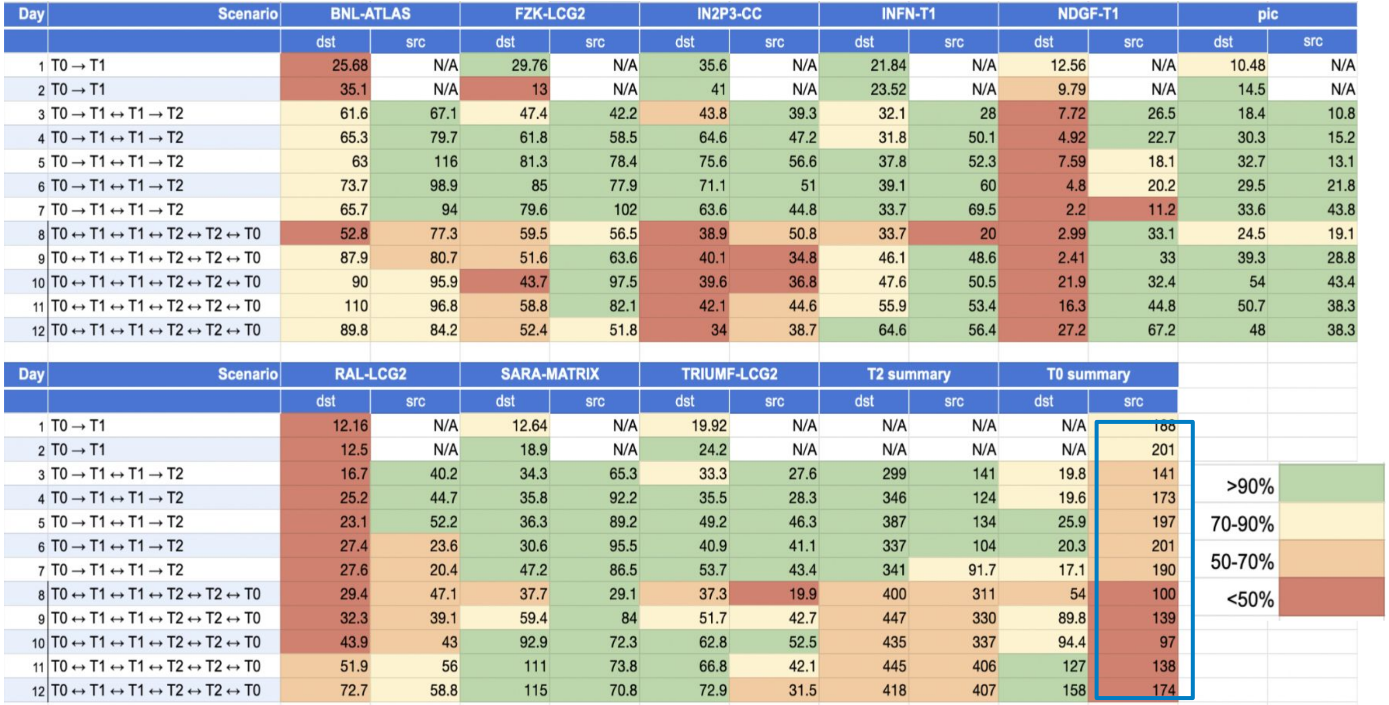
Day	Scenario	BNL-ATLAS		FZK-LCG2		IN2P3-CC		INFN-T1		NDGF-T1		pic	
		dst	src	dst	src	dst	src	dst	src	dst	src	dst	src
1	T0 → T1	25.68	N/A	29.76	N/A	35.6	N/A	21.84	N/A	12.56	N/A	10.48	N/A
2	T0 → T1	35.1	N/A	13	N/A	41	N/A	23.52	N/A	9.79	N/A	14.5	N/A
3	T0 → T1 ↔ T1 → T2	61.6	67.1	47.4	42.2	43.8	39.3	32.1	28	7.72	26.5	18.4	10.8
4	T0 → T1 ↔ T1 → T2	65.3	79.7	61.8	58.5	64.6	47.2	31.8	50.1	4.92	22.7	30.3	15.2
5	T0 → T1 ↔ T1 → T2	63	116	81.3	78.4	75.6	56.6	37.8	52.3	7.59	18.1	32.7	13.1
6	T0 → T1 ↔ T1 → T2	73.7	98.9	85	77.9	71.1	51	39.1	60	4.8	20.2	29.5	21.8
7	T0 → T1 ↔ T1 → T2	65.7	94	79.6	102	63.6	44.8	33.7	69.5	2.2	11.2	33.6	43.8
8	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	52.8	77.3	59.5	56.5	38.9	50.8	33.7	20	2.99	33.1	24.5	19.1
9	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	87.9	80.7	51.6	63.6	40.1	34.8	46.1	48.6	2.41	33	39.3	28.8
10	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	90	95.9	43.7	97.5	39.6	36.8	47.6	50.5	21.9	32.4	54	43.4
11	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	110	96.8	58.8	82.1	42.1	44.6	55.9	53.4	16.3	44.8	50.7	38.3
12	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	89.8	84.2	52.4	51.8	34	38.7	64.6	56.4	27.2	67.2	48	38.3

Day	Scenario	RAL-LCG2		SARA-MATRIX		TRIUMF-LCG2		T2 summary		T0 summary	
		dst	src	dst	src	dst	src	dst	src	dst	src
1	T0 → T1	12.16	N/A	12.64	N/A	19.92	N/A	N/A	N/A	N/A	188
2	T0 → T1	12.5	N/A	18.9	N/A	24.2	N/A	N/A	N/A	N/A	201
3	T0 → T1 ↔ T1 → T2	16.7	40.2	34.3	65.3	33.3	27.6	299	141	19.8	141
4	T0 → T1 ↔ T1 → T2	25.2	44.7	35.8	92.2	35.5	28.3	346	124	19.6	173
5	T0 → T1 ↔ T1 → T2	23.1	52.2	36.3	89.2	49.2	46.3	387	134	25.9	197
6	T0 → T1 ↔ T1 → T2	27.4	23.6	30.6	95.5	40.9	41.1	337	104	20.3	201
7	T0 → T1 ↔ T1 → T2	27.6	20.4	47.2	86.5	53.7	43.4	341	91.7	17.1	190
8	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	29.4	47.1	37.7	29.1	37.3	19.9	400	311	54	100
9	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	32.3	39.1	59.4	84	51.7	42.7	447	330	89.8	139
10	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	43.9	43	92.9	72.3	62.8	52.5	435	337	94.4	97
11	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	51.9	56	111	73.8	66.8	42.1	445	406	127	138
12	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	72.7	58.8	115	70.8	72.9	31.5	418	407	158	174

>90%	🟩
70-90%	🟨
50-70%	🟧
<50%	🟥

Zooming in on DC24

- Tier-0 outgoing rates were not what we expected
  - 3 Tier-1s had problem with storage, bandwidth starvation and/or with the monitoring resolution
  - In the second week a complete degradation of the T0 export rates



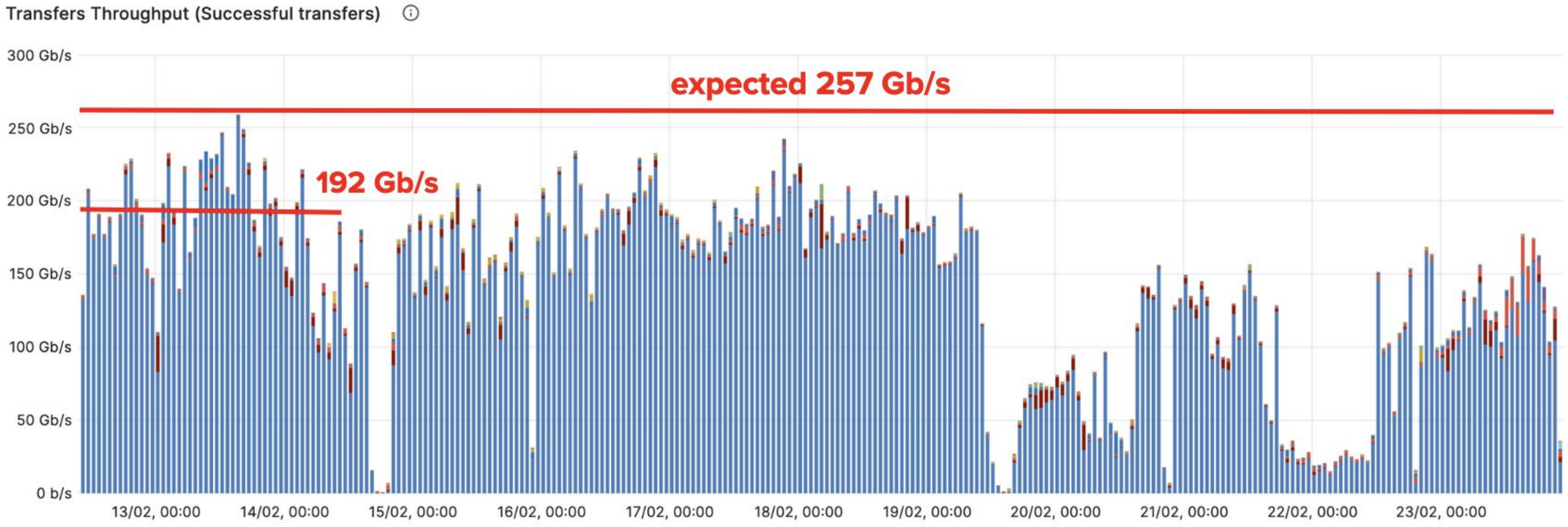
Day	Scenario	BNL-ATLAS		FZK-LCG2		IN2P3-CC		INFN-T1		NDGF-T1		pic	
		dst	src	dst	src	dst	src	dst	src	dst	src	dst	src
1	T0 → T1	25.68	N/A	29.76	N/A	35.6	N/A	21.84	N/A	12.56	N/A	10.48	N/A
2	T0 → T1	35.1	N/A	13	N/A	41	N/A	23.52	N/A	9.79	N/A	14.5	N/A
3	T0 → T1 ↔ T1 → T2	61.6	67.1	47.4	42.2	43.8	39.3	32.1	28	7.72	26.5	18.4	10.8
4	T0 → T1 ↔ T1 → T2	65.3	79.7	61.8	58.5	64.6	47.2	31.8	50.1	4.92	22.7	30.3	15.2
5	T0 → T1 ↔ T1 → T2	63	116	81.3	78.4	75.6	56.6	37.8	52.3	7.59	18.1	32.7	13.1
6	T0 → T1 ↔ T1 → T2	73.7	98.9	85	77.9	71.1	51	39.1	60	4.8	20.2	29.5	21.8
7	T0 → T1 ↔ T1 → T2	65.7	94	79.6	102	63.6	44.8	33.7	69.5	2.2	11.2	33.6	43.8
8	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	52.8	77.3	59.5	56.5	38.9	50.8	33.7	20	2.99	33.1	24.5	19.1
9	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	87.9	80.7	51.6	63.6	40.1	34.8	46.1	48.6	2.41	33	39.3	28.8
10	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	90	95.9	43.7	97.5	39.6	36.8	47.6	50.5	21.9	32.4	54	43.4
11	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	110	96.8	58.8	82.1	42.1	44.6	55.9	53.4	16.3	44.8	50.7	38.3
12	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	89.8	84.2	52.4	51.8	34	38.7	64.6	56.4	27.2	67.2	48	38.3

Day	Scenario	RAL-LCG2		SARA-MATRIX		TRIUMF-LCG2		T2 summary		T0 summary	
		dst	src	dst	src	dst	src	dst	src	dst	src
1	T0 → T1	12.16	N/A	12.64	N/A	19.92	N/A	N/A	N/A	N/A	188
2	T0 → T1	12.5	N/A	18.9	N/A	24.2	N/A	N/A	N/A	N/A	201
3	T0 → T1 ↔ T1 → T2	16.7	40.2	34.3	65.3	33.3	27.6	299	141	19.8	141
4	T0 → T1 ↔ T1 → T2	25.2	44.7	35.8	92.2	35.5	28.3	346	124	19.6	173
5	T0 → T1 ↔ T1 → T2	23.1	52.2	36.3	89.2	49.2	46.3	387	134	25.9	197
6	T0 → T1 ↔ T1 → T2	27.4	23.6	30.6	95.5	40.9	41.1	337	104	20.3	201
7	T0 → T1 ↔ T1 → T2	27.6	20.4	47.2	86.5	53.7	43.4	341	91.7	17.1	190
8	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	29.4	47.1	37.7	29.1	37.3	19.9	400	311	54	100
9	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	32.3	39.1	59.4	84	51.7	42.7	447	330	89.8	139
10	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	43.9	43	92.9	72.3	62.8	52.5	435	337	94.4	97
11	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	51.9	56	111	73.8	66.8	42.1	445	406	127	138
12	T0 ↔ T1 ↔ T1 ↔ T2 ↔ T2 ↔ T0	72.7	58.8	115	70.8	72.9	31.5	418	407	158	174

>90%	🟩
70-90%	🟨
50-70%	🟧
<50%	🟥

Zooming in on DC24

- Another view on Tier-0 outgoing rates
- Complete degradation in the second week
  - Mostly because of the number of unprioritized transfers within the DC activity
  - On top of the Data Challenge activity being in backfill mode



Zooming in on DC24

- Clear conflict between optimizing number of transfers and maintaining high rates
  - FTS called it misunderstanding but…
- In DC26 **both will have to be metrics**
  - **Number of transfers** important because we need to move many files between many locations
  - **Transfer rates important** because we have to build a resilient network
- DC26 = 2 x

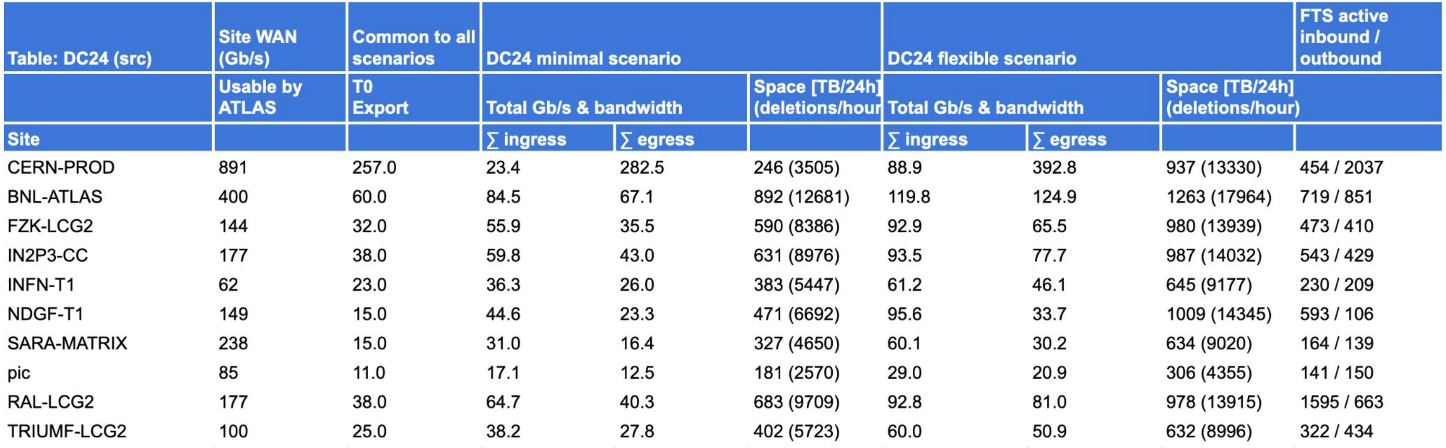


Table: DC24 (src)	Site WAN (Gb/s)	Common to all scenarios	DC24 minimal scenario			DC24 flexible scenario			FTS active inbound / outbound
	Usable by ATLAS	T0 Export	Total Gb/s & bandwidth		Space [TB/24h] (deletions/hour	Total Gb/s & bandwidth		Space [TB/24h] (deletions/hour)	
Site			∑ ingress	∑ egress		∑ ingress	∑ egress		
CERN-PROD	891	257.0	23.4	282.5	246 (3505)	88.9	392.8	937 (13330)	454 / 2037
BNL-ATLAS	400	60.0	84.5	67.1	892 (12681)	119.8	124.9	1263 (17964)	719 / 851
FZK-LCG2	144	32.0	55.9	35.5	590 (8386)	92.9	65.5	980 (13939)	473 / 410
IN2P3-CC	177	38.0	59.8	43.0	631 (8976)	93.5	77.7	987 (14032)	543 / 429
INFN-T1	62	23.0	36.3	26.0	383 (5447)	61.2	46.1	645 (9177)	230 / 209
NDGF-T1	149	15.0	44.6	23.3	471 (6692)	95.6	33.7	1009 (14345)	593 / 106
SARA-MATRIX	238	15.0	31.0	16.4	327 (4650)	60.1	30.2	634 (9020)	164 / 139
pic	85	11.0	17.1	12.5	181 (2570)	29.0	20.9	306 (4355)	141 / 150
RAL-LCG2	177	38.0	64.7	40.3	683 (9709)	92.8	81.0	978 (13915)	1595 / 663
TRIUMF-LCG2	100	25.0	38.2	27.8	402 (5723)	60.0	50.9	632 (8996)	322 / 434

Mini-DC heads up

- As per DOMA general presentation
  - At least US and UK will carry out further Mini Data Challenges
  - Starting from repeating with the DC24 rates
- Not as much pressure as DC24 because fewer sites will do concurrent transfers
  - Coordination of different countries challenges needs to be discussed
- Something to do in **cooperation also with FTS**

Part II

Wishlist

Priority Item: Communication

- For ATLAS there is not enough regular communication among stakeholders
    - Many requests dispersed in tickets
- Often not clear whose responsibility it is to follow up on an issue
    - Many teams are involved: storage, FTS, experiment, network, monitoring, …
    - Sometimes not directly an FTS problem, but FTS needs to be able to quickly adapt to changes
- We can discuss what is better, or if we need all of these
    - **Number one request: Resurrect FTS steering meetings** with all stakeholders
    - Have more frequent technical meetings between experiment and FTS
    - Ability to continuously adjust requirements
- What doesn’t work for ATLAS
    - Give us a list of requirements, we go away and come back in few months with a finished product

Development: Scheduler algorithm

- Current algorithm within an activity seemed to follow FIFO
  - During DC24 within an activity we had thousands of T2-T2 transfers queued in front of T0-T1
  - Transfers kept on being piled on links that were full
- New algorithms may be written….
  - To spread better the transfers
  - To follow prioritized links
  - Or throughput aware
  - ……
- FTS should support more than one algorithm
  - It should be easy to add new algorithms
  - It should be easy to switch from between algorithms
  - It would make easy to compare different algorithms

Development: Algorithms

- **Automatic tuning to optimise throughput**
  - Extensive **manual tuning** of links and storage was necessary
  - Sometimes **conflicting values** had to be set, all related to missing performance/throughput awareness
- **Priority of transfers other than per activity**
  - Within an activity there is **no prioritisation**
    - It should be possible to **prioritise links according to some weight**
    - **Faster links should be prioritised automatically**
  - **Track link performance** and use this information to help prioritise transfers
  - **Manually prioritise specific links** (source-destination) vs others using the same set of sources or destination
  - Add **deadline based scheduling**
- **The optimizer needs to be reviewed**
  - Cycle eventually was **taking multiple hours** and **couldn't be restored** to a working state easily or **turned off**
    - Enable the ability to turn off the optimizer
    - 2 out of 4 of the optimizer settings are not useful without parallel streams
  - Needed improvements
    - Fast scaling with the number of **active transfers** as well as a **fixed value**
    - **Fast automatically scale down** with the number of failures
      - It wasn’t clear if it was tied to the max number of transfers, but if it is it shouldn’t depend on it.
    - Significantly **improve the optimizer latency**, decreasing the cycle to a few minutes
    - When scheduling transfers, **available and used bandwidth** needs to be taken into account

Development: Parallel transfers

- In DC24 more concurrent transfers was the only way to increase the throughput
  - This can improve with better algorithms but …
- … the number of files to transfer and the bandwidth to fill will be larger anyway
  - HTTP-TPC doesn’t support parallel streams for a file transfers like GridFTP
  - Do we need parallel streams for HTTP-TPC?
    - Not a favourite solution, as it works only in pull mode
      - Clouds and some HPCs only work with push
    - Would require development work also from some storage systems
    - If xrootd http protocol already supports it we need to start testing it
      - UK miniDC is a good place to do it TBD

- Better support is needed for thousands of concurrent transfers
  - Repeated mini Data Challenges to test FTS hardware and sites storage limits
  - Including testing at scale

High-availability, resilience and scalability

- **Require high-availability and resilience for FTS components**
  - Consider **scaling as a central requirement** for future FTS, including underlying tools like gfal/davix
  - The weekly **defragmentation of the database**
    - During DC24 this standard **maintenance operation blocked transfers** on at least one occasion
    - Is there a way to redesign the DB such that it won’t block production?
  - **Cancelled jobs** were accumulating in the DB making it unresponsive and should be **removed automatically**
  - Improve the **latency in handling cancelled jobs**
    - Need to better **understand the reason** for the many cancelled transfers
    - At HL-LHC scale, with the given files/link scale, is **having mass cancellation of transfers normal**?
      - Significant site storage degradation or incidents are daily business

- **Memory usage**
  - It was recognised that currently the **way to scale right now is to add more memory** on fts3-atlas
  - Clearly **define the memory requirements** and ensure the production instances are compliant
  - During DC24 had to install a **second high memory instance** on fts3-pilot
    - Move all the T2s there to achieve the necessary rates
    - Before deciding to end DC24 some sites had been moved to FTS BNL too to spread the load further
  - Determine if FTS will **require multiple instances** to support ATLAS HL-LHC needs

Operations: WFMS tickets

- List of unresolved or partially resolved tickets
  - FTS-1748 Increment/decrement of channel maxActive based on success rate
    Took 2.5yrs to change 1 to 10 and still not obvious % of currently active
  - FTS-1497 Distribution of active transfers across channels from src at maxActive limit
    Very imbalanced but should follow activity shares.
  - FTS-1717 Combine endpoints on same limited network connection, e.g. tape+disk door
    Bandwidth limit does not work well AFAIK
    maxActive may be shares too, e.g.alias to same doors
  - FTS-1718 Don`t stop slow but progressing transfer after some long timeout (then retry!)
    If we want stop it, then do it as soon as clear it will hit the timeout
    based on size and rate
  - FTS-1750 Is about clearer errors for the client, e.g. on timeout
    Single stuck files from otherwise completed FTS job
    1 reason was landing on particular FTS server
    O(10) servers to *steer* TPC transfers should not be necessary.
  - HTTP-TPC transfers with no progress should be killed after X minutes (GridFTP timeout was 300s)

- They couldn't be (fully) applied to current FTS
  - Need to see if they can enter the list of requirements now

Tape recall and buffers

- Observed issues in ATLAS
  - Tape management systems seem to generally **prefer longer queues** to optimise recalls
    - However, we only do **partial requests in waves**, a.k.a *Data Carousel*
    - No need to recall more than we can process, usually multi-week long tasks with limited disk space
  - Additionally, **files can get wiped** from tape buffer due to timeouts
    - Typical case is that the **destination becomes unavailable** during *bringonline* and/or *pin lifetime*

- FTS should be able to help preventing *impossible-to-fulfil* recall from tape
  - Eventually, FTS should ensure it is able to transfer data from buffer to the disk
  - **Allow cancelling the recall** for files with a known bad destination, such as *unreachable*, *broken*, *full*, …
    - We can **propagate to FTS** these site states, with various levels of sophistication and speed
      - **Quick propagation** may lead to unnecessary cancelled queued transfers
      - **Slow propagation** still might cause full tape buffers
    - Additionally, support for potential **fallback destination** would be useful
  - There are **exceptions**, where we allow tape recall only to disk storage local to the site hosting the tape
    - This has a big impact on the required disk space — currently RAL and CERN are running in this mode

- Tape buffer management bugs, as mentioned in FTS-2043
  - Failed transfers are not removed from dCache buffers until the pin lifetime expires
    - Even if there is second successful transfer, i.e., all pins need to be released upon success
  - StoRM remove files after 3 days, CTA remove files after 1 day — The pin lifetime is ignored
  - **Pins need to be refreshed** when transfer not yet done but still ongoing
  - **Don't request a recall** when file is already in online buffer

Monitoring

- Currently there are some unresolved discrepancies between FTS and DDM dashboards.
  - Some are due to missing parameters in CRIC
  - Some are due to multi endpoint transfers
  - Some are due to how rates are calculated differently (probably also multi-replica)
  - Some transfers log are missing in FTS, and vice versa
  - Some transfers are marked as failed, but MONIT receives OK
  - Some transfers are sent to MONIT with a wrong link pair
  - ….
- The monitoring from same/similar sources has to show the same numbers and patterns
  - If they differ, they offer a completely different narrative and that is wrong!
- Monitoring needs to be also in line with the network monitoring
  - During DC24 there were a number of sites seeing different rates even when the accounted traffic was only FTS
- Last but not least
  - 1h minimal bin size in FTS dashboards is a problem we need better resolution in line with network plots

- **Monitoring cannot be an afterthought!**

Operations: CI/CD support and configuration

- More integrated CI/CD setup to allow us to experiment with feature prototyping
- This is important and could reduce the request for more meetings
  - A good part of the daily operations adjustments could be sped up if there was an integrated CI/CD in place
  - Allow experiments to dynamically enable and/or configure extensions/plugins/etc…

- Consistent enforced configuration
  - Currently configurations can only be easily changed via a web interface
    - If an experiment has multiple instances they will quickly and easily go out of sync
    - Web interface is buggy
  - There should be the possibility for FTS servers to exchange configurations
    - Or at least automatically point out differences in configurations, and make it easy to migrate
  - Is there a case also for software updates? TBD

Operations: Tokens

- During DC24 they were a secondary goal for ATLAS
  - They have been a success story, but they were an extra load
    - Created a number of rate drops and had to be switched off to reach target rates
- In DC26 **it will not be a secondary goal**
  - Battle-tested OIDC Token support a requirement
  - Timeline for tokens workflows to be supported at Run-4 scale and available for stress testing?
    - Do we need miniDCs for this?

- See Dimitrios talk for more in depth technical details

Other topics

- **Short(ish) term topics**
  - Development: Limit enforcement
  - Development: Fair-sharing per endpoint
  - Development: Modify prioritisation of transfers
  - Development: Improve web interface
- **Medium term topics**
  - Development: Better automatic source selection
  - Development: Revisit automatic session reuse
  - Development: Smarter back pressure mechanism from storage to FTS
    - Relevant during DC24 and requested by sites during the last tape consolidation campaigns
    - Will need stakeholder involvement
    - And no, fail-retry is not back pressure
  - Operations: Easier debugging of failed FTS transfer
    - Development: Coherence of returns / errors from storage providers
- **Long term topics**
  - Labelling of transfers for networks via SciTags
  - Load balancing
    - Required for site deployments with multiple destination endpoints sharing a single storage, e.g., DTNs for HPCs
    - Requires FTS awareness of VO specific storage endpoints and their characteristics
  - Cross-experiment scheduling
    - Many of our storage and networks are shared across experiments
    - Scheduling needs to take that into account

- **ATLAS has had a document with some long standing requests shared with FTS team since the Alejandro A Aylon times**

Summary

- FTS is a crucial component for the future of ATLAS
  - Fast and reliable data transfers are fundamental to what we do — and FTS does it!
  - However, we do spend a considerable time debugging ongoing transfer issues
- We have identified issues and proposed several enhancements for FTS
  - Addressing ongoing operational issues
  - Anticipating future needs, incl. Data Challenges, site needs, and HL-LHC scale
- ATLAS is worried that FTS is not getting the necessary support it needs and deserves
  - History of rotating persons with no long-term perspectives
  - We also believe the FTS team is critically understaffed
- ATLAS, and especially ADC, is looking forward to our future collaboration!
  - Step 1: Re-establish steering meetings :-)

ATLAS

EXPERIMENT