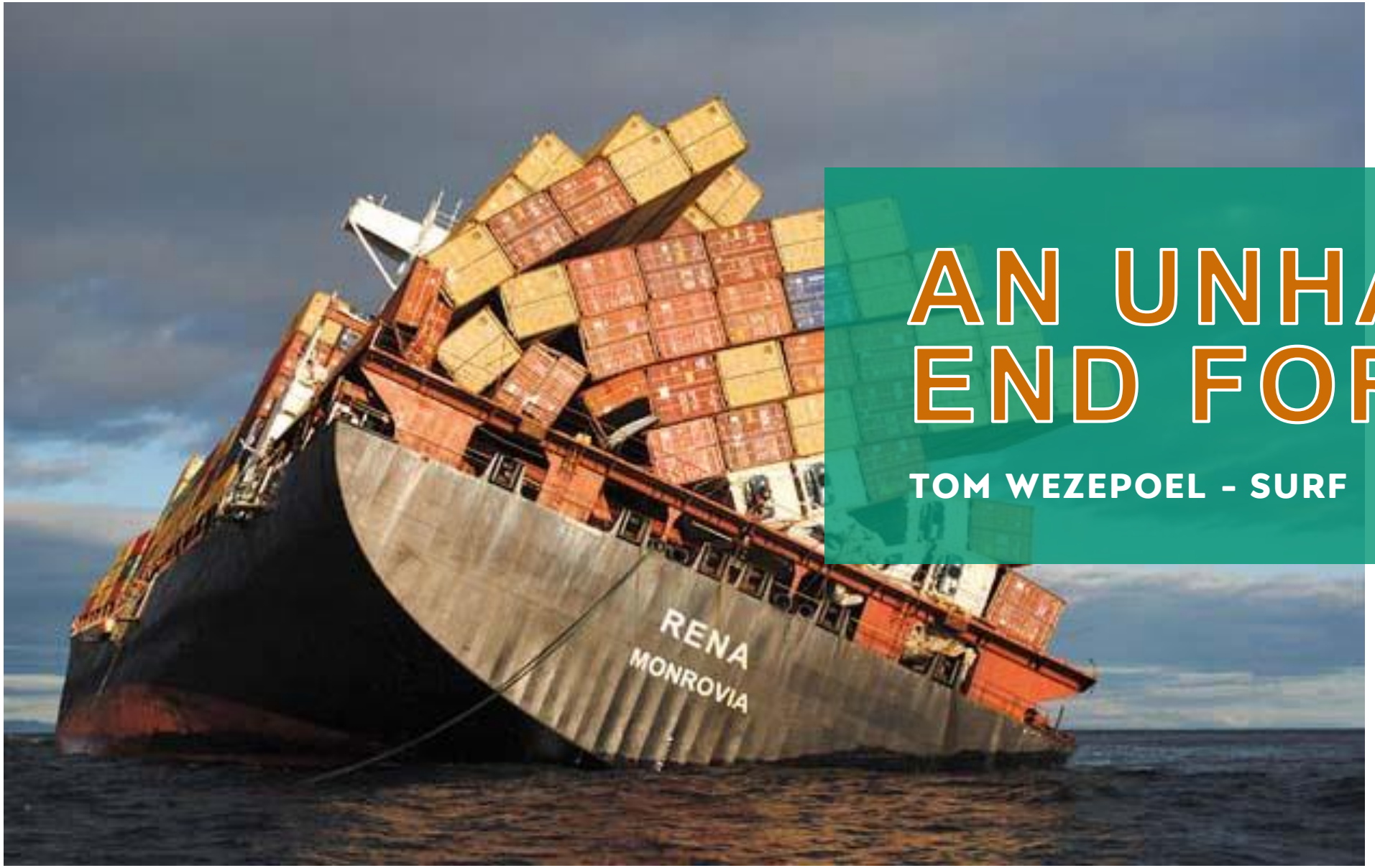
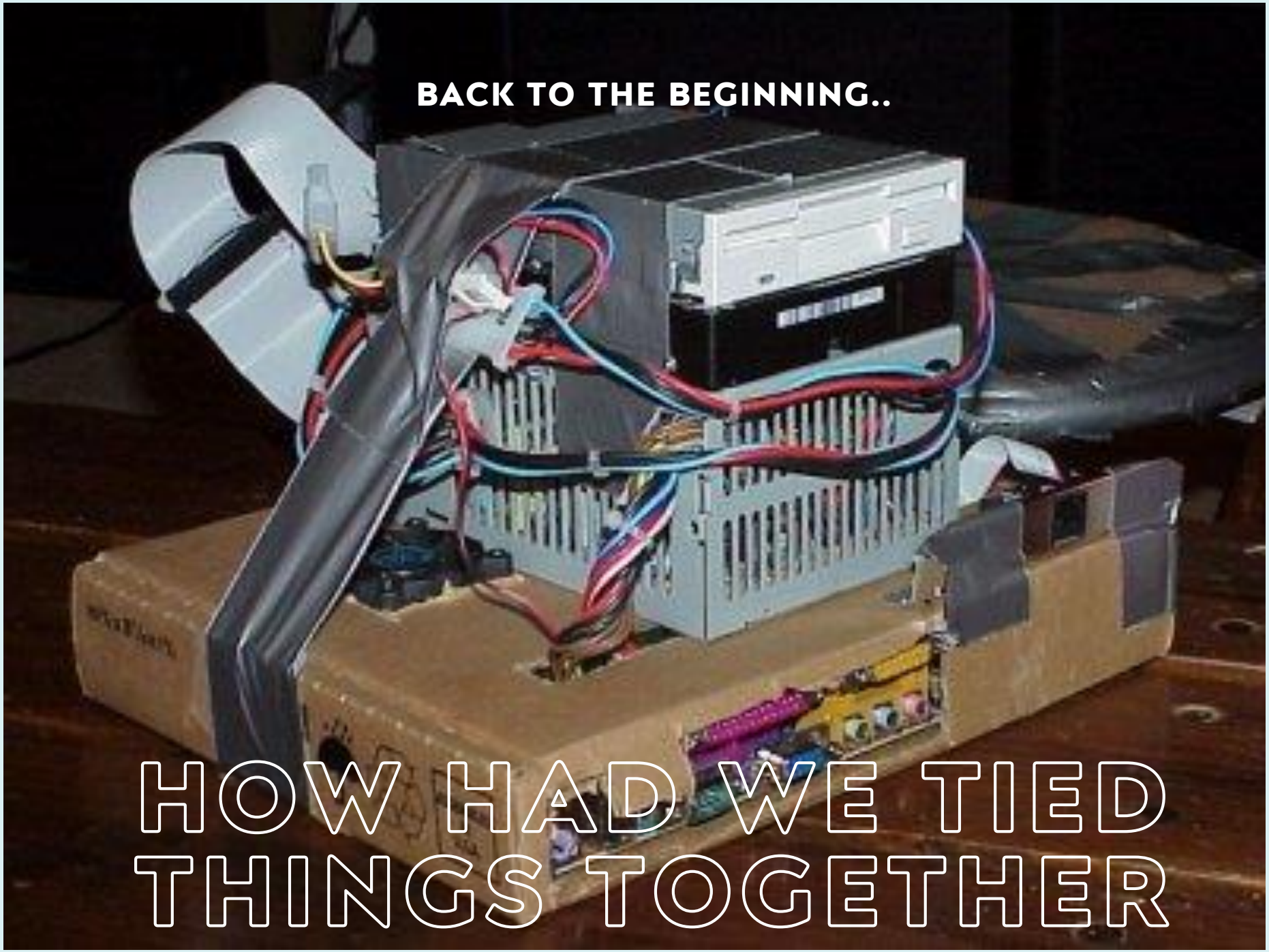


AN UNHAPPY END FOR S3

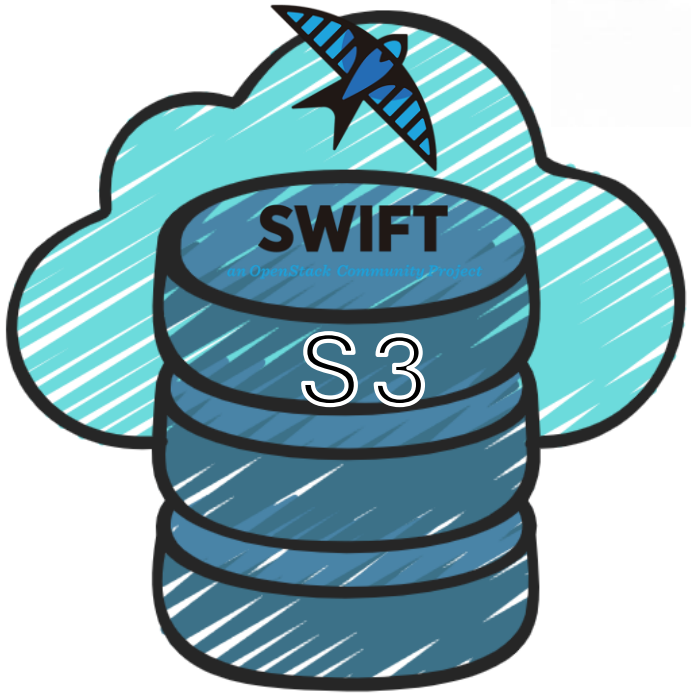
TOM WEZEPOEL – SURF


BACK TO THE BEGINNING..
HOW HAD WE TIED THINGS TOGETHER

So you take two web applications.

Both actually do GET, PUT , POST, DELETE

Connect them together with some magic



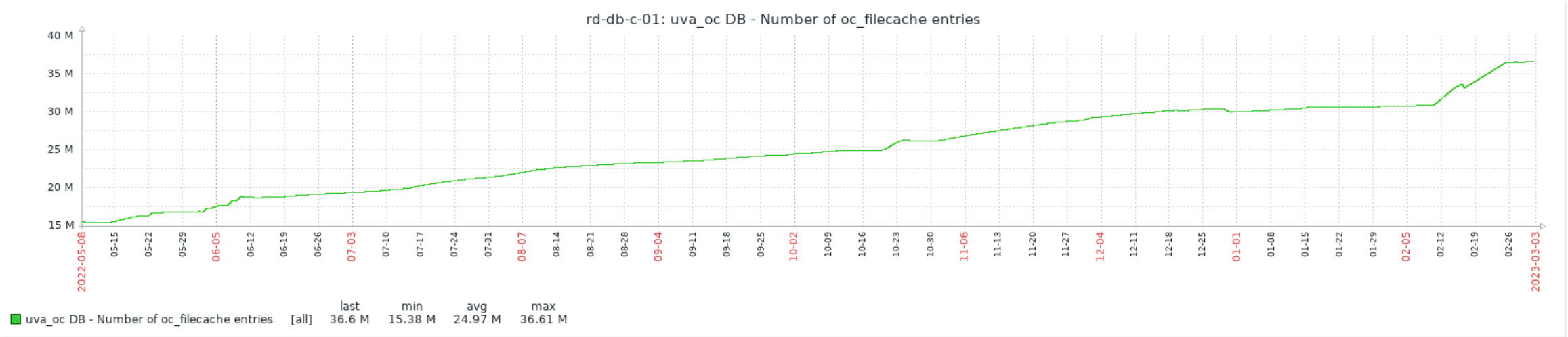
1 + 1 =

EVERYTHING WORKS

Performance is fine

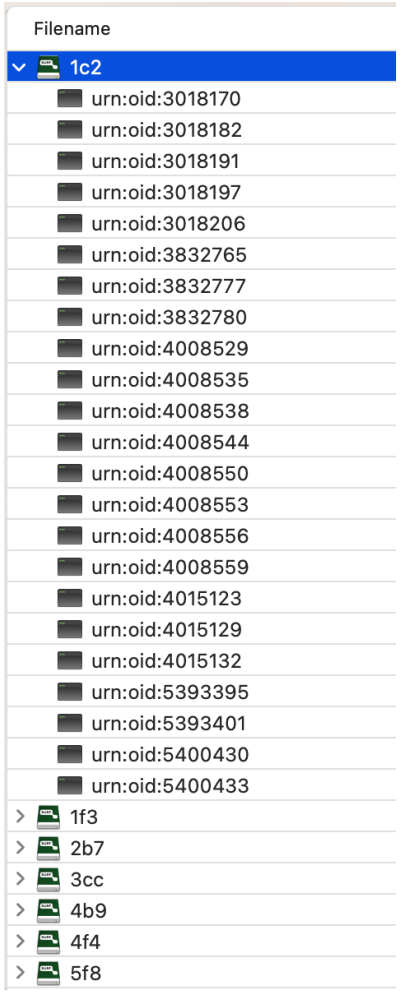
Users get busy, number of files grows.

So time to sit back, for a cup of coffee!



HAVE A BETTER LOOK AT THE SYSTEM

- Filescans are not possible. Annoying, but oh well..
- How does it actually work with File Versions ?



FILE VERSIONS

Each file is uploaded twice

- As File
- As Version

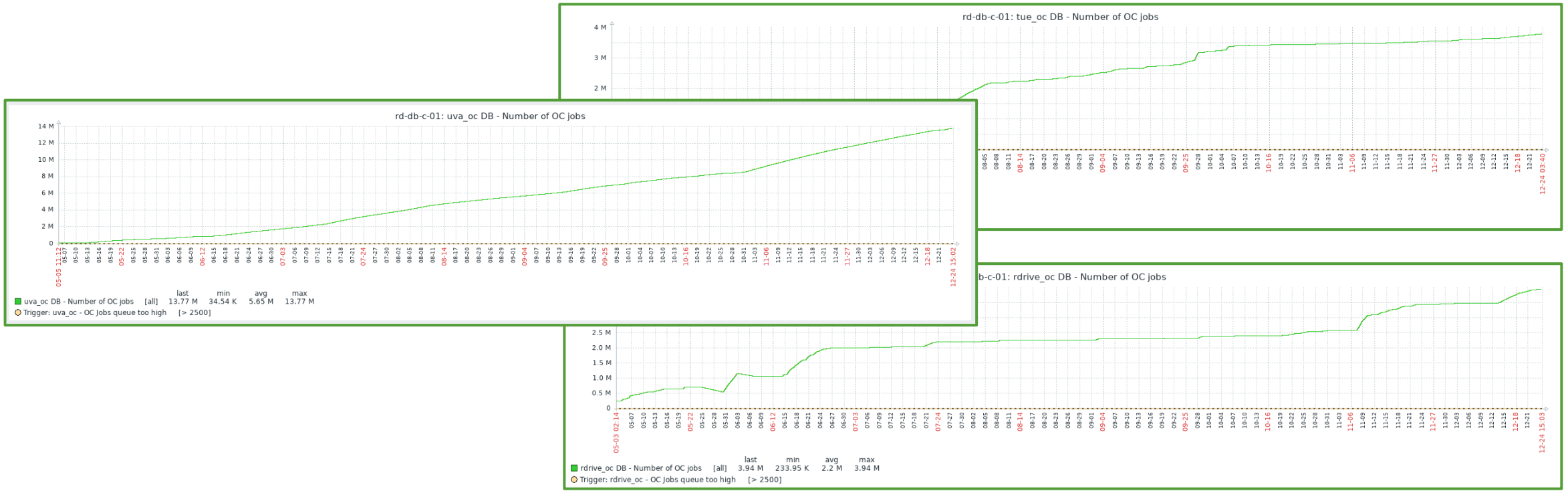
Size of files	Percentage
< 500kb	87,43%
500k - 1M	2,20%
1 - 10M	3,02%
10-100M	4,85%
100M - 1G	2,31%
> 1G	0,16%

FILE VERSIONS

Each version has a certain lifetime

and have to be cleaned up later on

But.. Each cleanup job keeps running for days..



FILE VERSIONS

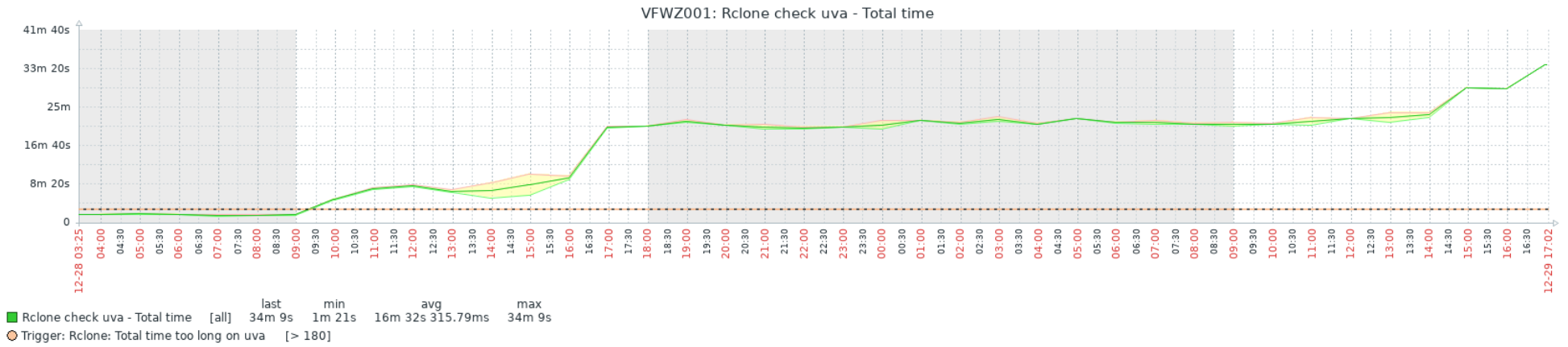
Why not using bucket versioning ?

Example

Time Period before last Expiration	Maximum Number of Versions:
1 second	1
10 seconds	5
1 minute	6
1 hour	59
1 day	23
30 days	30

IMPACT

1. Versions were cleaned up quickly enough.
2. Performance of system deteriorates rapidly
3. Own created clean-up scripts cause issues with storage usage calculation



MIGRATION TO POSIX

SCALITY

IBM Spectrum Scale

SWIFT
an OpenStack Community Project

S3

ceph

WHICH POSIX SOLUTION ?

ceph

SCALITY

IBM Spectrum Scale

GOAL

1. Current amount of storage in use
   - Research Drive in total reported by the filecache = 955TB
   - Research Drive files only in filecache = 404TB
   - Research Drive trashbin and versions = 551TB

2. Architecture.
   - Our service is spread over 3 datacenters arround Amsterdam

TEST - ARCHITECTURE

Ceph Cluster over 3 datacenters in VMs
- Each zone contains 5 nodes
- Each node contains 5 drives of 5 GB
- Meta & Data on same drives

IBM Spectrum Scale Cluster over 3 datacenters in VMs
- Each zone contains 4 nodes
- Each node contains 3 data drives of 5GB & 1 meta drive

Scality Ring 8 over 3 datacenters on physical hardware
- Each zone contains 4 nodes
- Each node contains 10 data drives of 5.5 TB and 2 metadata SSDs

TEST – READ/WRITE

Write

- dd if=/dev/zero of=test1M bs=1K count=1024 conv=fdatasync
- dd if=/dev/zero of=test1G bs=1M count=1024 conv=fdatasync
- dd if=/dev/zero of=test10G bs=10M count=1024 conv=fdatasync

Read

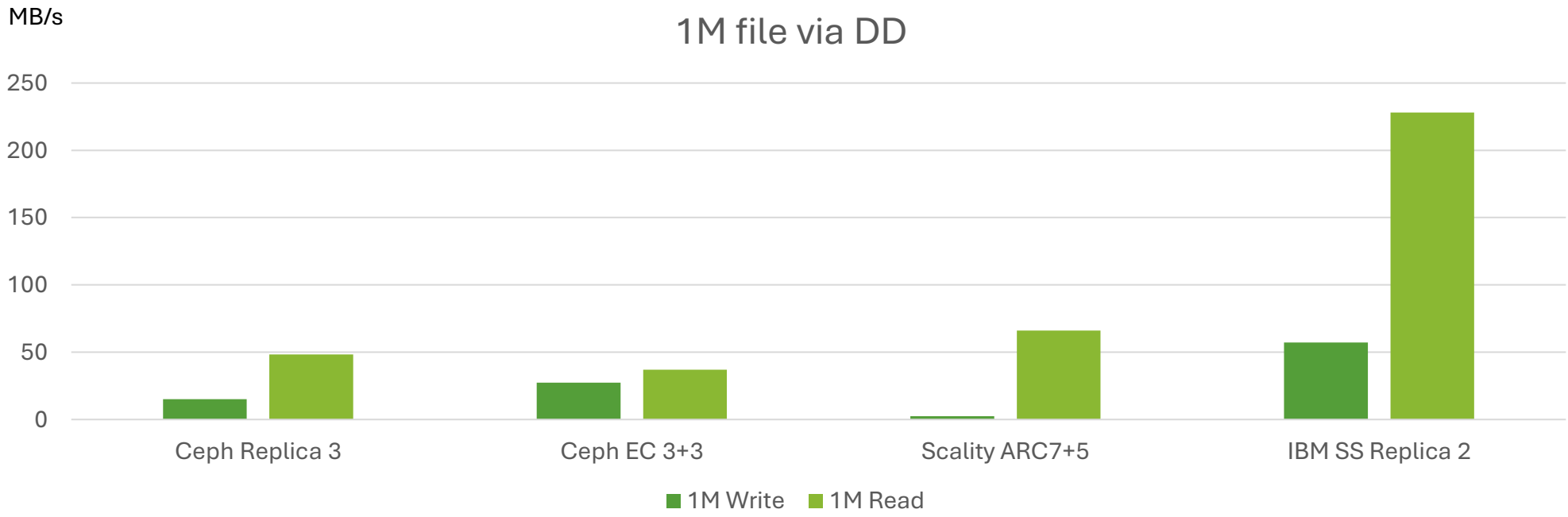
- dd if=test1M of=/dev/null bs=1K count=1024 oflag=sync
- dd if=test1G of=/dev/null bs=1M count=1024 oflag=sync
- dd if=test10G of=/dev/null bs=10M count=1024 oflag=sync

TEST – RESULTS

TEST – RESULTS

Ceph Replica 3	15	48,3
Ceph EC 3+3	27,3	37
Scality ARC7+5	2,4	66,1
IBM SS Replica 2	57,2	228

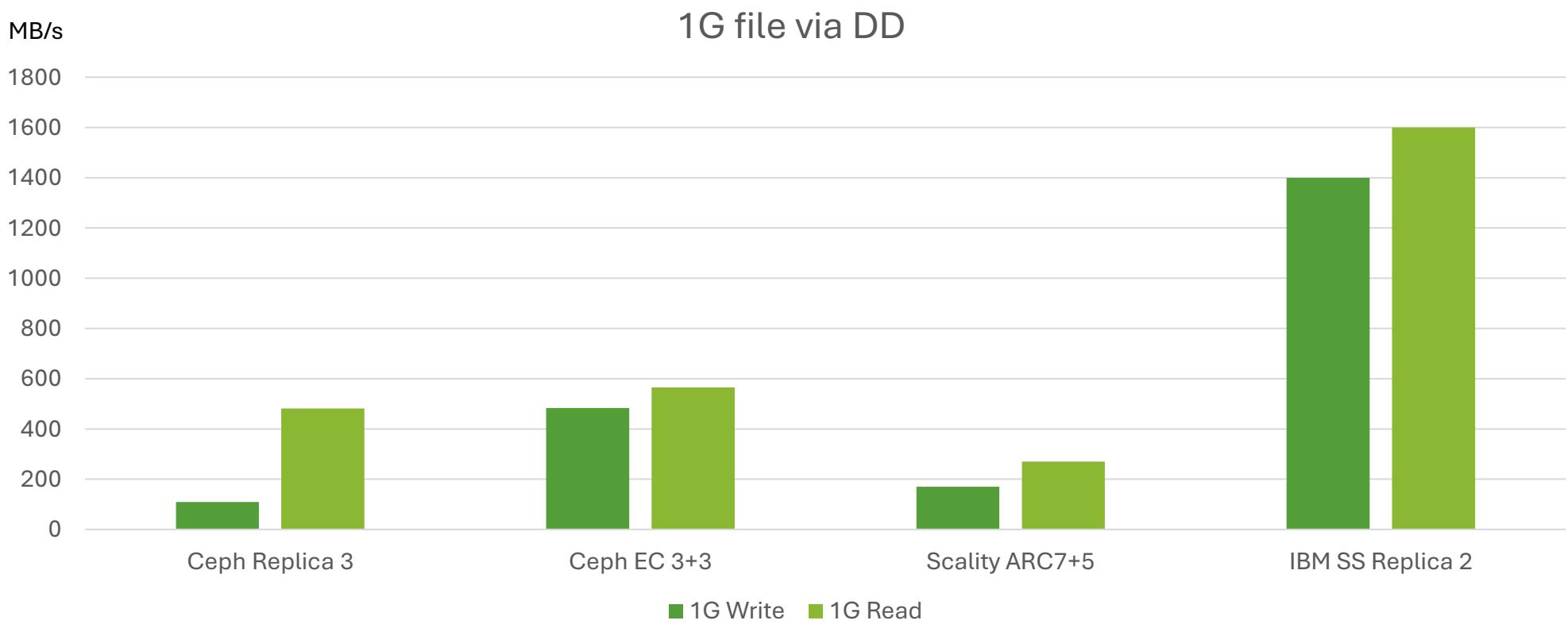
1M FILE READ AND WRITE



TEST – RESULTS

Ceph Replica 3	109	481
Ceph EC 3+3	484	565
Scality ARC7+5	170	270
IBM SS Replica 2	1400	1600

1G FILE READ AND WRITE



TEST – RESULTS

Ceph Replica 3	443	505
Ceph EC 3+3	635	535
Scality ARC7+5	232	332
IBM SS Replica 2	1500	1700

10G FILE READ AND WRITE

TEST - IBM SS EXPLANATION

It’s not normal replication..

```
[root@ibmss-tst-b-01 ~]# /usr/lpp/mmfs/samples/fpo/mmgetlocation -f /mnt/gpfs/ubuntu-20.04.4-
live-server-amd64.iso
[FILE: /mnt/gpfs/ubuntu-20.04.4-live-server-amd64.iso SUMMARY INFO]

replica1:                                   replica2:
ibmss-tst-c-04: 27 chunk(s)                 ibmss-tst-b-01: 24 chunk(s)
ibmss-tst-a-02: 26 chunk(s)                 ibmss-tst-b-02: 29 chunk(s)
ibmss-tst-b-04: 25 chunk(s)                 ibmss-tst-c-02: 29 chunk(s)
..                                          ..
ibmss-tst-c-01: 28 chunk(s)                 ibmss-tst-b-04: 29 chunk(s)
ibmss-tst-b-02: 26 chunk(s)                 ibmss-tst-b-03: 22 chunk(s)
```

TEST – CONCLUSION

1. IBM was clearly the winner
2. CephFS with other settings performed better than these charts
3. But of course, the price of a solution also has a big role too..

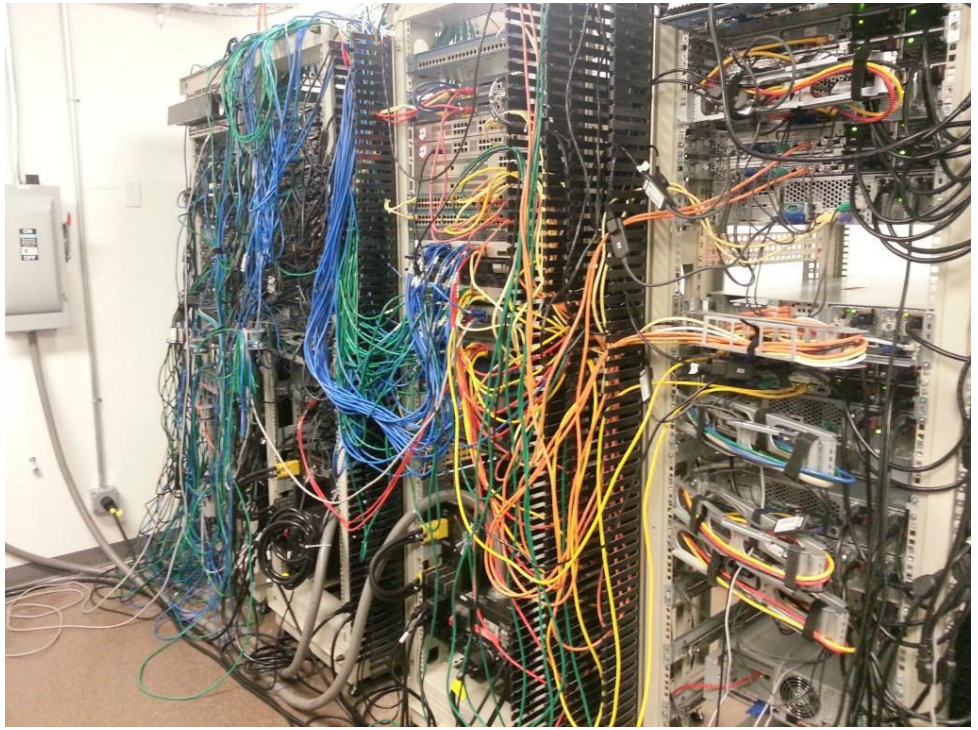
The price for the IBM SS solution over 3 DC was so many times more expensive than the solutions with Ceph & Scality.

Which was then unfortunately a bummer, despite of the nice performance and options.

NEXT STEPS – SETUP CEPHFS

1. Hardware has been ordered & is being racked.
2. Installing the hardware with AlmaLinux
3. Setup CephFS
4. Do some performances tests.

5. Then.. The migration..



NEXT STEPS - MIGRATE FROM S3

SURFdrive migration from POSIX GlusterFS to POSIX Scality was easier..

```
while true
do
  rsync_userfile_to_new_storage()

  lock_user()

  rename_old_user_homestore_on_glusterfs()

  create_symlink_to_new_scality_homestore()

  unlock_user()
done
```

NEXT STEPS - MIGRATE FROM S3

Reconstruct filecache table user by user

1. Get bucket of an user
2. Create all folders by forehand
3. Save current timestamp
4. Sync file by file
5. Sync file by file starting from last saved timestamp
6. Remove objectstore from database and move to posix structure

Footnote: Objectstore S3 only contains the files with the fileid as name.
Folders doesn't exists on S3

NEXT YEAR ON CS3

HORROR STORIES OF MIGRATE FROM S3

Tom Wezepoel
tom.wezepoel@surf.nl

SURF