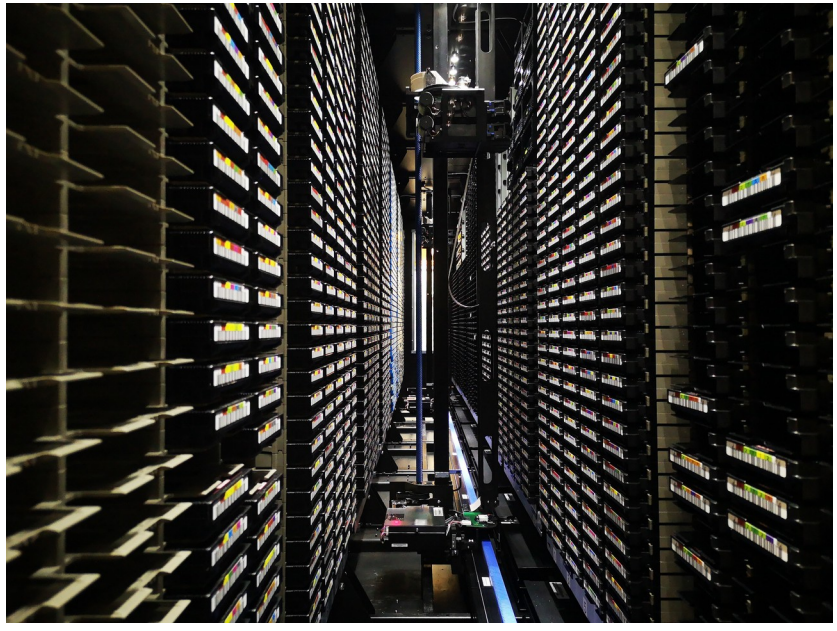
Tape evolution with dCache

Tigran Mkrtchyan for dCache team

pre-GDB



DESY.

Fermilab

neic
NORDIC E-INFRASTRUCTURE COLLABORATION

HELMHOLTZ RESEARCH FOR GRAND CHALLENGES

dCache Tape Connectivity

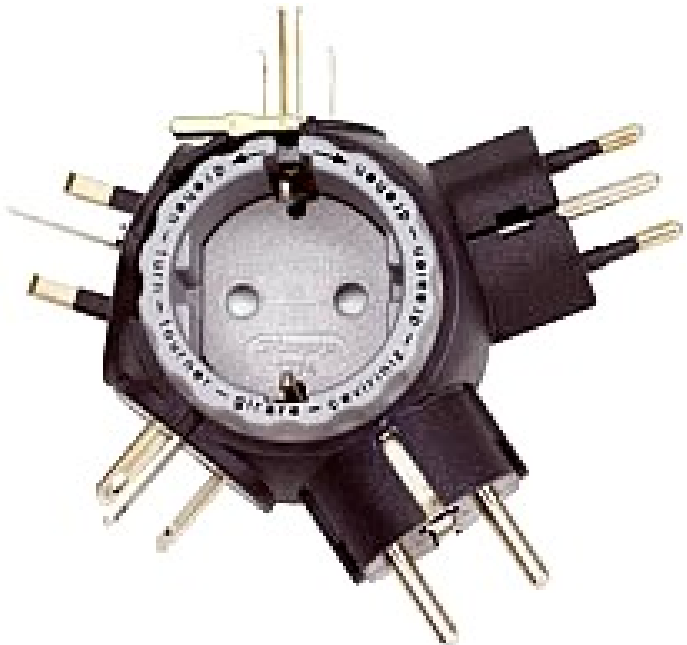
- Native to dCache
  - dCache == disk cache on front of tape
  - The essential part of the dCache design
  - Provides fill functionality with/without HSM
  - Tape and disk files can be mixed on a single data server
- Write-back / Read-through cache behavior
  - Transparent for the users
  - Available via all protocols (subject to authorization)
- Stage Protection
  - user/VO/storage class/protocol
- Supports multiple HSM on a single instance
- Stores tape location as opaque data provided by HSM

Interfaces to HSM

- Execute external migration script
  - Stupid, Simple, Genius …
  - Reference implementation of driver API
- Plugable driver Java API:
  - Suitable to create efficient HSM connectivity
  - ENDIT *(Efficient Northern dCache Interface to TSM)*

Stupid, Simple – Genius!

- Works with any thing that cans store data
  - OSM
  - EnSTORE
  - HPSS
  - TSM
  - SGI DMF
  - Amazon S3
  - dCacache



dCache ⟺ HSM Link

- Files belong to storage classes & and HSM type

```
<storage-class>@osm
```

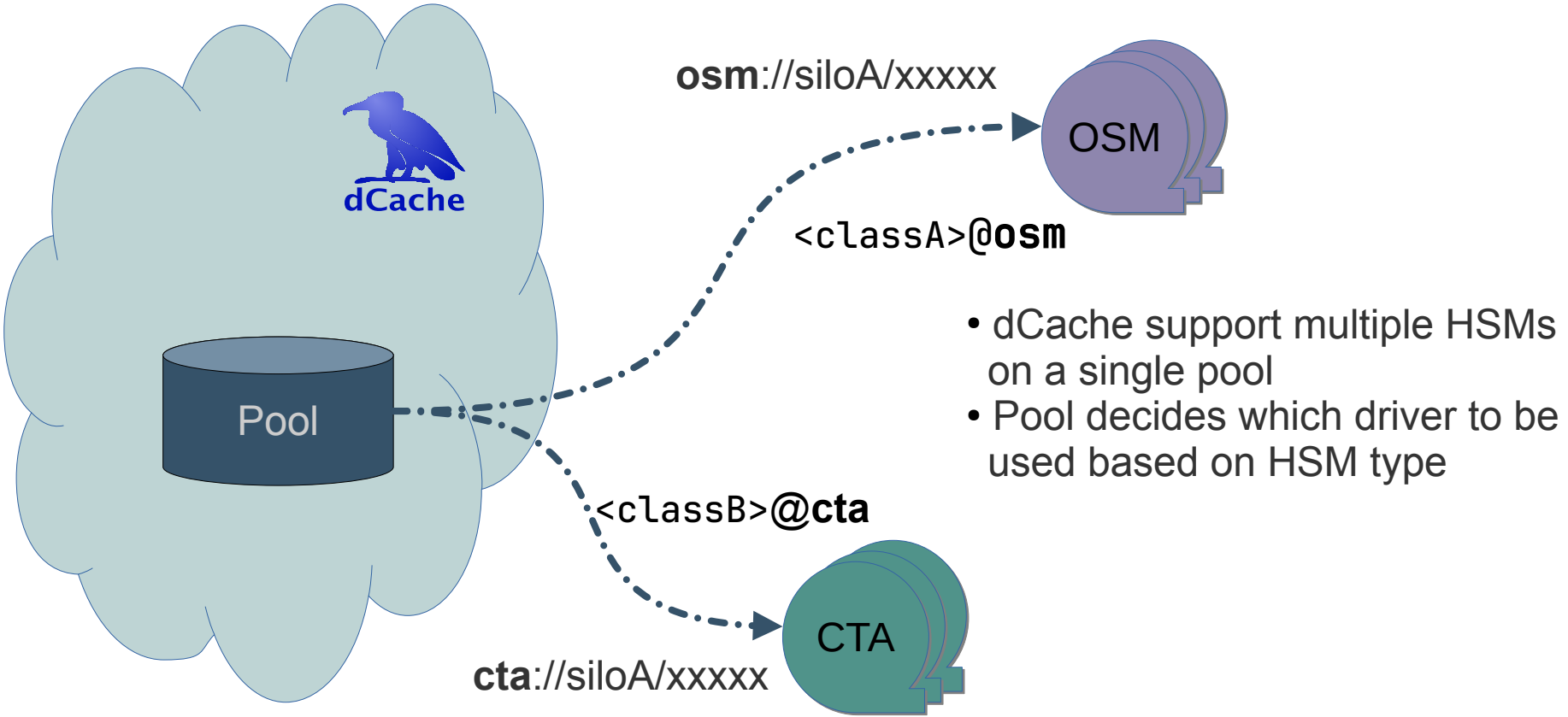
- Pools configured to interact with HSMs

```
hsm create osm siloA script \
                    -command=hsmcp.py
```

- Stored file have information about HSM type and hsm specific identifier

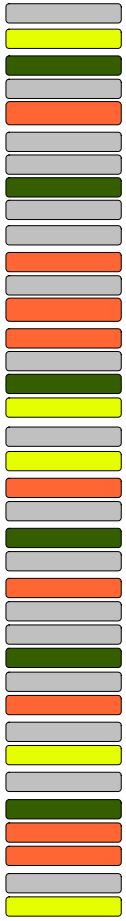
```
osm://siloA/xxxxxxxxxxx
```

Pool ⟺ HSM Connectivity

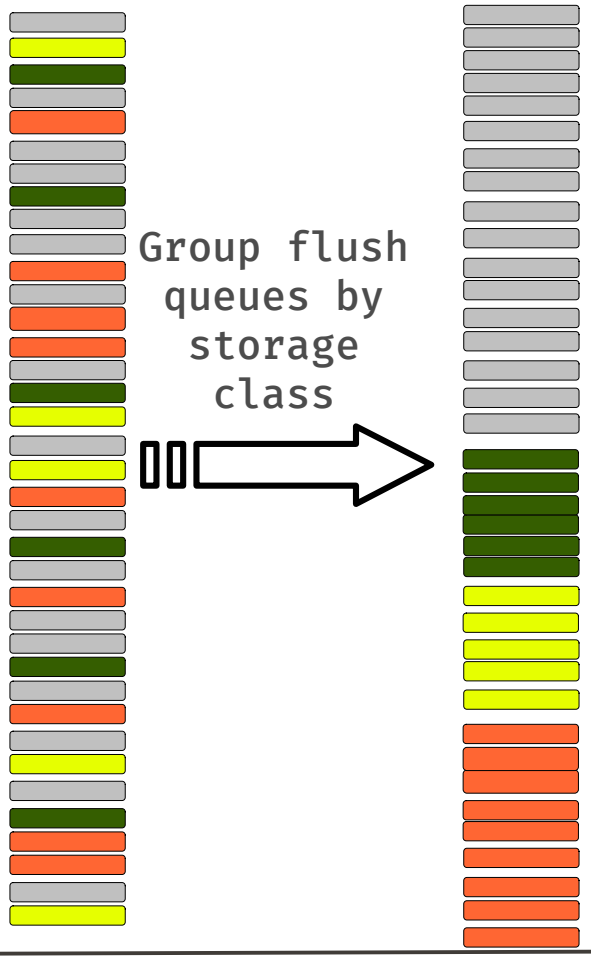


- dCache support multiple HSMs on a single pool
- Pool decides which driver to be used based on HSM type

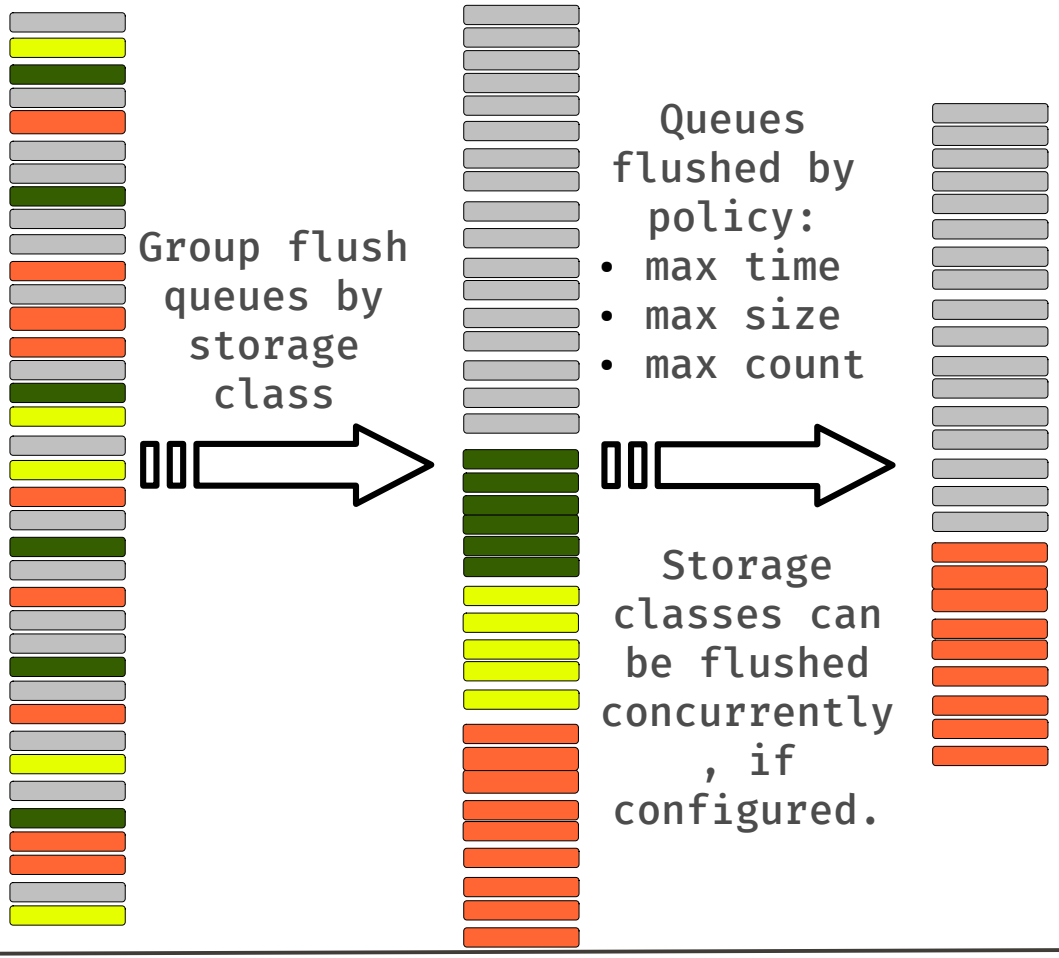
The Flush Queue



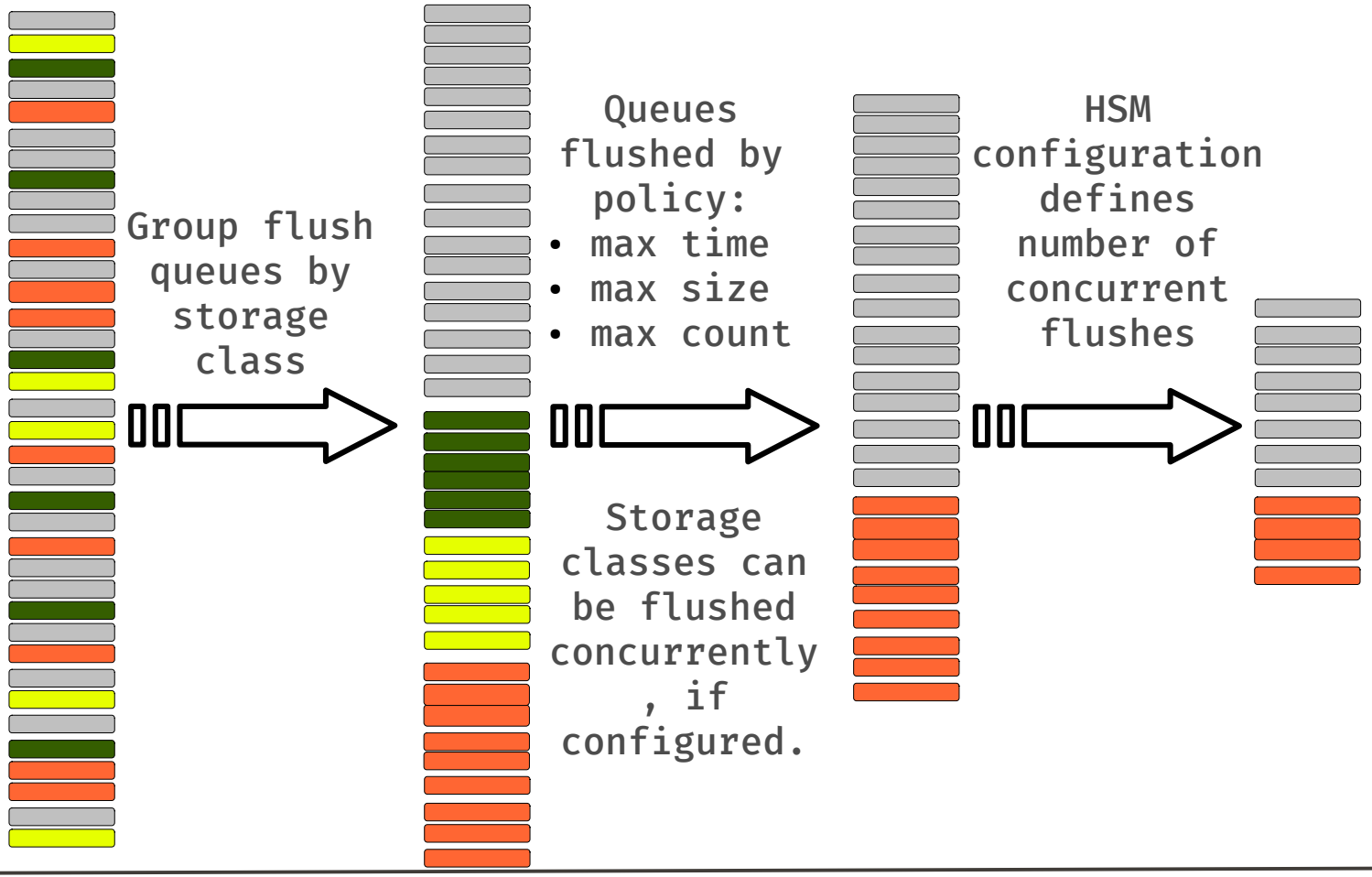
The Flush Queue



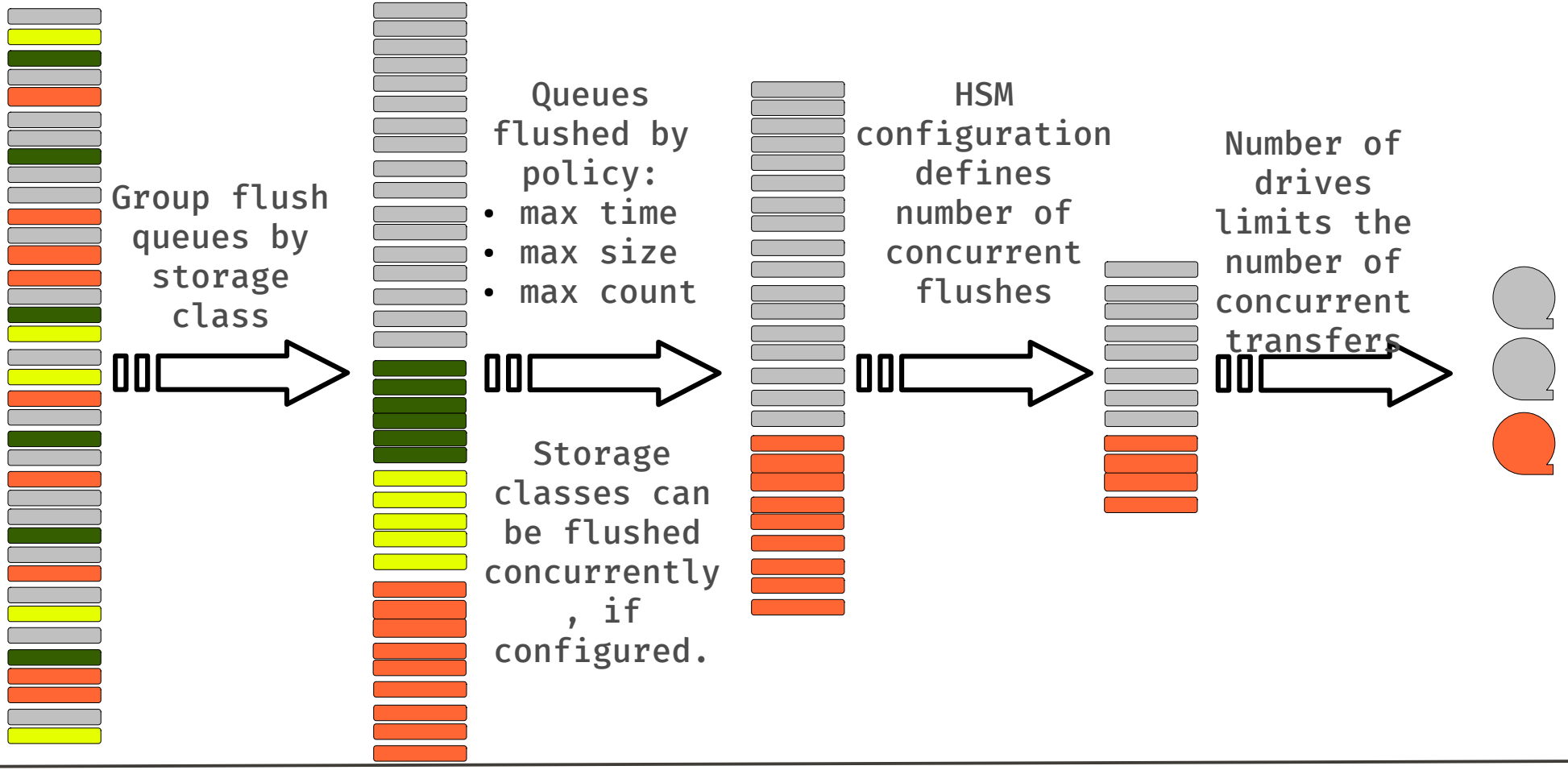
The Flush Queue



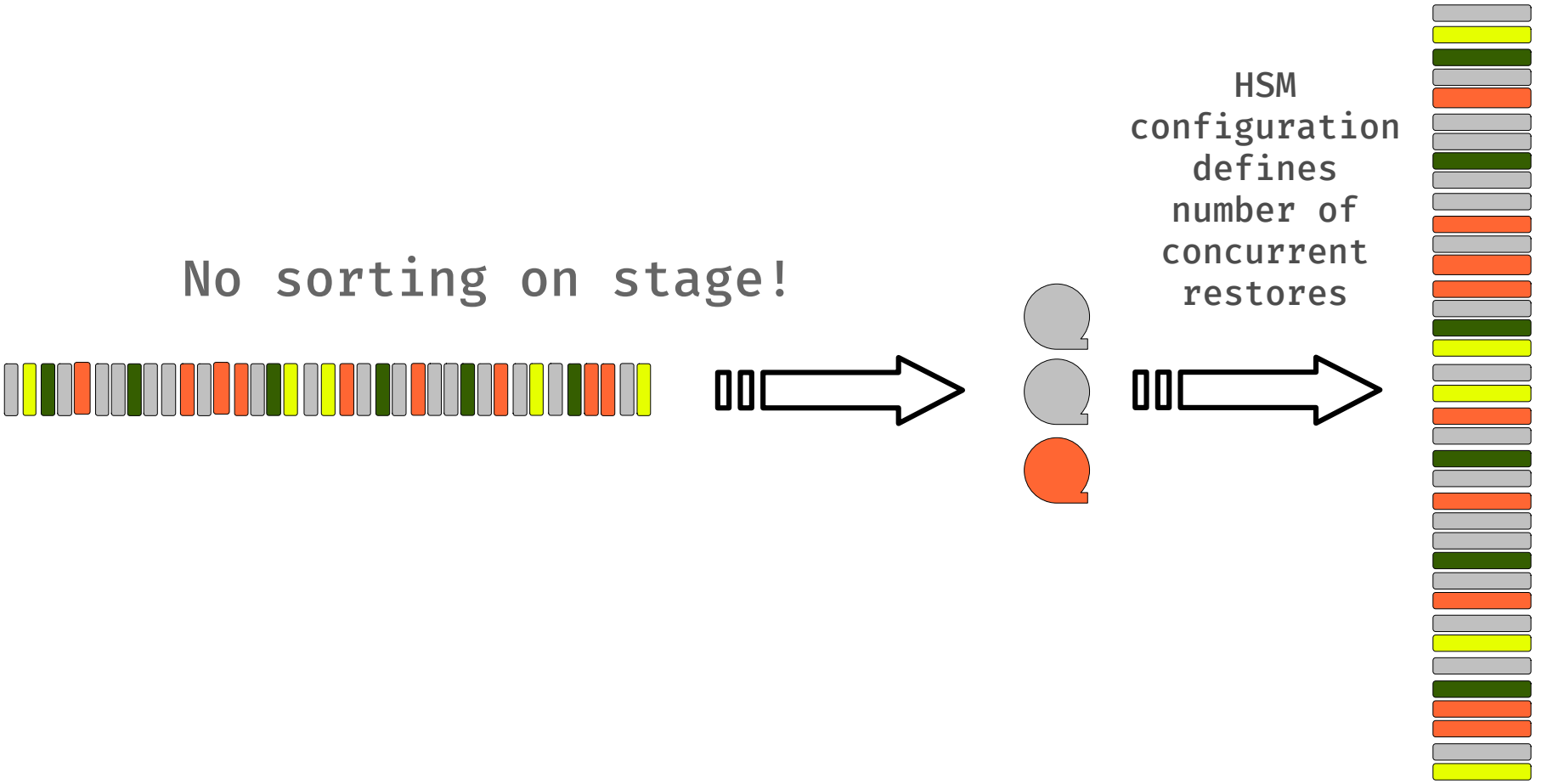
The Flush Queue



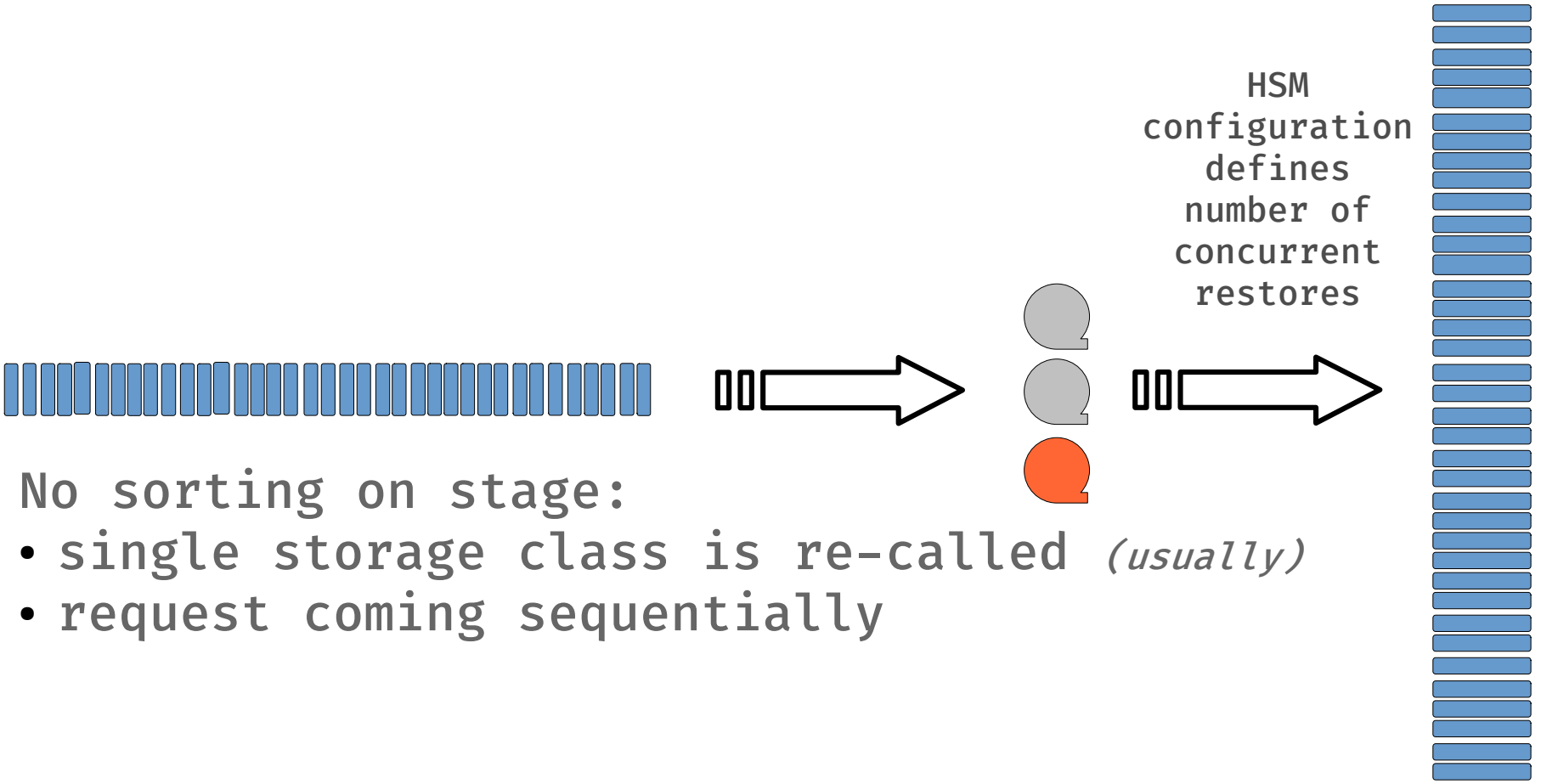
The Flush Queue



The Restore Queue



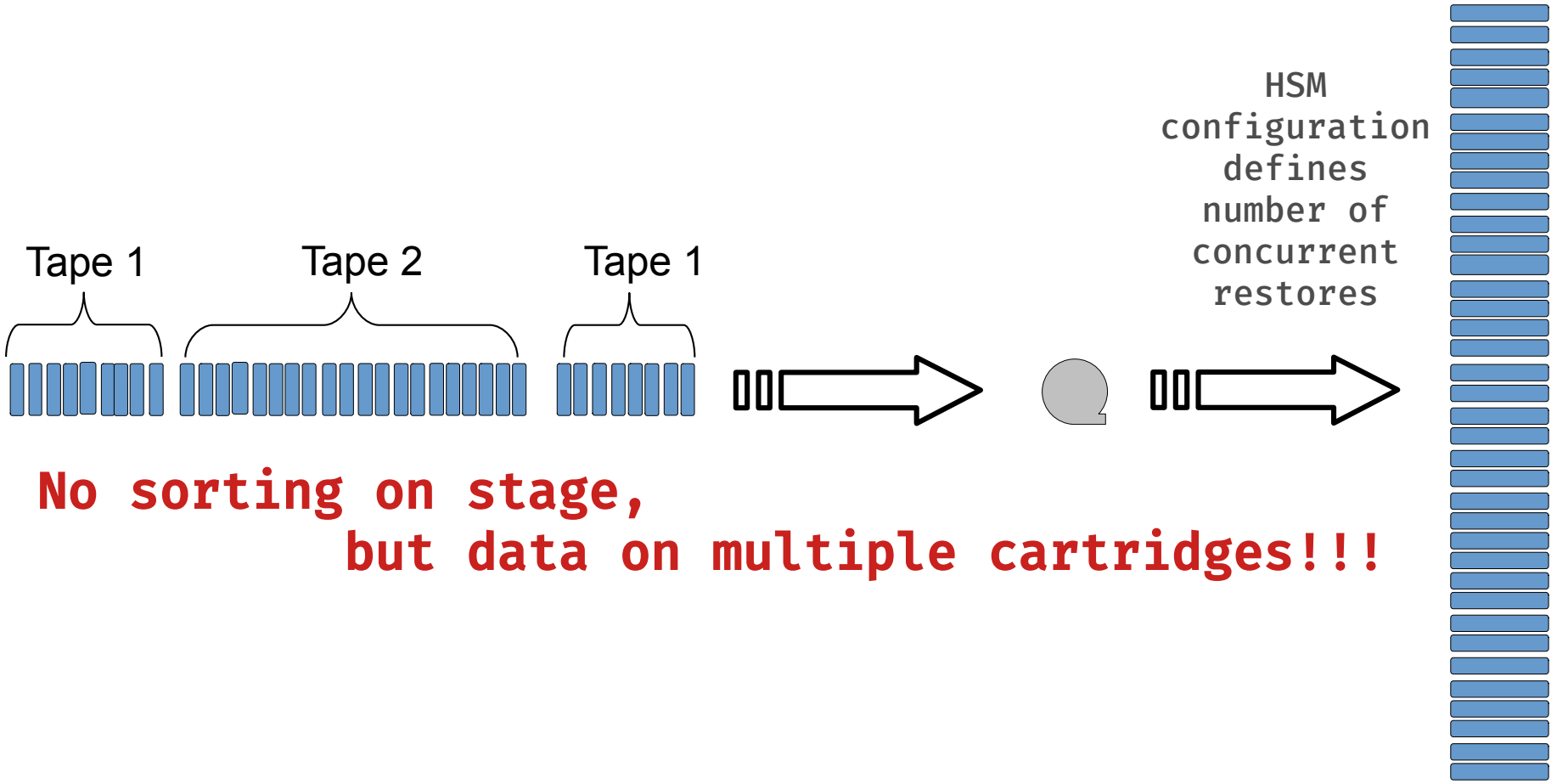
The Restore Queue



No sorting on stage:

- single storage class is re-called *(usually)*
- request coming sequentially

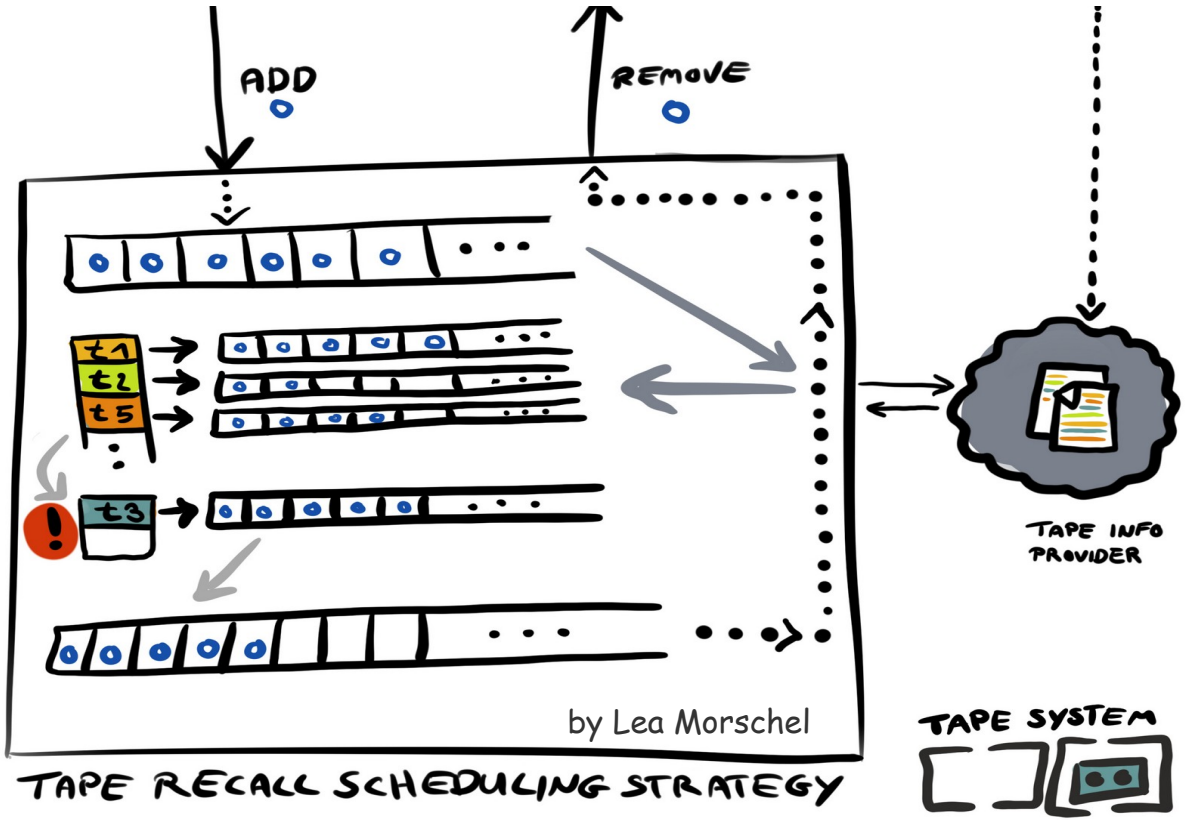
The Restore Queue



No sorting on stage,
but data on multiple cartridges!!!

Tape Recall Grouping

- Group recalls by tapes
- Recall triggered by:
  - size
  - max waiting time
- Number of parallel recalls based on number of tape drives

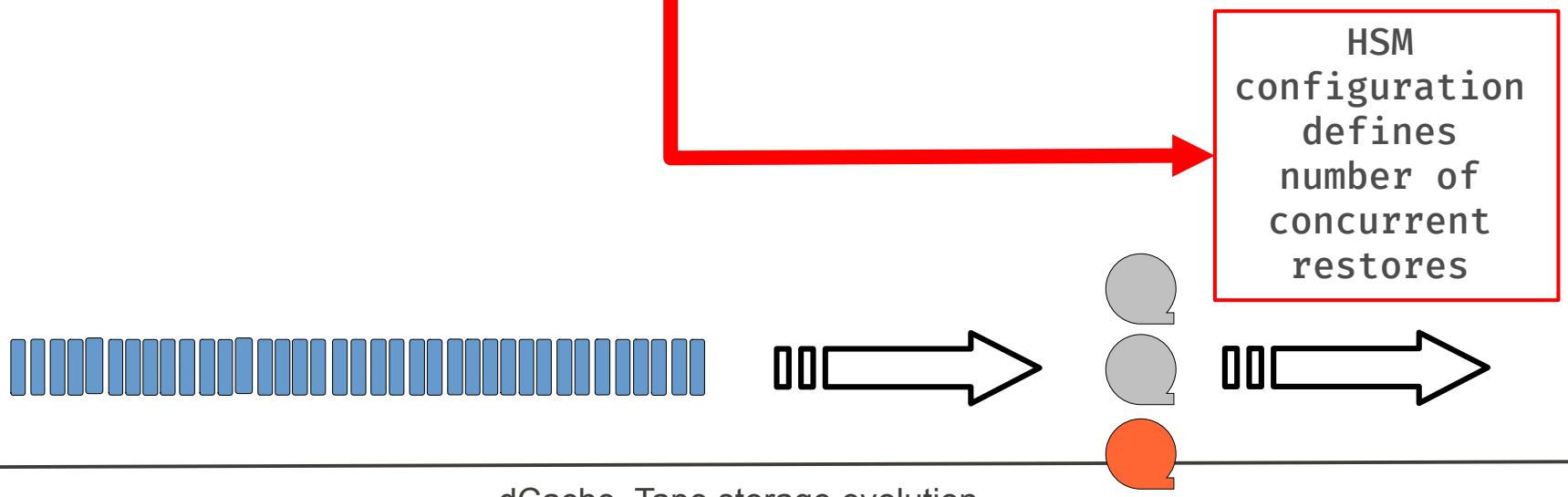


pre-Restore Space Allocation

- **Never block on space allocation when tape is mounted!**
- Space allocated only for requests sent to HSM

pre-Restore Space Allocation

- **Never block on space allocation when tape is mounted!**
- Space allocated only for requests sent to HSM

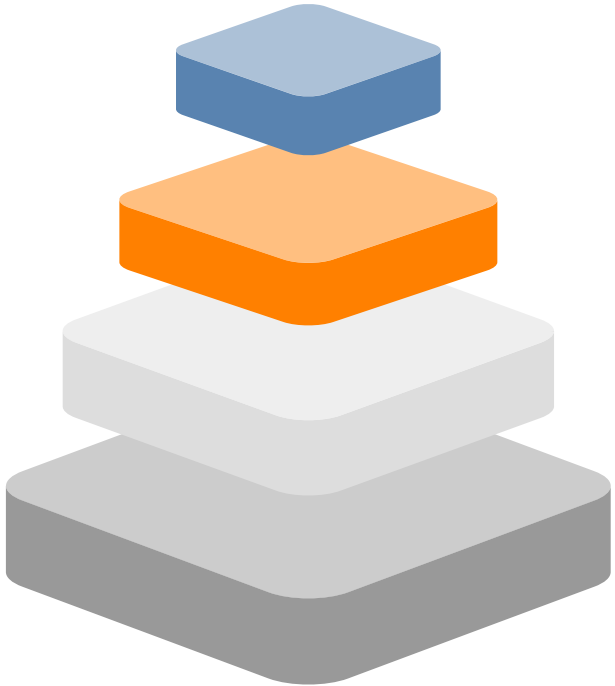


Lazy Space Allocation (ENDIT 2.0?)

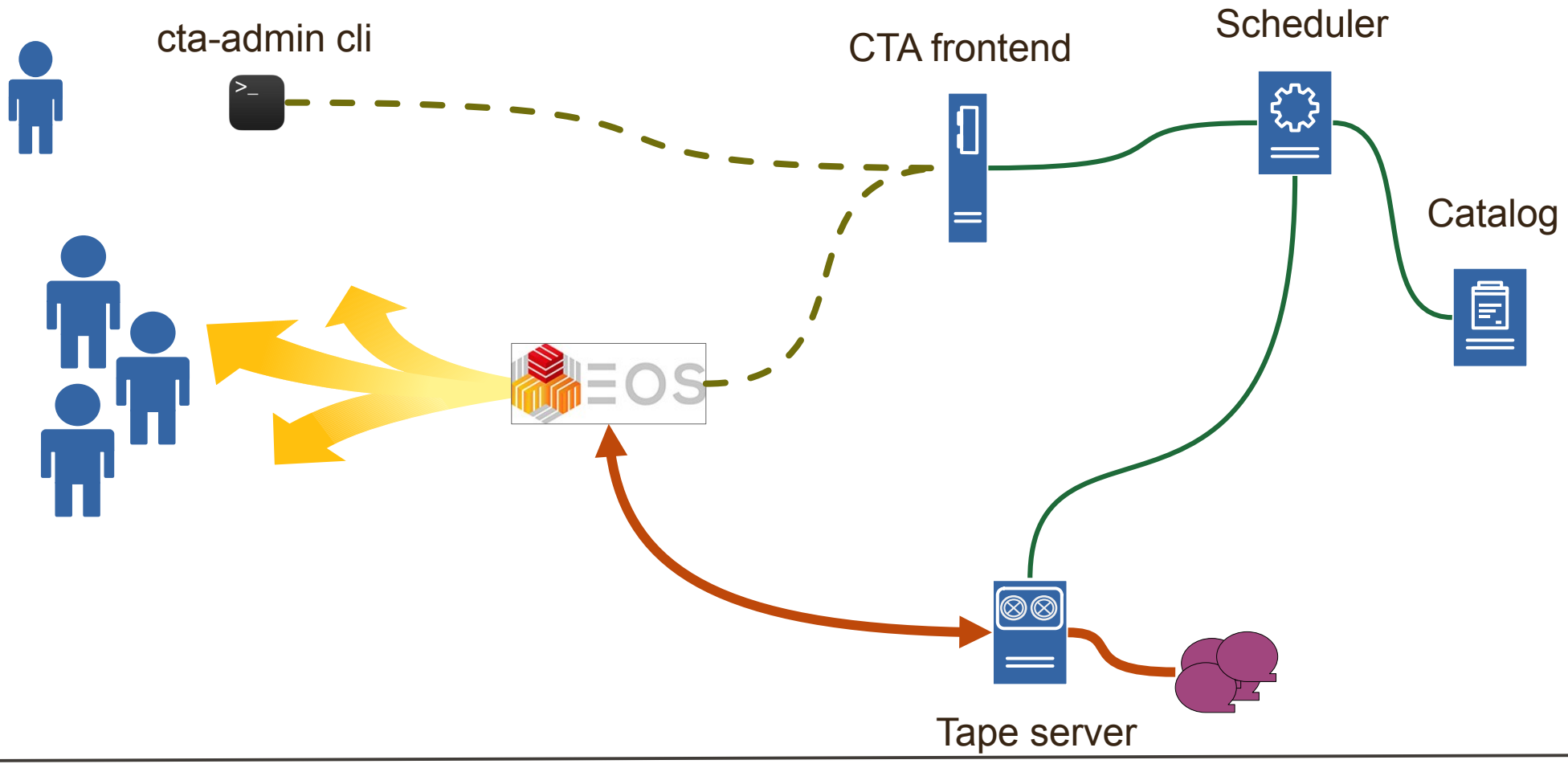
- **Do (all must apply)**
  - Plugin is used
  - Tape system has internal disk
  - HSM tolerates ‘blocked on allocation’
- **Don’t (if any is true)**
  - Script is used
  - Drive directly writes to pool
  - ‘blocked on allocation’ blocks HSM (or Drive)

Tape Aware Components

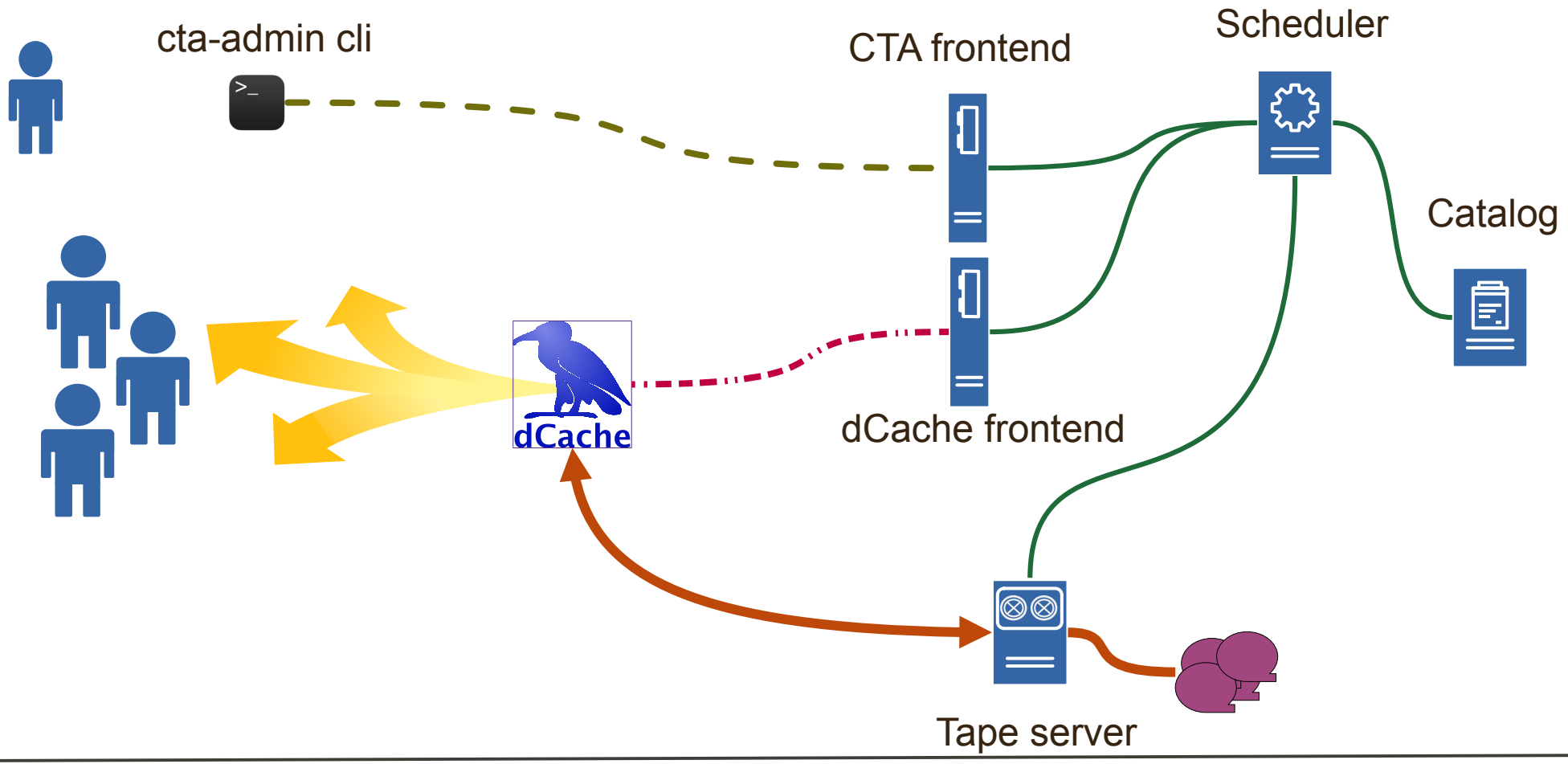
- SRM, TAPE-API (BULK)
  - Accept request from users and forwards to PinManager
  - Keeps state in local DB for reporting
- Pin Manager
  - Squashes multiple requests
  - Uses Pool Manager to stage the file
  - Keeps state in local DB to handle unpin and restarts
- Pool Manager
  - Selects appropriate pool for stage
  - Retries on errors
- Pool
  - Allocates space on the pool
  - Forwards the request to the tape system



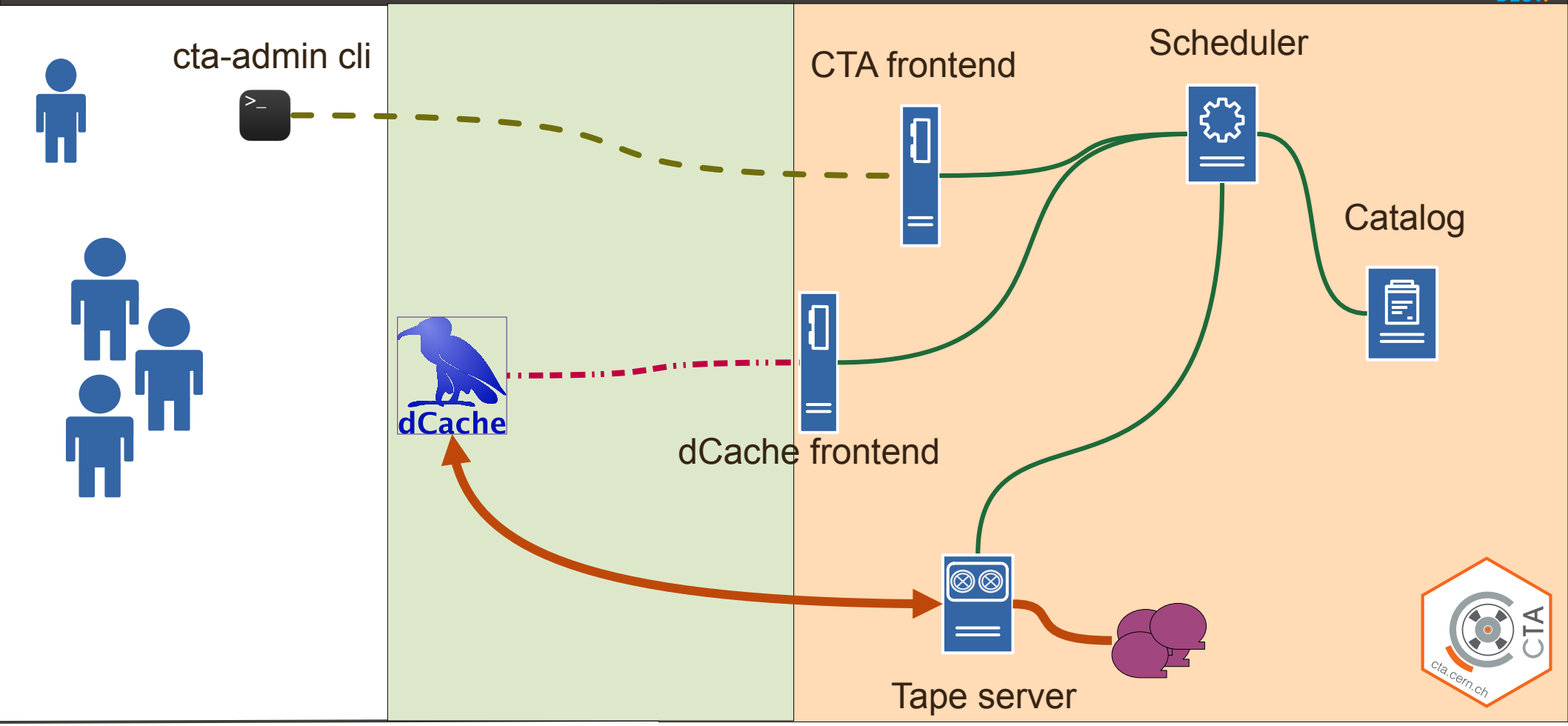
(Extremely) Simplified CTA design



(Extremely) Simplified CTA design



Deployment at DESY



dCache+CTA Status

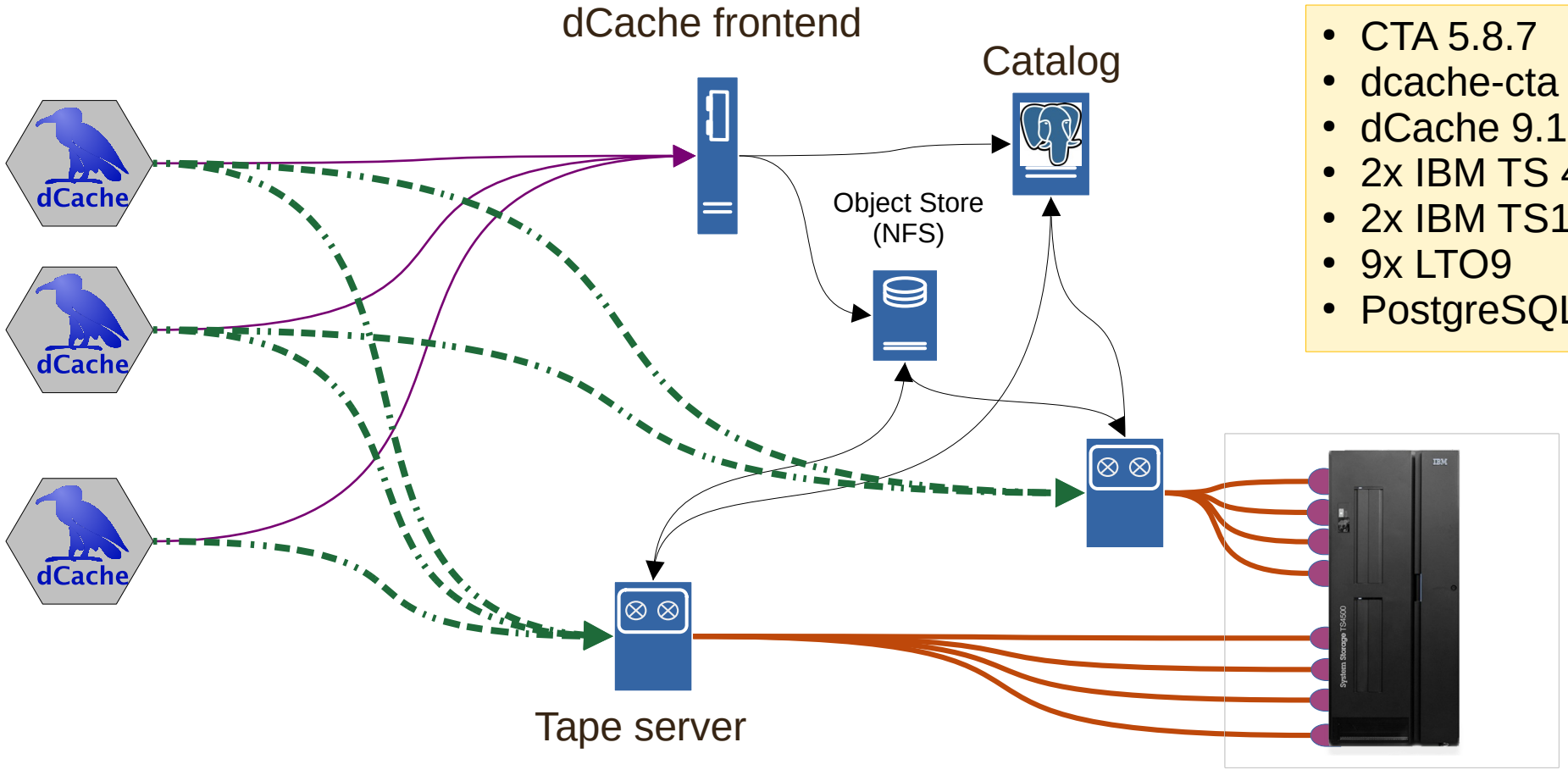
- Seamless integration with dCache is merged into upstream CTA code at CERN
  - Starting CTA release {4,5}.7.12
- The existing ENSTORE/OSM tape format is supported for READ
  - The ENSTORE/OSM tape catalog conversion procedures are successfully tested at DESY and Fermilab.
    - dCache+OSM → dCache+CTA
    - dCache+EnSTORE → dCache+CTA
- dCache+CTA is deployed at DESY for BELLE-II, EuXFEL
  - ~2PB/week (3.4 GB/s, 9 drives)
- dCache+CTA deployment replicate to by other HEP sites
  - Fermilab and PIC Barcelona have successfully replicated DESY setup (currently dCache + ENSTORE).

Setup at DESY (Duct Tape & WD40)

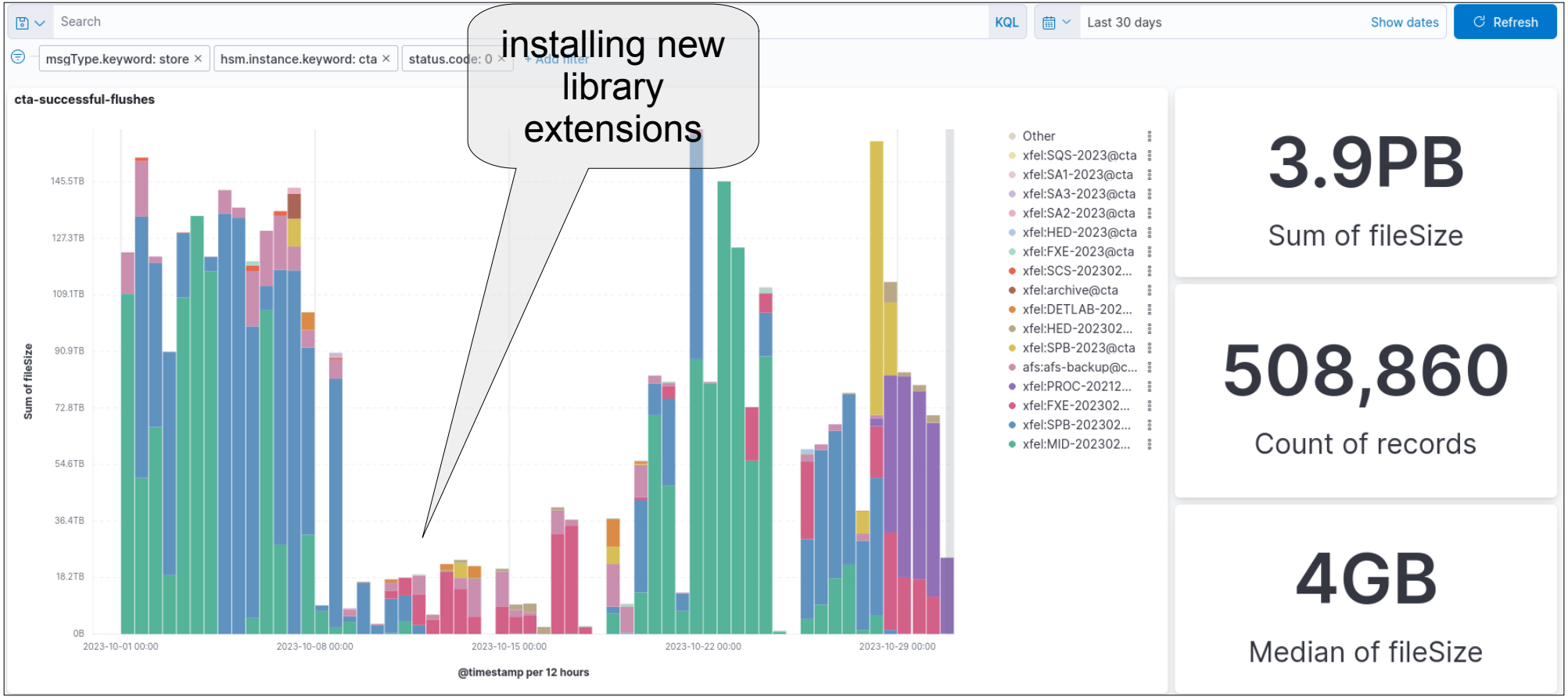
- PostgreSQL as DB
  - Long experience with dCache and OSM
- NetApp NFSv4.1 volume as ObjectStore
  - Little to no expertise in CEPH
- Four drives per tape server
  - Operational mode of OSM
  - (Almost) No issues observes
- Dedicated node (taped) for the maintenance task
  - Much better stability

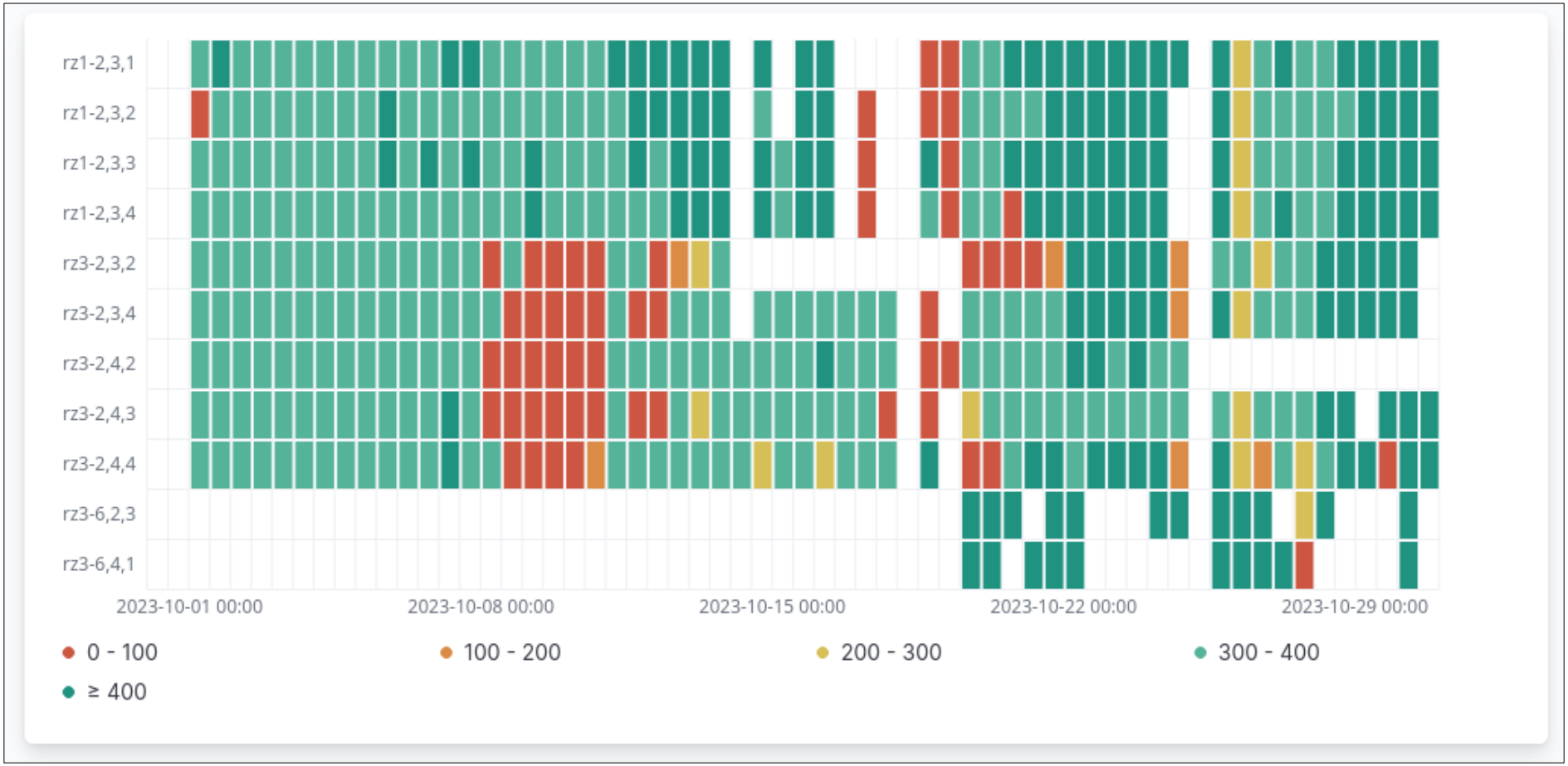
by Norm Wright

Production Deployment at DESY



- CTA 5.8.7
- dcache-cta 0.10.0
- dCache 9.1.6
- 2x IBM TS 4500
- 2x IBM TS1160
- 9x LTO9
- PostgreSQL 14.1


Search
KQL
Last 30 days
Show dates
Refresh
msgType.keyword: store
hsm.instance.keyword: cta
status.code: 0
installing new library extensions
cta-successful-flushes
Sum of fileSize
145.5TB
127.3TB
109.1TB
90.9TB
72.8TB
54.6TB
36.4TB
18.2TB
0B
2023-10-01 00:00
2023-10-08 00:00
2023-10-15 00:00
2023-10-22 00:00
2023-10-29 00:00
@timestamp per 12 hours
Other
xfel:SQS-2023@cta
xfel:SA1-2023@cta
xfel:SA3-2023@cta
xfel:SA2-2023@cta
xfel:HED-2023@cta
xfel:FXE-2023@cta
xfel:SCS-202302...
xfel:archive@cta
xfel:DETLAB-202...
xfel:HED-202302...
xfel:SPB-2023@cta
afs:afs-backup@c...
xfel:PROC-20212...
xfel:FXE-202302...
xfel:SPB-202302...
xfel:MID-202302...
3.9PB
Sum of fileSize
508,860
Count of records
4GB
Median of fileSize


rz1-2,3,1
rz1-2,3,2
rz1-2,3,3
rz1-2,3,4
rz3-2,3,2
rz3-2,3,4
rz3-2,4,2
rz3-2,4,3
rz3-2,4,4
rz3-6,2,3
rz3-6,4,1
2023-10-01 00:00
2023-10-08 00:00
2023-10-15 00:00
2023-10-22 00:00
2023-10-29 00:00
0 - 100
100 - 200
200 - 300
300 - 400
≥ 400

Future Work

- Tape DDoS protection
  - Some experiment workflows are not tape friendly
- True ‘stage buffer’
  - Mark transferred files as best candidates for eviction
- Migrating Tape scheduler to PinManager
- Deeper integration with CTA
  - Handle pool restarts
  - Process requests by creation time

Prominent Changes in Tape Interface

- Bulk cancel of store/restore requests
- Update driver-based HSM (ENDIT, dcache-cta) connectivity without restart
- Propagate file creation time/path/xattr to HSM driver
- Improved error handling, scalability, configuration

Summary

- Tape operation is an essential part of dCache design (dCache = disk cache on front of tape)
- All dCache development sites (DESY, FNAL, NeIC) depend on tape connectivity and constantly improving it functionality
- DESY and FNAL teams work on dCache⟺CTA integration for and migration of existing systems
- Despite dCache, tape is a non-shareable resource, therefore, it should be used wisely

Thank You!

More info:

https://dcache.org

To steal and contribute:

https://github.com/dCache/dcache

https://github.com/dCache/dcache-cta

Help and support:

support@dcache.org, user-forum@dcache.org

Multiple Faces of Tape

At Tier-0

- High data ingest rate
- Multiple parallel streams
- High durability, multiple copies on different media
- Long-term nearline access
- Small file handling

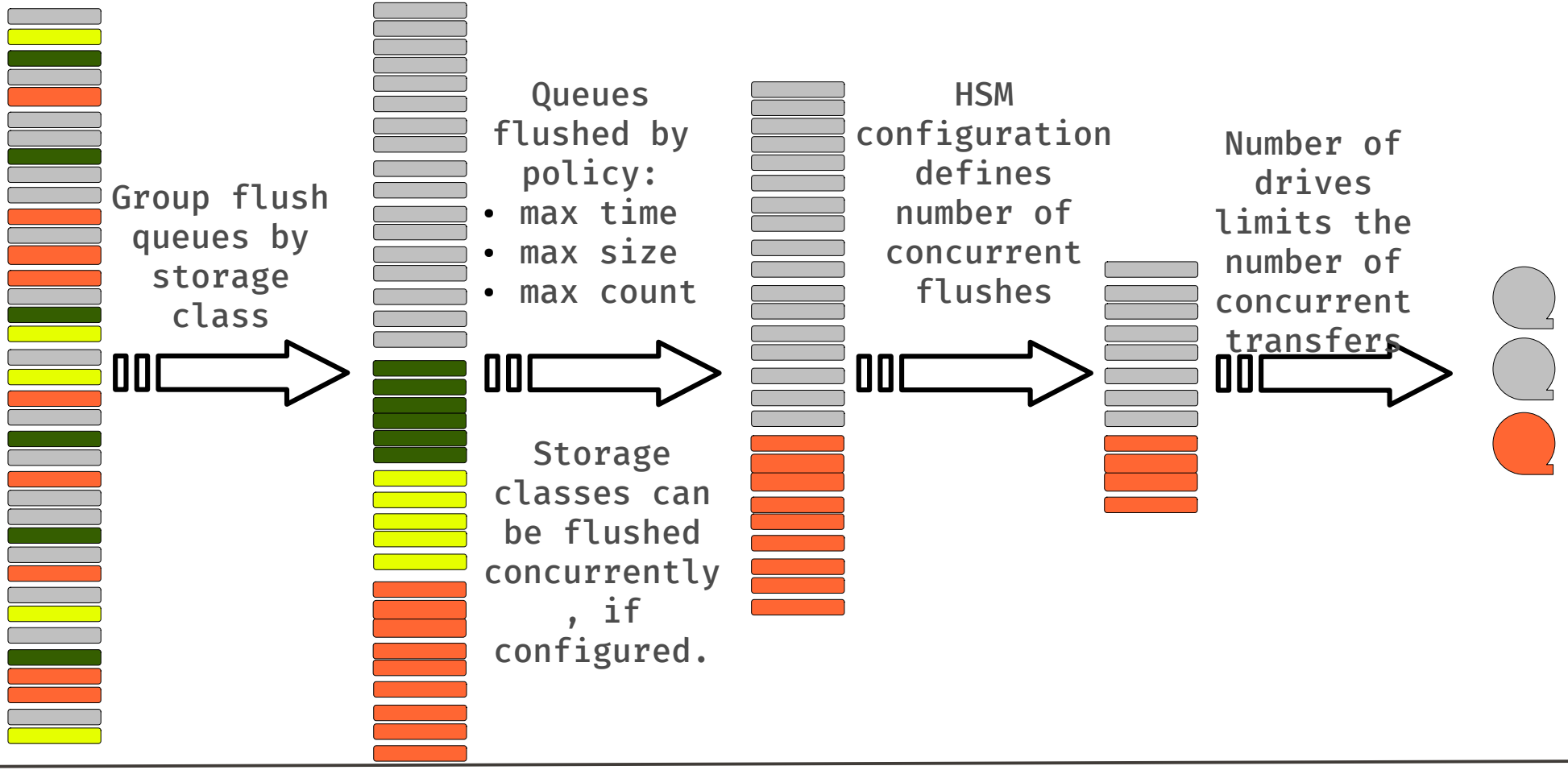
At analysis facility

- Automatic data migration
- Bulk recall on periodic basis
- Long-term nearline access
- Recall prioritization

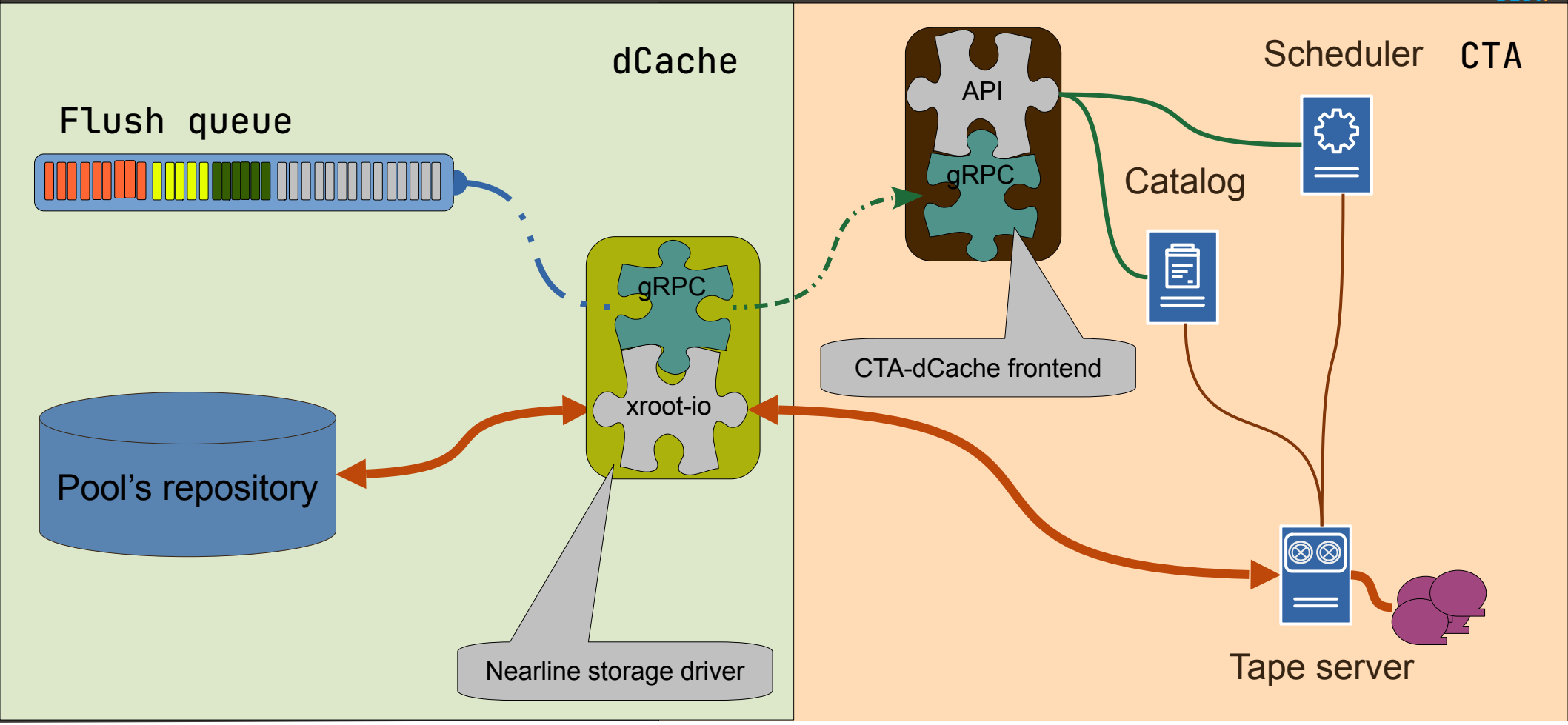
Data Archive

- Manual data migration
- Long-term preservation
- Automatic technology migration
- Self-healing

The Flush Queue



Nearline CTA Storage Driver



Restore Aware Components