



CMS Community Report

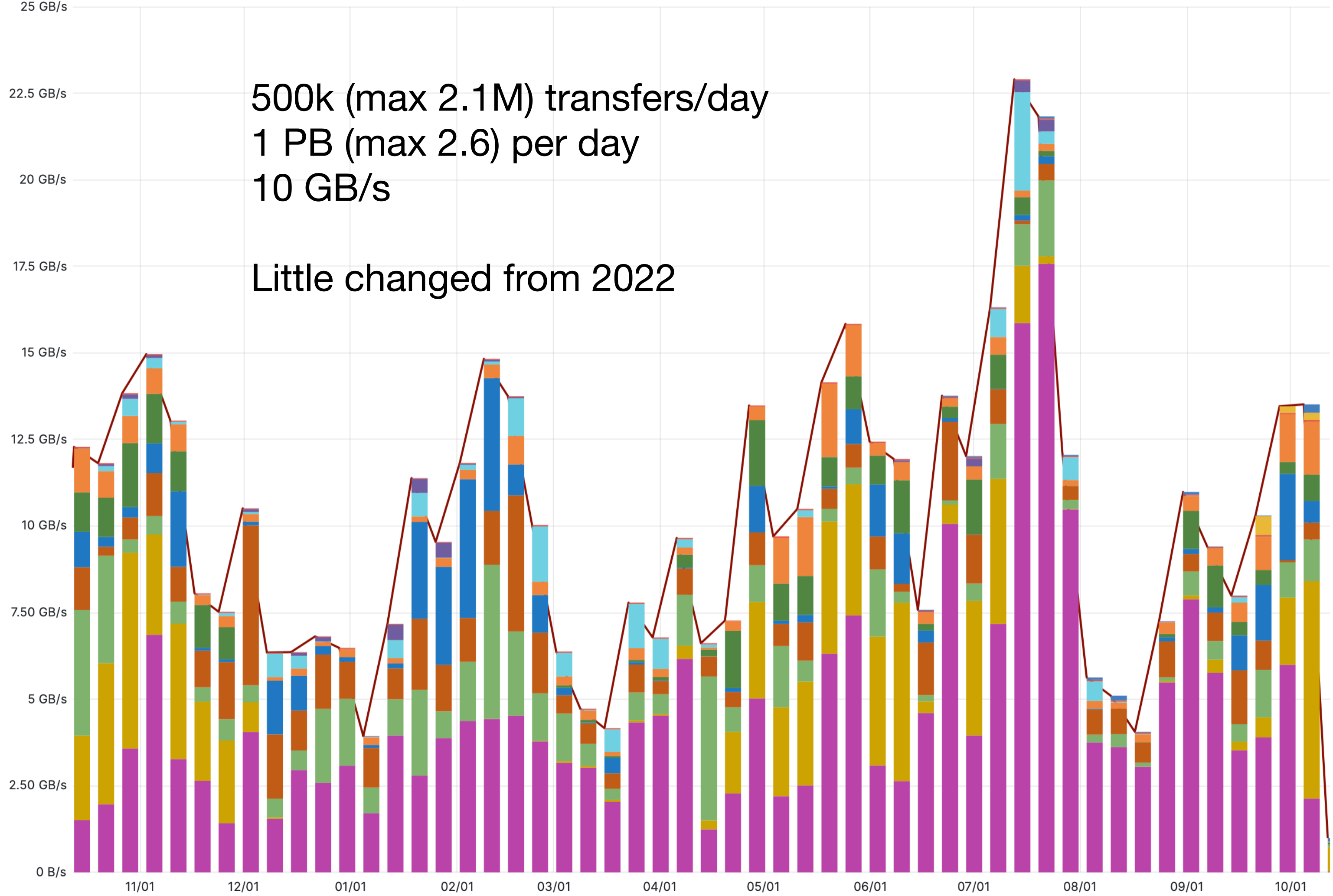
E. Vaandering for the CMS Rucio Team
DIRAC and Rucio Workshop 2023
16-20 October 2023

Outline

- Operations
- Recent developments
- Plan for 2023-2024
- Suggestions

CMS Data Rates

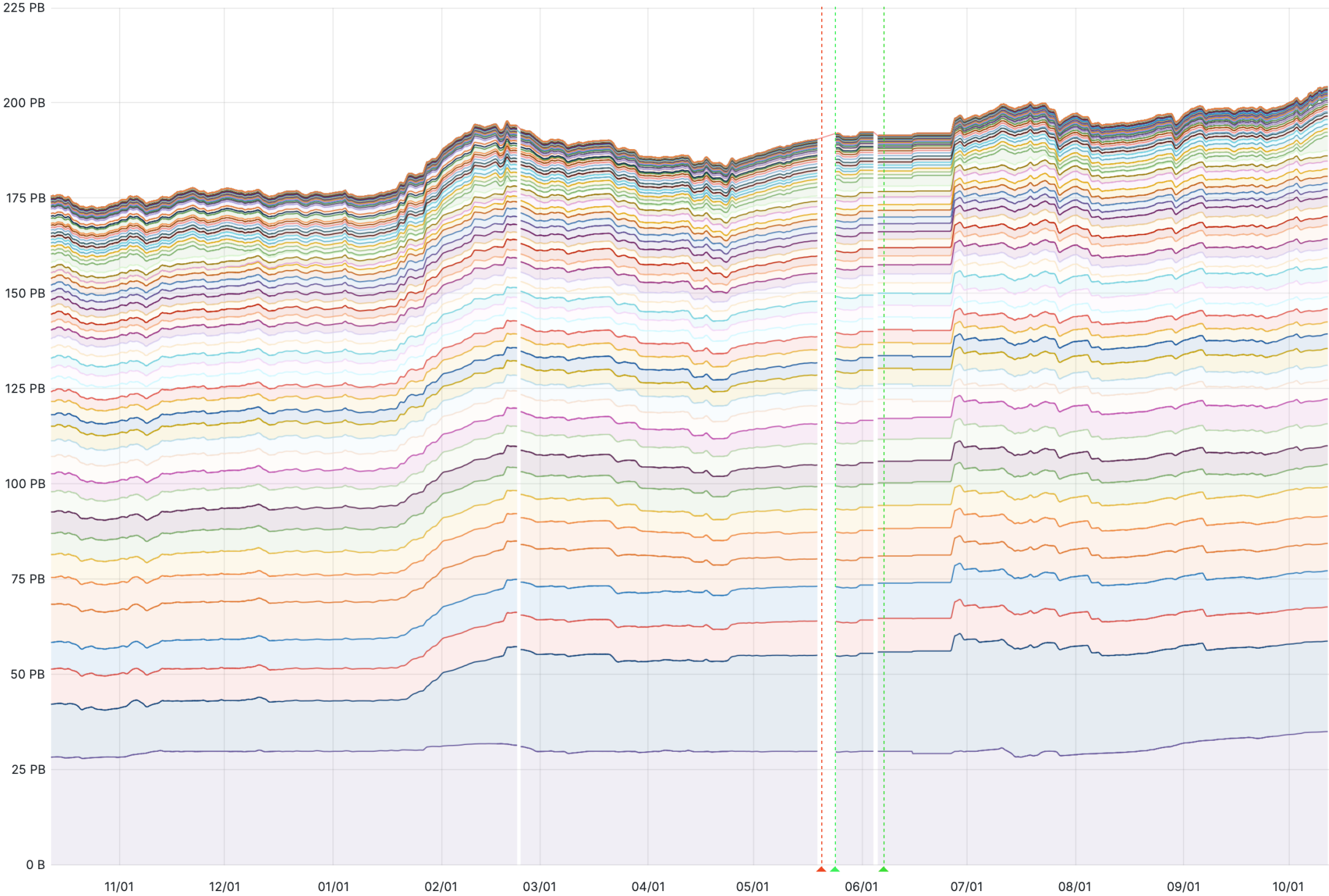
Transfer Throughput



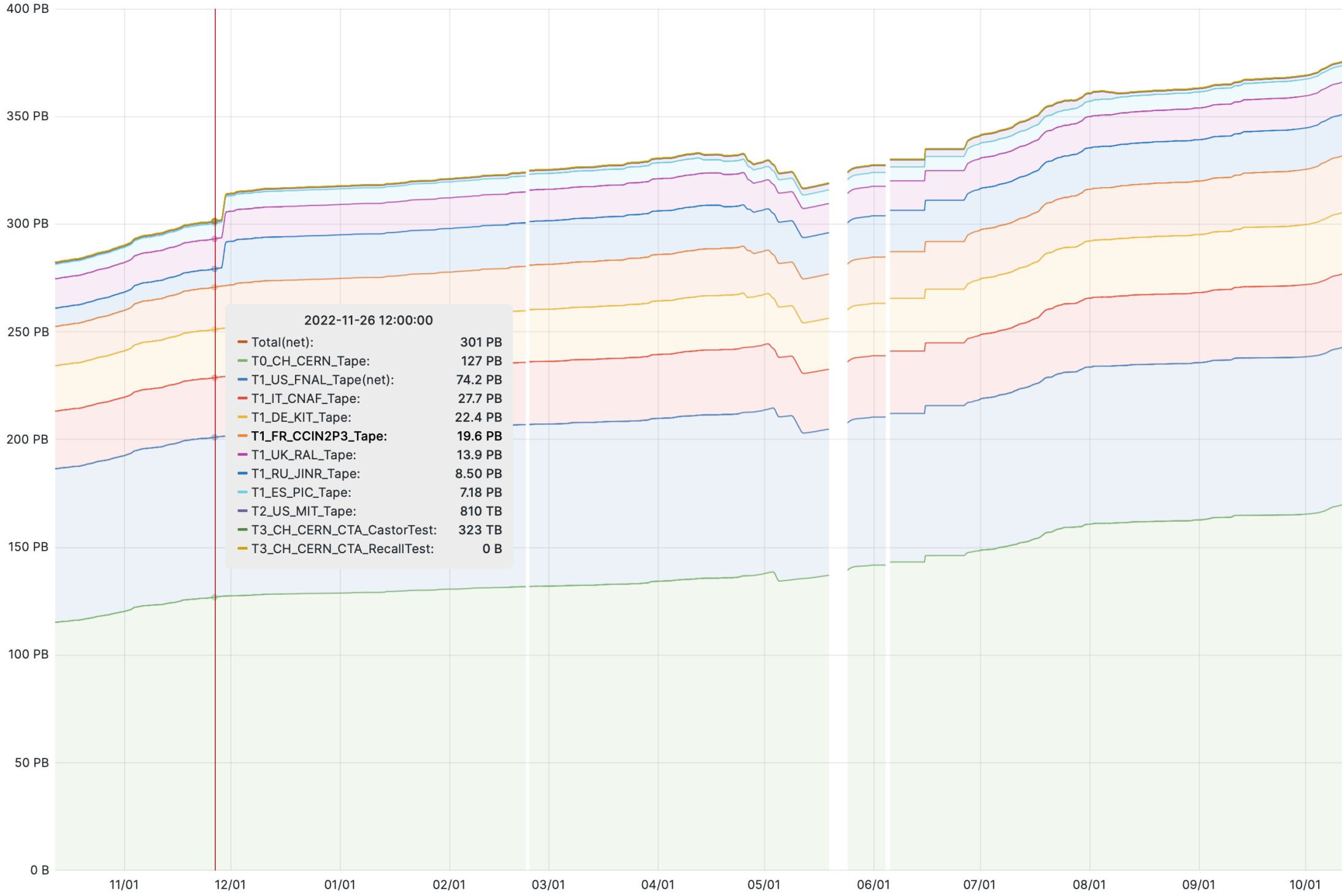
	avg	current	total
Total	9.95 GB/s	990 MB/s	538 GB/s
Production Output	4.37 GB/s	32.4 MB/s	236 GB/s
T0 Tape	1.36 GB/s	685 MB/s	73.3 GB/s
Analysis Input	1.09 GB/s	83.3 MB/s	58.9 GB/s
User Subscriptions	1.03 GB/s	1.49 MB/s	55.7 GB/s
Production Input	689 MB/s	465 kB/s	37.2 GB/s
T0 Export	517 MB/s	62.7 MB/s	27.9 GB/s
Data rebalancing	492 MB/s	22.7 MB/s	26.6 GB/s
Data Consolidation	294 MB/s	449 kB/s	15.9 GB/s
Recovery	55.6 MB/s	12.8 kB/s	3.00 GB/s
Functional Test	29.5 MB/s	4.11 MB/s	1.59 GB/s
Data Challenge	17.7 MB/s	0 B/s	956 MB/s
User AutoApprove	16.2 MB/s	97.5 MB/s	873 MB/s

Space managed over the year

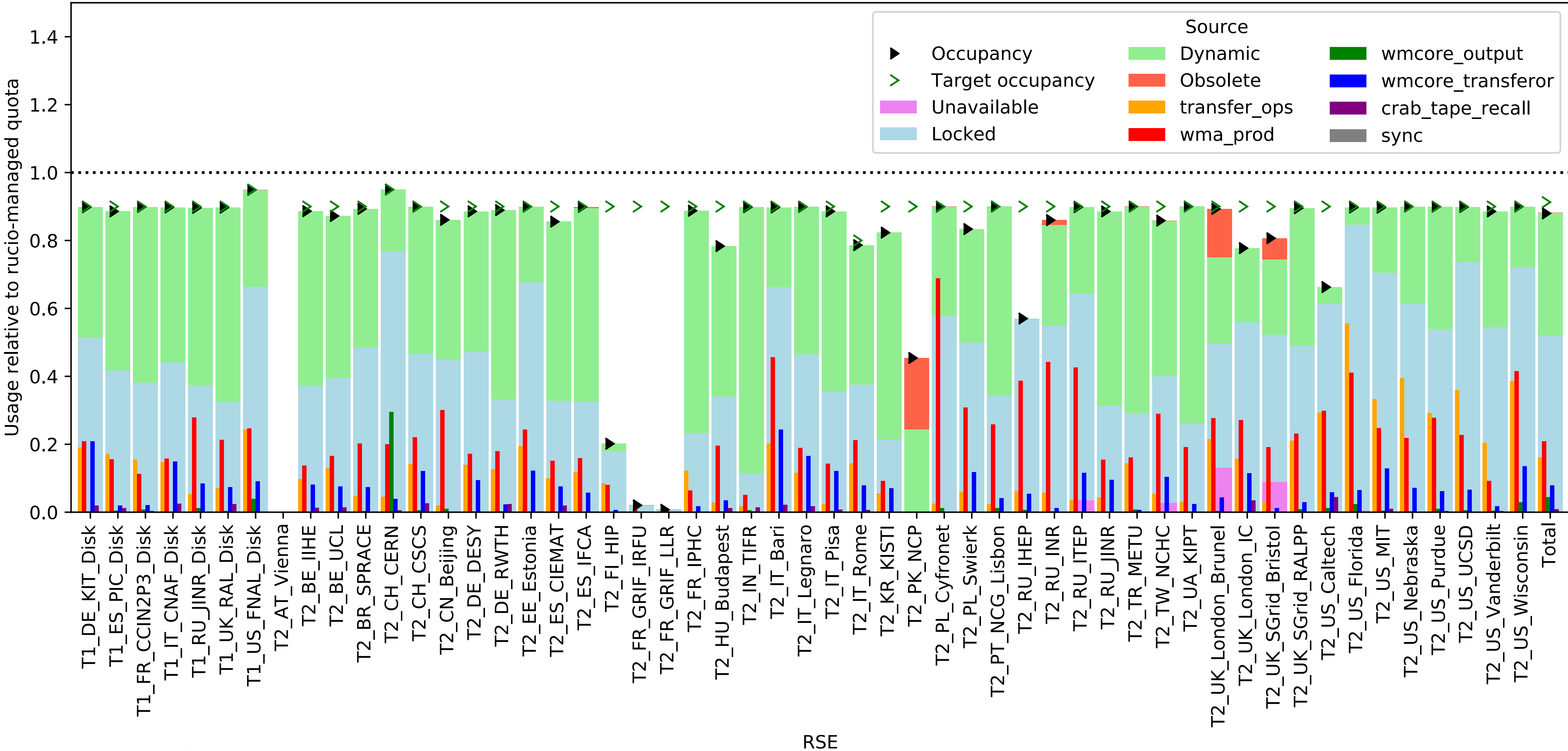
Disk Usage



Tape Usage



Disk Occupancy



2023-10-12 16:01:10 UTC

RSE

Recent Development: Rule approval plugin

- Allow users to be able to create rules with quota shared among all users, not individual quota
- Wanted to restrict rule parameters to configurable parameters.
- Modified rucio rule creation to call the auto_approve plugin.
 - The plugin detects if a algorithm is configured and reads it from the policy package
 - Plugin evaluates rule parameters vs. requirements
- CMS specific policy for auto approval
 - A rule with lifetime 6 months or less
 - Usage when selecting RSE < 50 TB, <1 PB if distributed
 - Allowed usage for all users - 10 PB
- Experience:
 - Users create rules instead of operators (saves time and often lacked expertise to evaluate)
 - We reserve the right to delete rules if need be (nothing bad has happened yet)
- Status: waiting on a PR to Rucio, running in our own patches

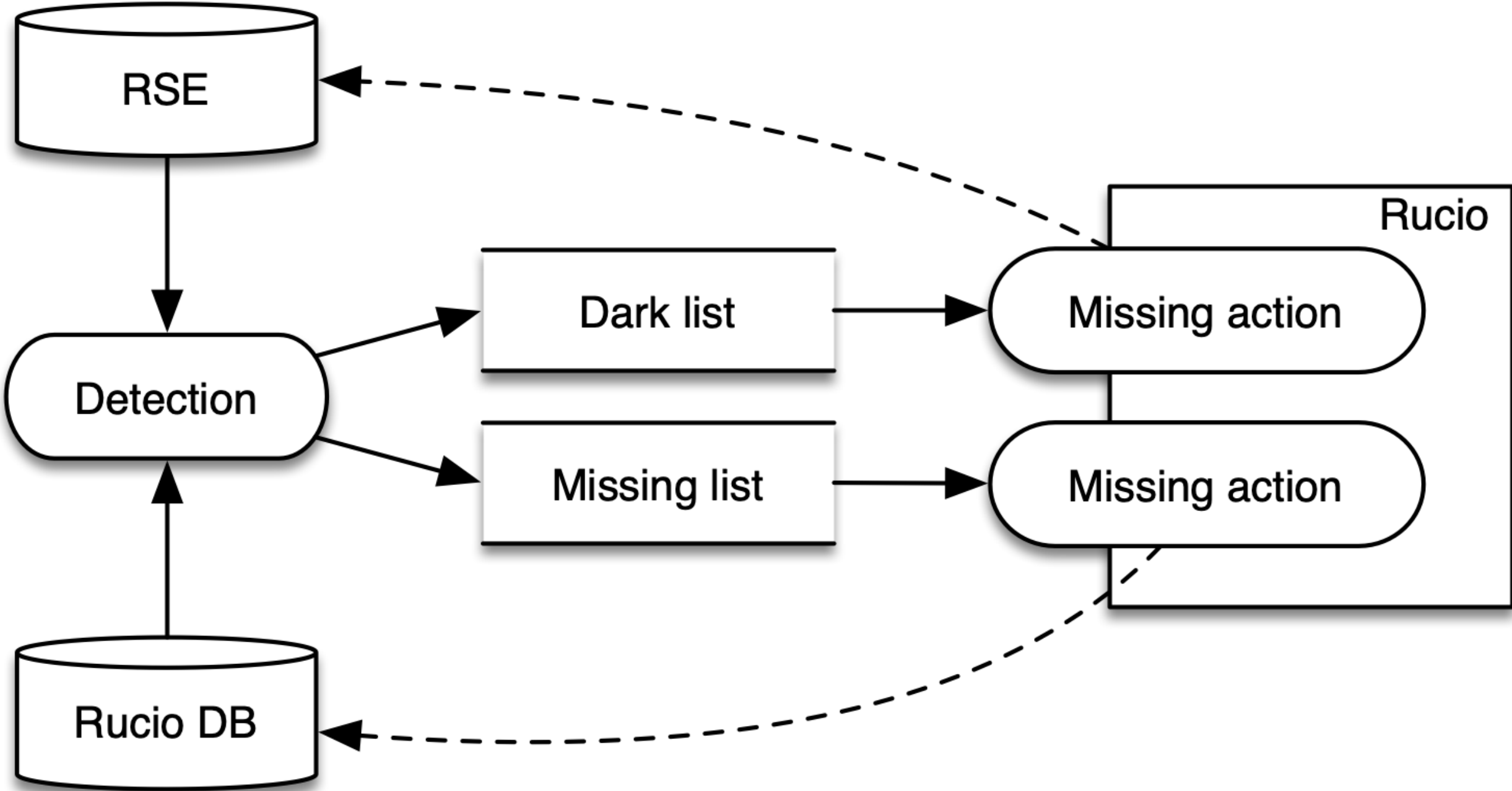
Consistency Enforcement

Inconsistency detection (theoretically):

1. Dump site contents -> list of replica paths
2. Dump Rucio database “replicas” table -> list of paths or LFNs
 - a. Deterministic: scopes, LFNs
 - b. Non-deterministic: scopes, paths
3. Convert paths to LFNs or vice versa
4. Compare site contents to DB contents
 - a. Dark replicas (paths)
 - b. Missing replicas (scopes, names)

Inconsistency correction actions:

1. Declare missing replicas as BAD (scopes, names)
 - a. Rucio will try to re-create replicas
2. Quarantine dark replicas (paths)
 - a. Dark Reaper daemon will remove replicas



Making a common consistency toolbox

We've made a **rucio-consistency** PyPi package

<https://github.com/rucio/consistency-enforcement.git>

Contents

- XRootD scanner
- Partitioned list handling library
- Comparison tools
- Rucio database dump tool

CMS specific parts glue this together

https://github.com/ivmfnal/cms_consistency

<https://github.com/dmwm/CMSRucio.git>

- Kubernetes infrastructure/deployment (helm chart)
- Web monitor
- Historic/current data storage
- CE run driver script (runs under jobber 1/week/RSE)

User produced data into Rucio

- We're a little late to this, but...
- Our users can write data now, but only to SE where they are allowed
 - Cannot be moved, duplicated, or easily deleted
- Implementation:
 - With user credentials write to an area at local site where they have permissions
 - Register this PFN and DID into a non-deterministic RSE
 - DID is scope user.USER and LFN /store/user/rucio/USER to avoid collisions with other data
 - Rucio rules move this data to our usual deterministic RSEs
- Some hope to remove our existing direct-to-FTS transfer tool for user data

Future development plans

Trying to plan things out for next 3-6 months

- Improve documentation for users and operators (leverage Rucio docs when possible)
- Training/tutorials to be worked out. Perhaps a place to collaborate
- Improve monitoring and its usage of Kubernetes infrastructure
 - Also logging
- Continued evolution of Rucio permission for smooth operation
- Tokens
 - OIDC authentication for users. In place now, but need to migrate and regularly test
 - OIDC authentication for deletion as Rucio developments allow

Future development plans, cont

- Do some automatic suspicious replica recovery
 - I know we talked before about source replica errors, need to resurrect
- Data rebalancing among sites
- Metadata for files on tape co-location
 - Can we help with the core Rucio effort on this?
- Consolidate monitoring on Prometheus
 - Including probes, which requires some unexciting development
- Upgrade CMS specific images (client, loadtest, traces, consistency) to Alma9

Technical suggestions

- Can we know who deletes a rule? Add to Rule History?
- Are there changes coming to how user authentication with OIDC is being done?
- Can we make the behavior of lost replicas optional?
 - Currently they are detached from their dataset
 - This causes inconsistencies between Rucio and our physics metadata system

Process Suggestions

- Pull requests not in the release plan take a lot of time/bugging to get approved
 - Can we improve this with some sort of nomination process to get them into the currently active release?
- Is there interest in a new repository to share dashboard definitions?
 - Probably with a sample screenshot too
 - And a description of what to look for, how/where data is generated
- Many of us are doing the same things