AsyncStageOut: Distributed user data management for CMS Analysis

Hassen Riahi   CERN IT-SDC
Tony Wildish   Princeton University (US)
Diego Ciangottini   University & INFN Perugia (IT)

Outline

- Overview

- Problem and strategy

- Architecture

- Integration

- Results

- Conclusions

Distributed data analysis in CMS

- 1000 individual users per month
- More than 60 sites
- 20k jobs/hour
- Typically 1 file/job
  - Files vary in size
- 200k completed jobs per day
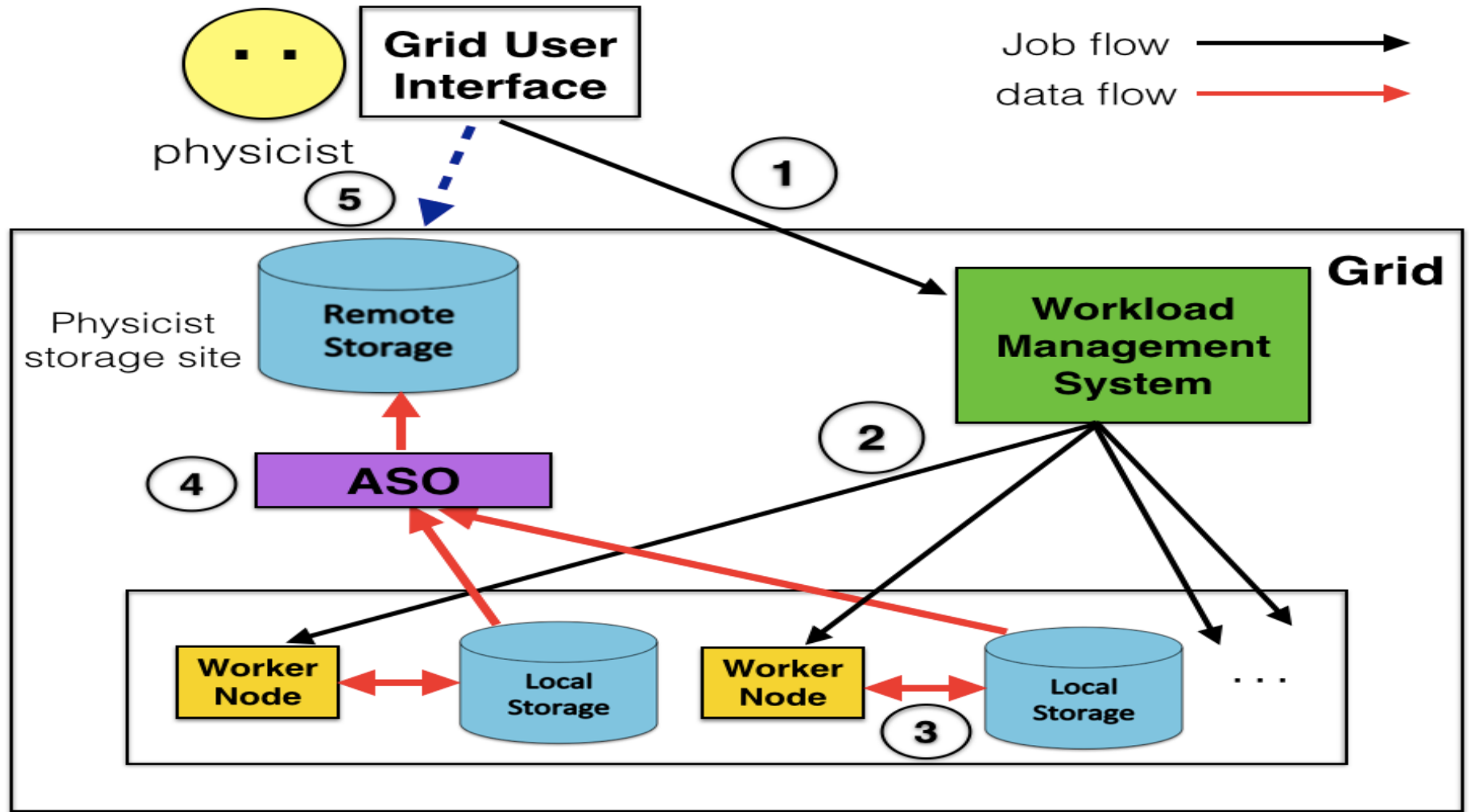- Minimal latencies
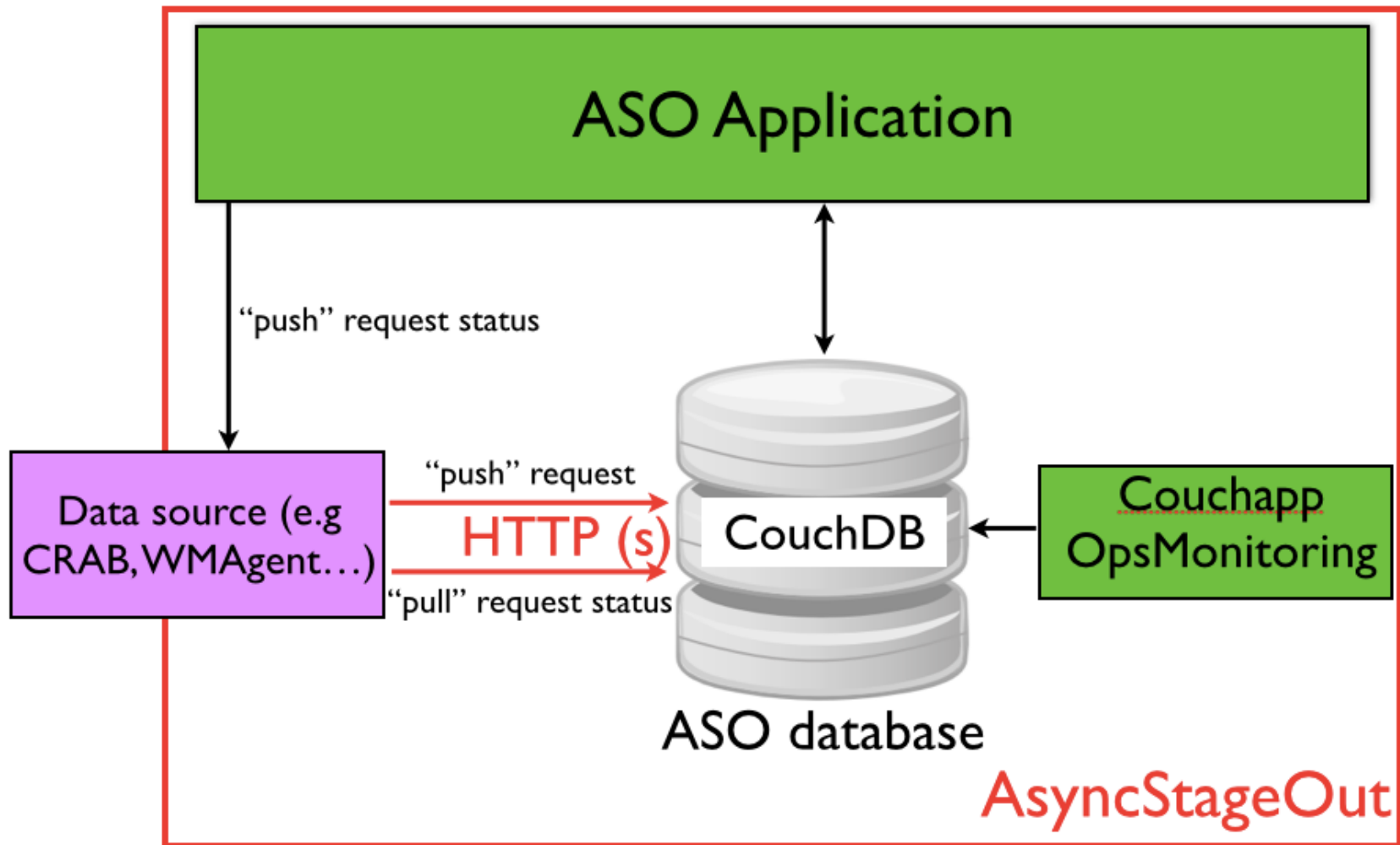- Chaotic environment
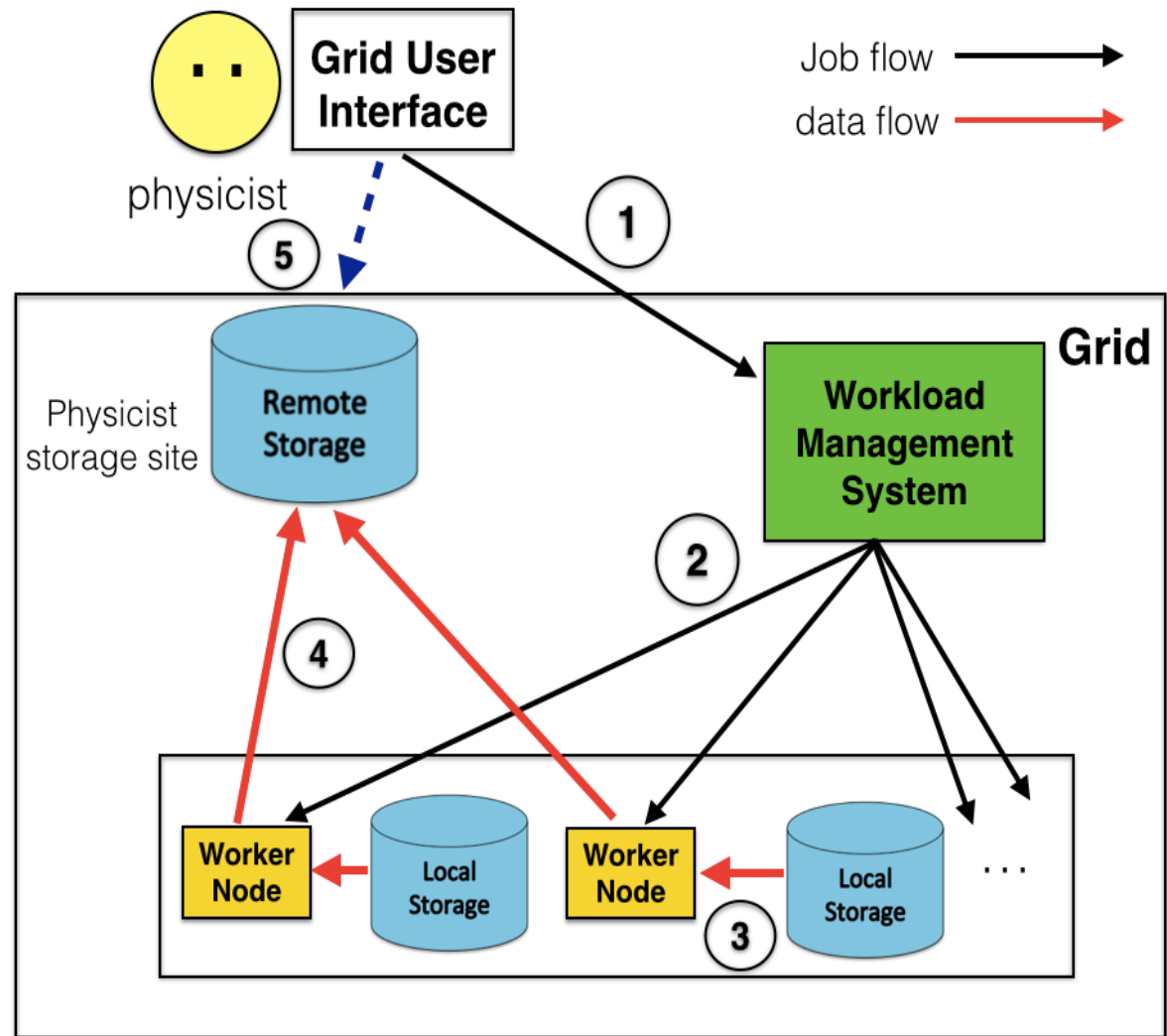
Problem

- 15% to 20% the jobs fail and about 30% to 50% of the failures are due to the jobs not being able to upload their output data to a remote disk storage
    - between 5% and 10% of jobs fail in the remote copy of outputs
    - the overall CPU loss is even higher than 5-10% since those jobs fail at the end of the processing after multiple retries
    - often it results in DDoS to CMS Tier-2 storage systems

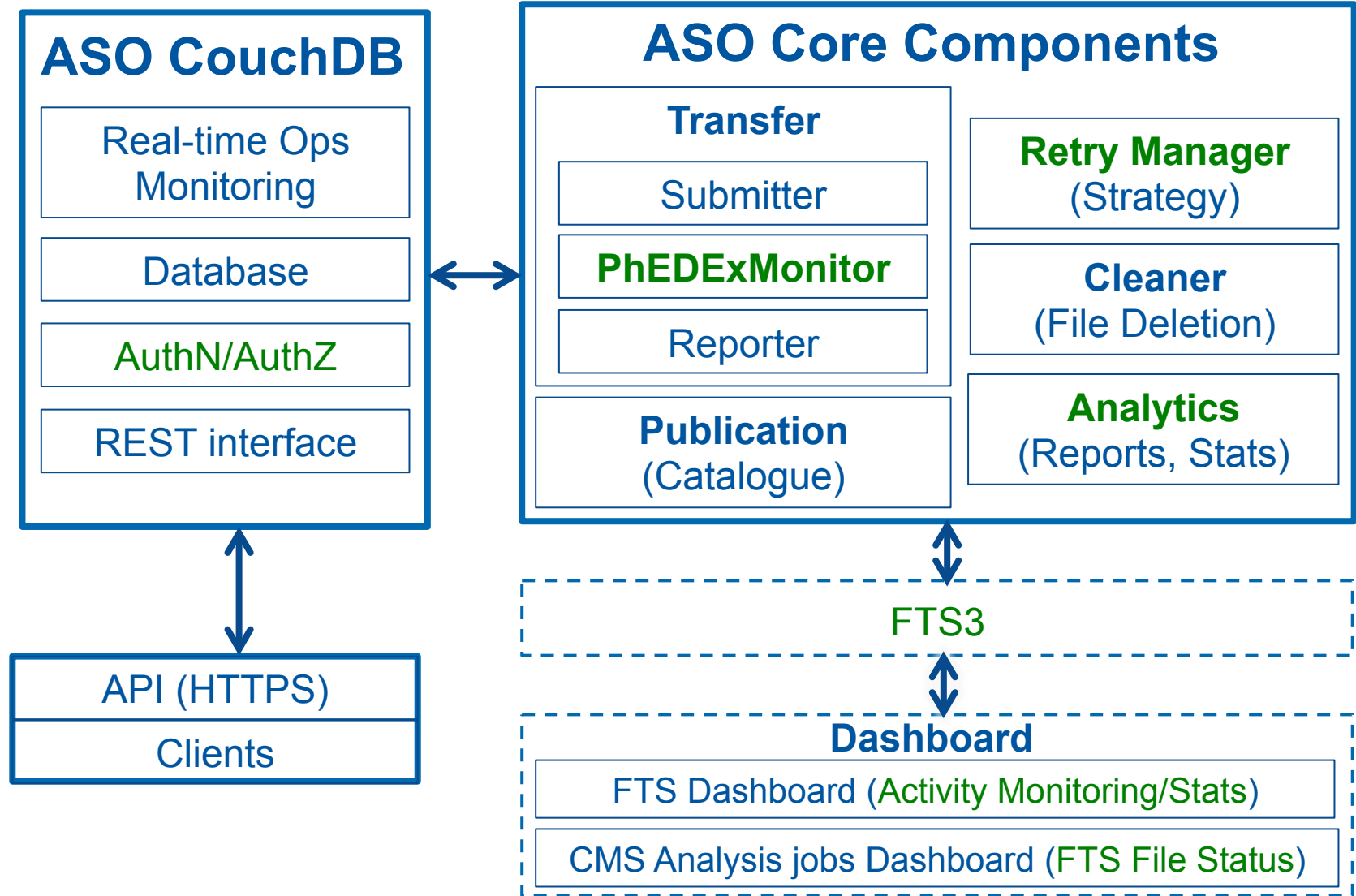AsyncStageOut (ASO) is implemented to reduce the most common failure mode of analysis jobs

Asynchronous stage-out stretagy
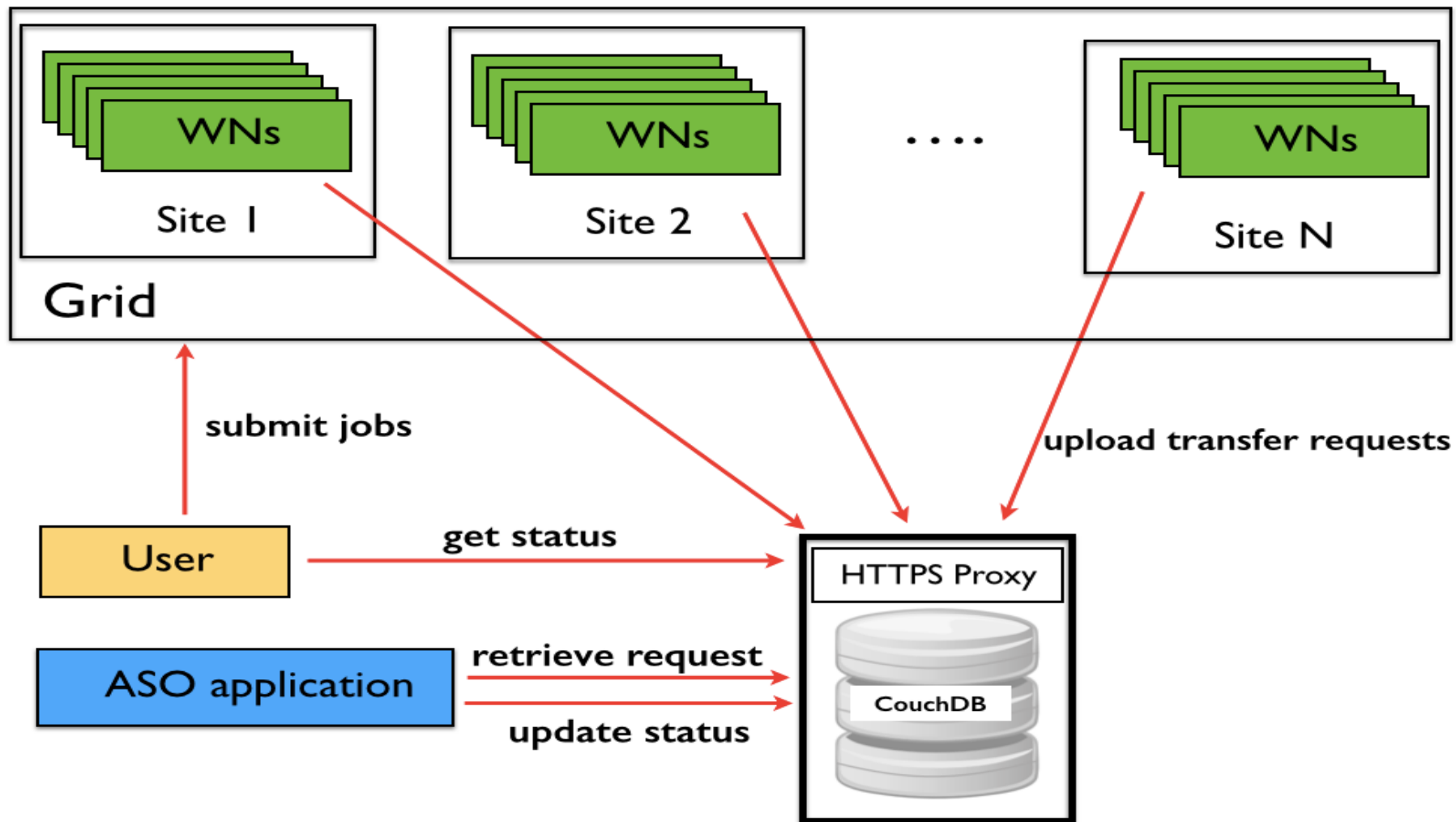
Architecture overview

Architecture and evolution

ASO CouchDB

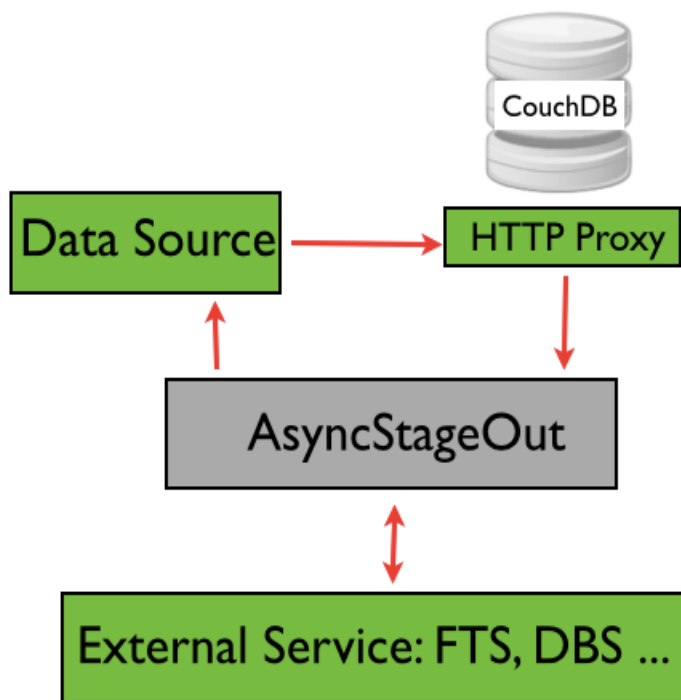| Real-time Ops Monitoring |
| --- |
| Database |
| AuthN/AuthZ |
| REST interface |

API (HTTPS)
Clients

ASO Core Components

Transfer

Submitter
PhEDExMonitor
Reporter

Publication
(Catalogue)

Retry Manager
(Strategy)

Cleaner
(File Deletion)

Analytics
(Reports, Stats)

FTS3

Dashboard

FTS Dashboard (Activity Monitoring/Stats)
CMS Analysis jobs Dashboard (FTS File Status)

CMS analysis jobs Dashboard

Integration

Deployment models
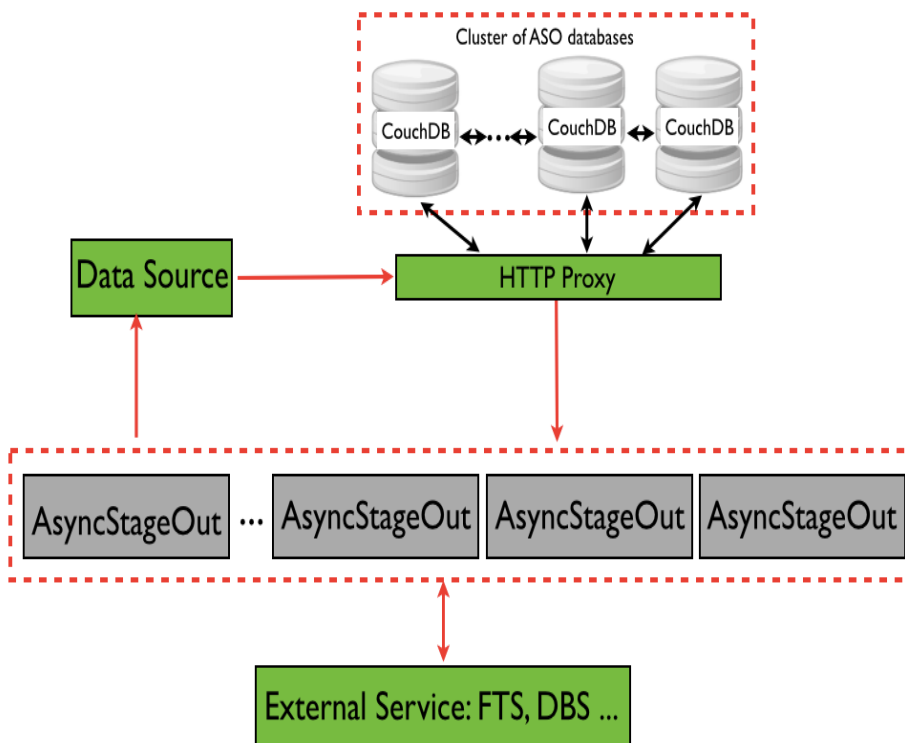
CSA14 exercise

Running CRAB jobs
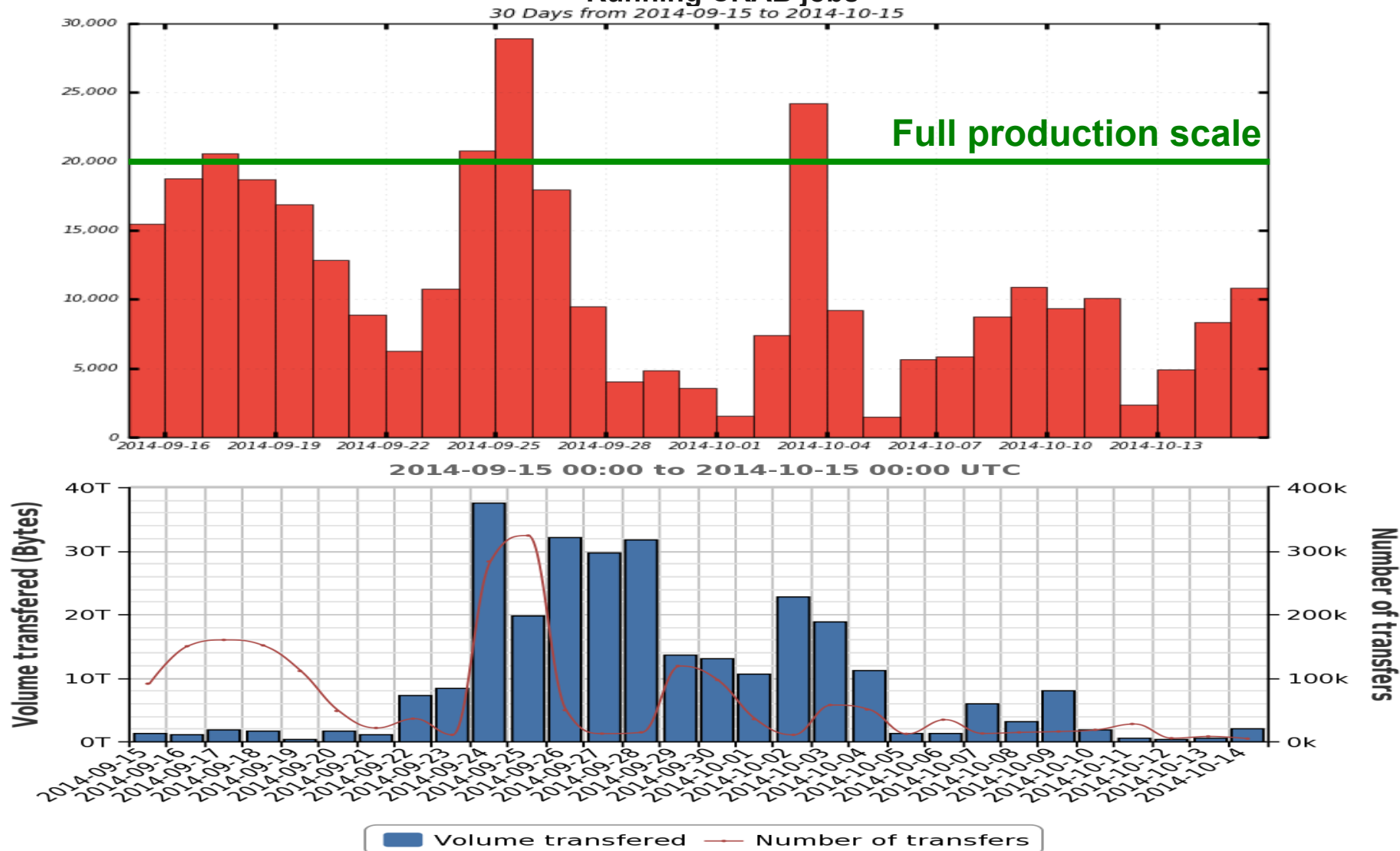
Production results

- ASO has started production in June 2014
  - More than 2 PB/7 M files transferred during the last 6 months
  - Peaks of more than 700k transfers per week

Volume transfered / Number of transfers (cms)

2014-09-01 00:00 to 2015-03-01 00:00 UTC
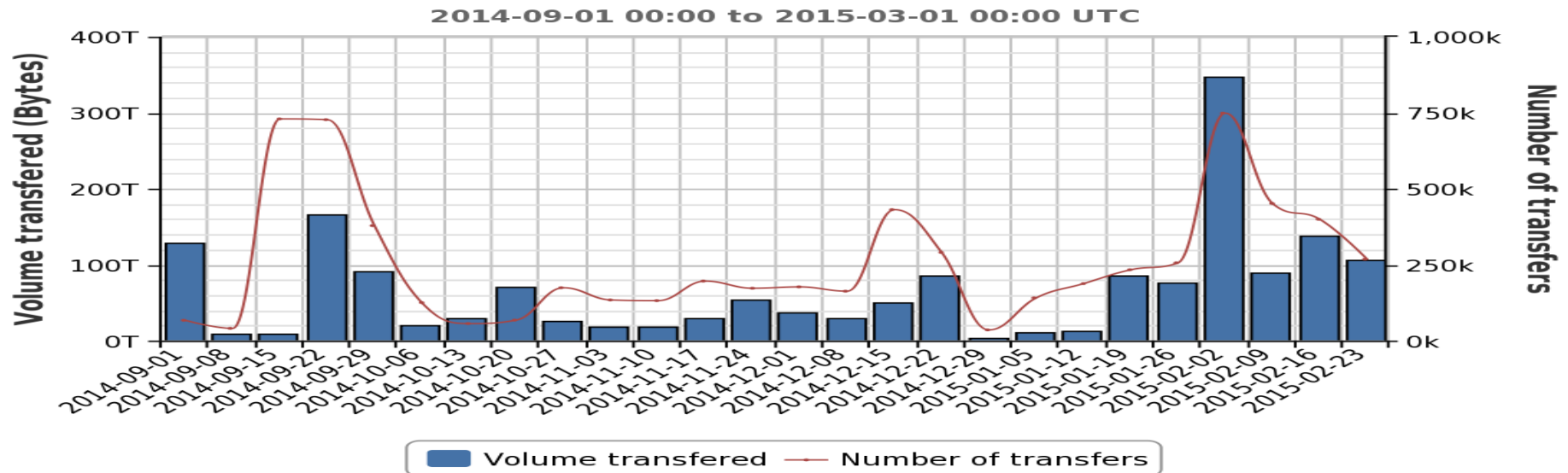
Scale tests

- Goal:
  - Simulate data transfers using PhEDEx LifeCycle Agent
  - Explore the scalability limits of ASO by scaling-up to 2-4 times the design load within the production and the new version of CouchDB
    - The design load is ~ 200 k completed analysis jobs/day
      - ➢ ~ 300 k completed files/day (~ 200k outputs + ~ 100k logs)
- Configuration:
  - 200 parallel users
  - 100 files per FTS job
  - 60 sites
  - File size: 2 GB
  - Transfer throughput: 100 MB/s
  - Run the system for more than 12 hours

2 times scale

600 k completed files/day
CouchDB 1.1 (production)

Injection starts

Injection ends

20,000

15:00

7:00

CouchDB 1.6

20,000

6:00

19:00
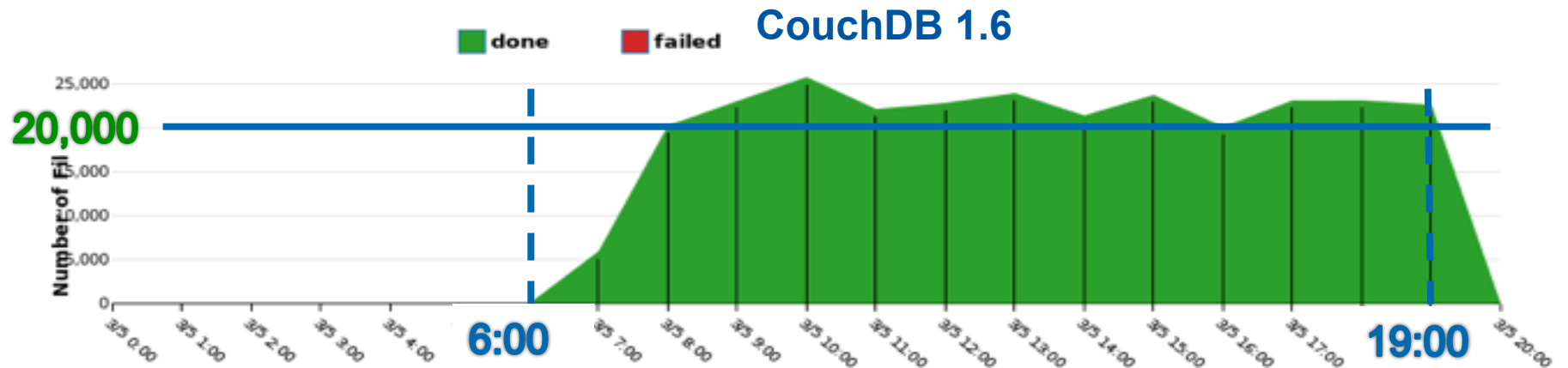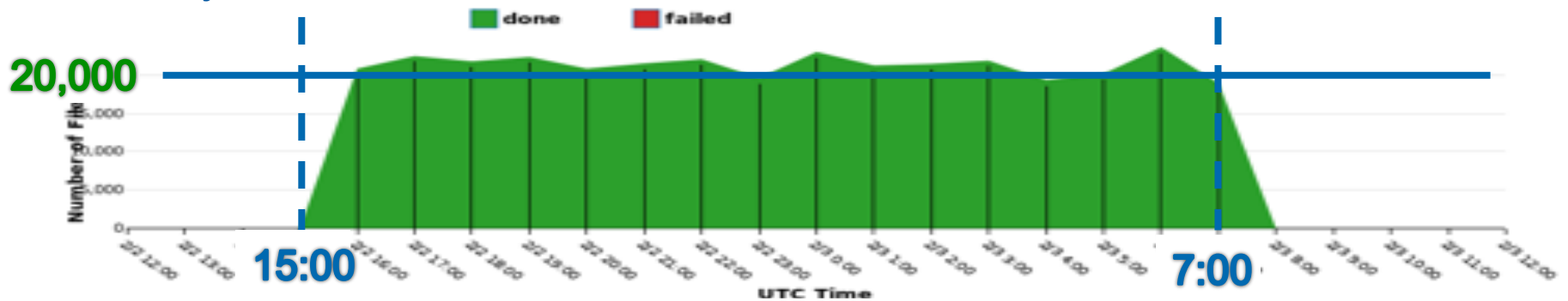
✓ Independently of the version of CouchDB, ASO can manage 2 times scale the design load

4 times scale

1,2 M completed files/day

Injection starts

Core dump in the Monitor component
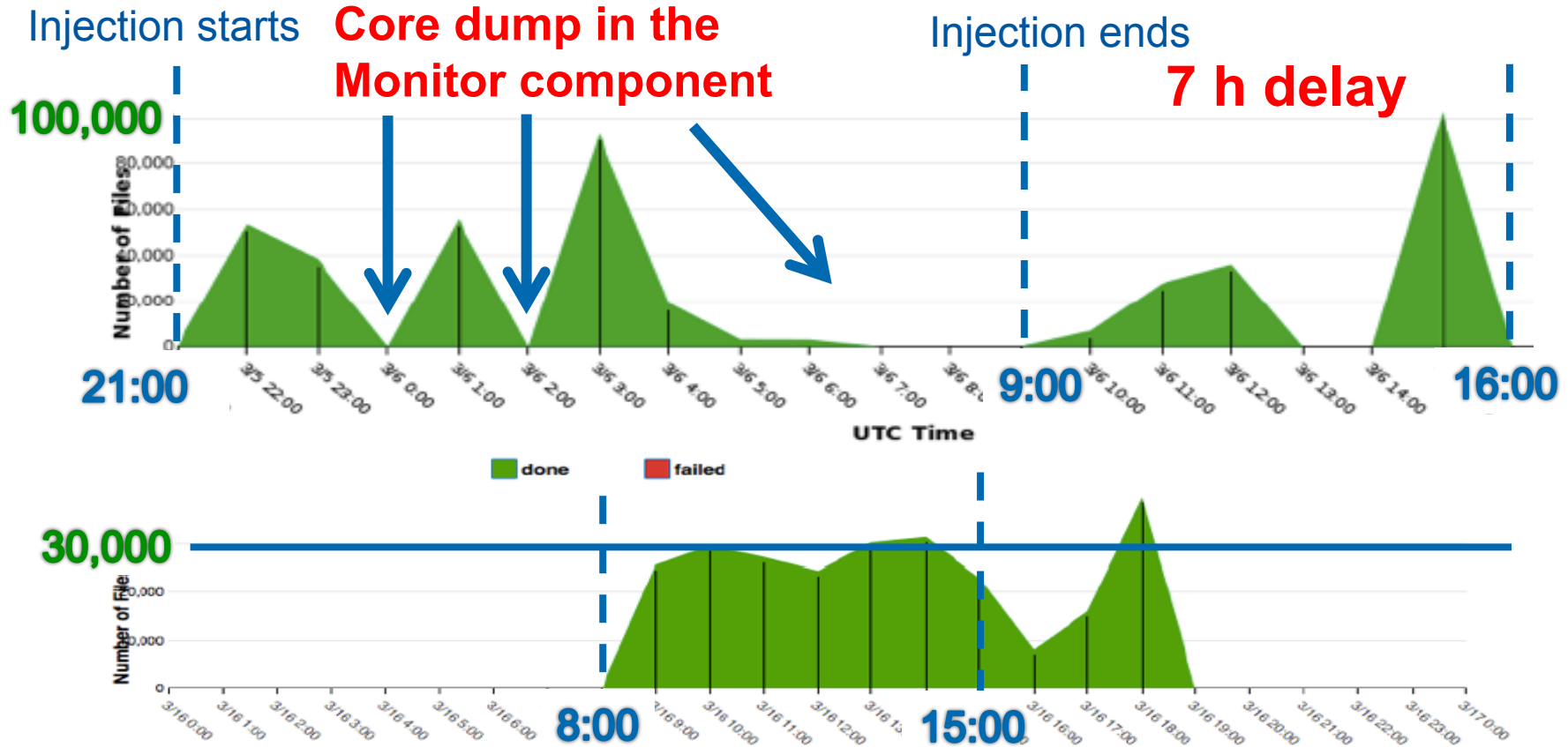
Injection ends

7 h delay

✓ ASO can manage ~ 3 times scale the design load with tuned parameters

Summary and conclusions

- ASO has evolved from a limited prototype to a highly adaptable and scalable service

- ASO has shown good performance during commissioning and production

- ASO can manage 2 times scale of the design load
    - The management of 4 times scale is possible with an accurate tune of ASO and system parameters

- Re-use of design and components from PhEDEx point the way to a more modular architecture for data-management tools in CMS
    - ➢ Long-term maintainability, performance and adaptability