SRCNet prototyping activities: Testing Rucio Data Lake

S. Sánchez Expósito, M. Parra, M.A. Mendoza, J. Sánchez

IAA INSTITUTO DE ASTROFÍSICA DE ANDALUCÍA
EXCELENCIA SEVERO OCHOA
CSIC

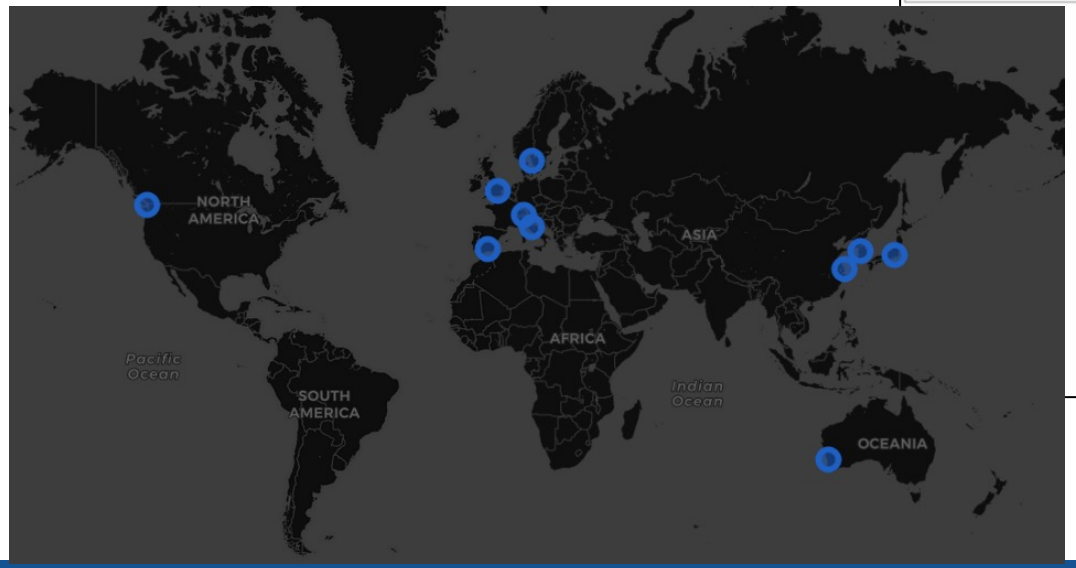
Outline

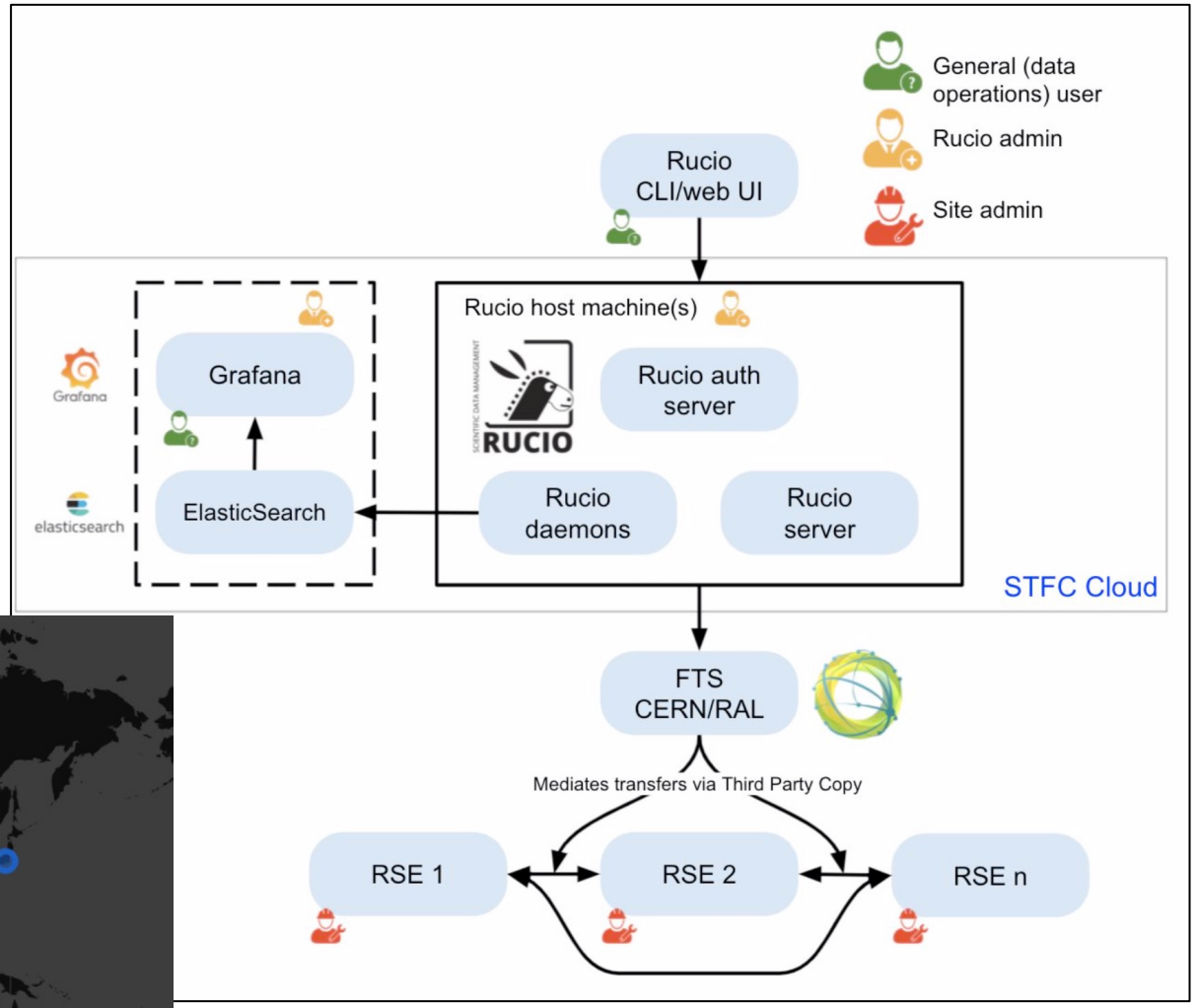
- SKAO Rucio Data Lake
- The Spanish SRC prototype as a Rucio Storage Endpoint
- Results from the tests executed through the Rucio client command line (Tests Ax)
- Results from the tests executed through the Rucio client python API/rucio-task-manager (Tests Bx)
- Next tests
- Questions about Rucio and our network configuration

SKAO Rucio Data Lake

A prototype of a distributed data management system

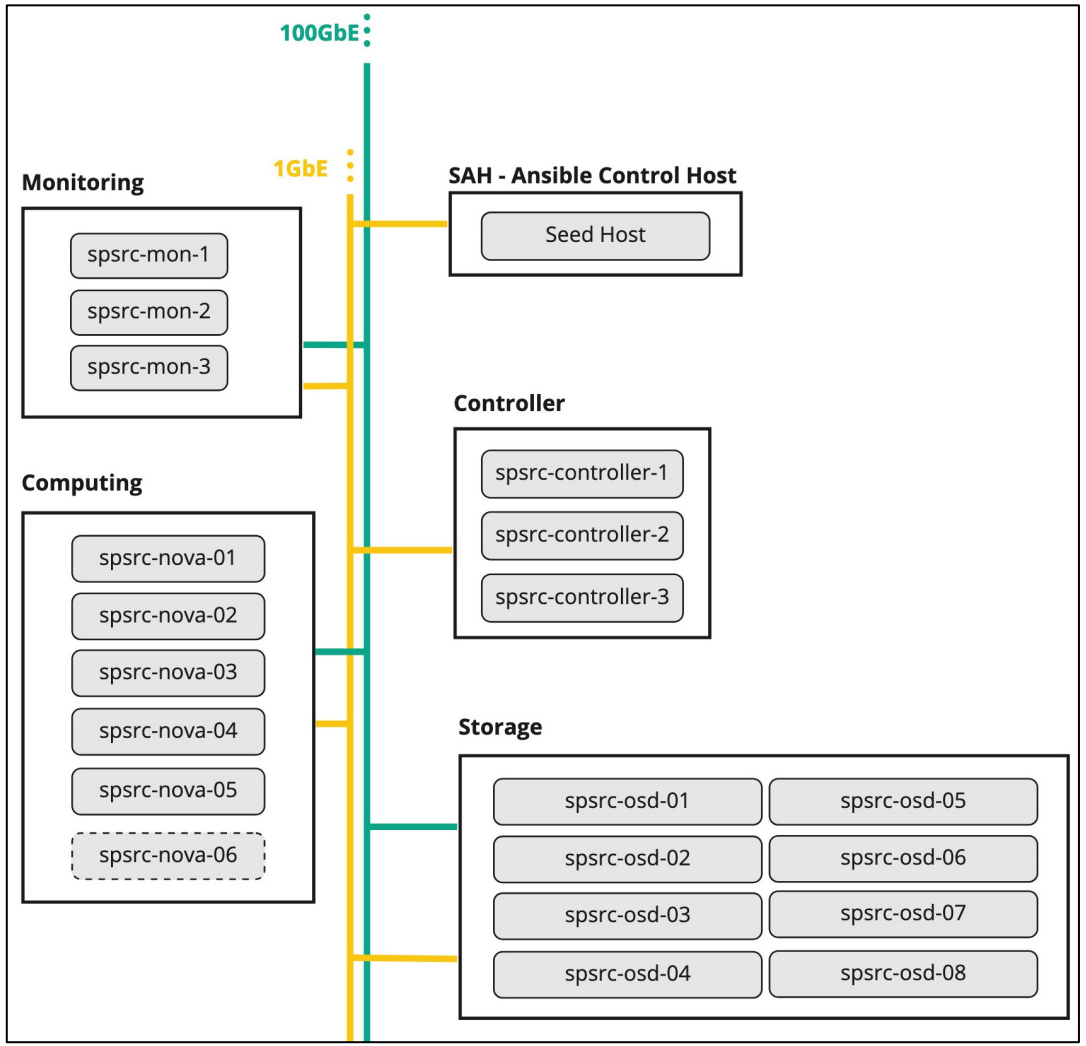
- Operated by a SKAO team (R. Barnsley, R. Joshi, +)
- 11 international sites are integrated
- A functional system
- Testbed to study Rucio tool





Credits: SKAO team

The Spanish SRC prototype integrated as a Rucio Storage Endpoint

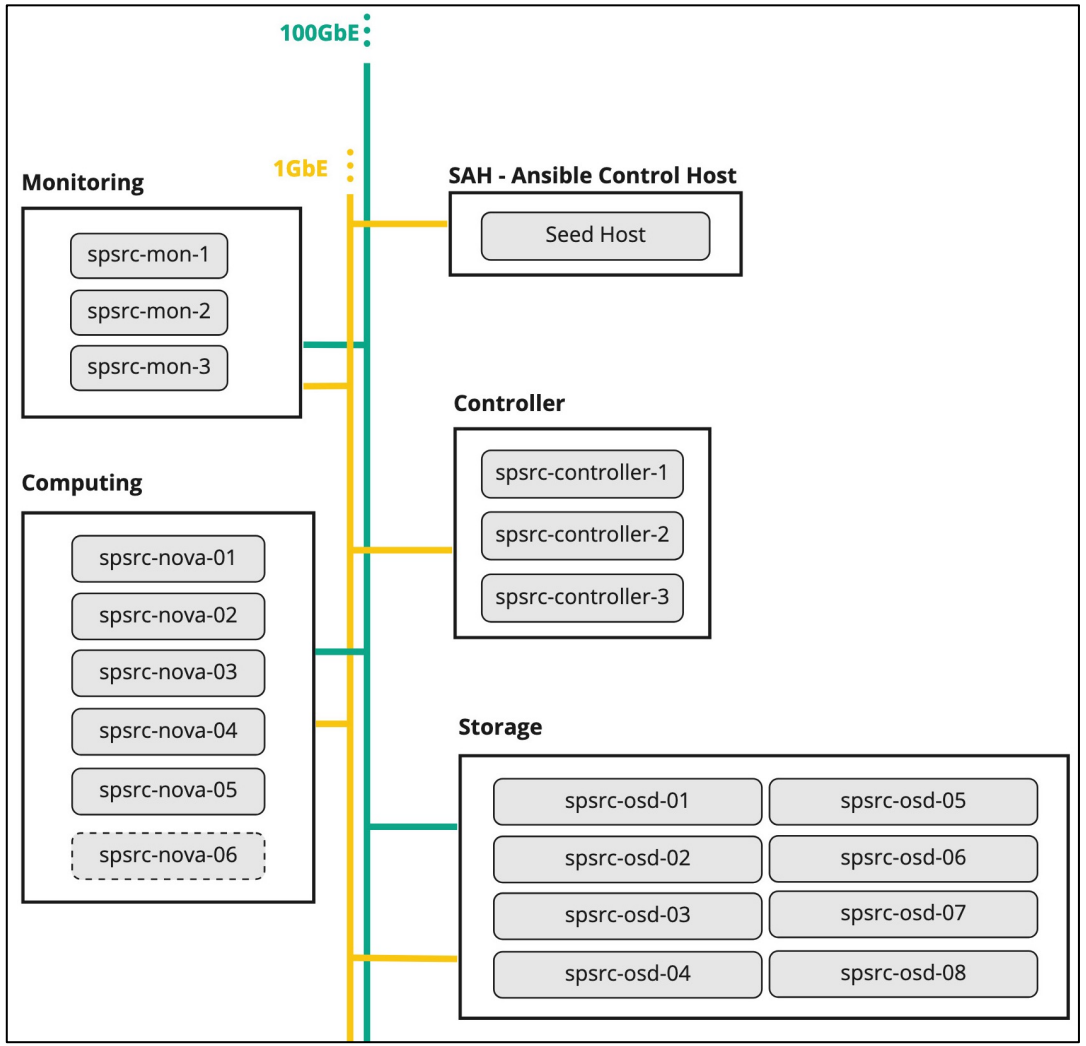


OPENSTACK VICTORIA

ceph

200 CPU Cores
1.5PB Raw storage
10Gbps link to RedIRIS

The Spanish SRC prototype integrated as a Rucio Storage Endpoint



OPENSTACK VICTORIA

StoRM WebDAV

Rucio SE (Virtual Machine)

ceph

Ceph Volume (20TB)

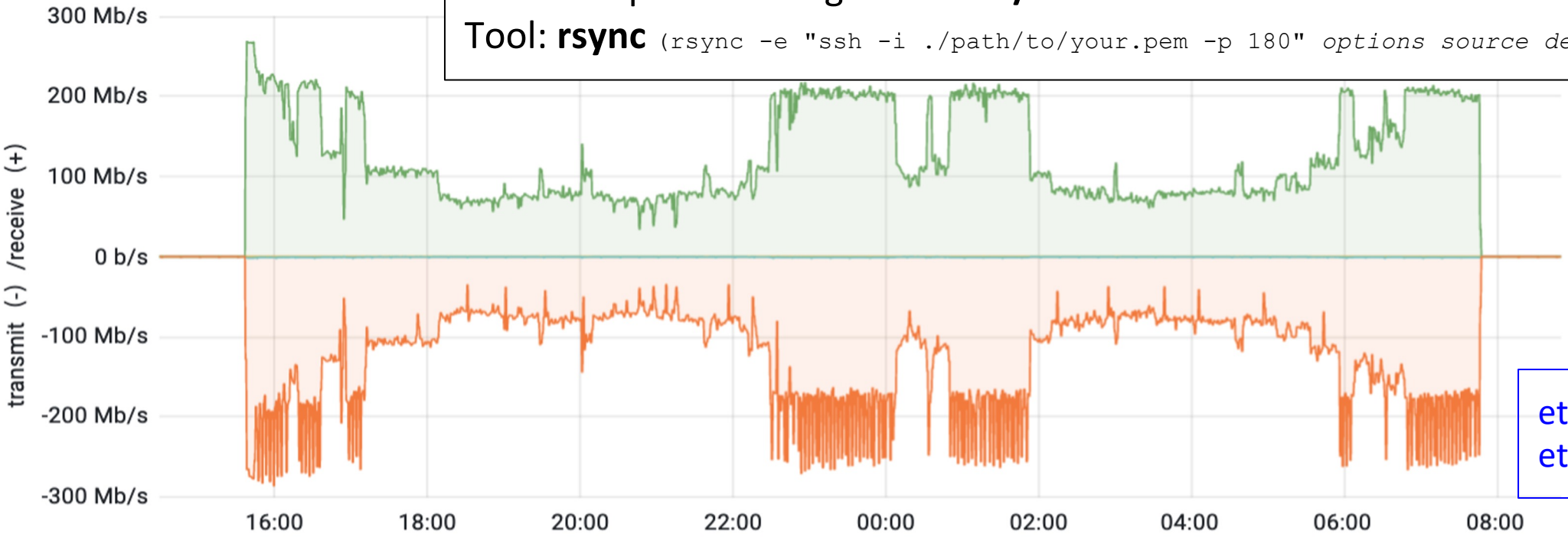
VLAN - storage

200 CPU Cores
1.5PB Raw storage
10Gbps link to RedIRIS

Test A1: Upload the SDC2 dataset to the Rucio Data Lake

MANCHESTER - GRANADA

File: SDC2 dataset (sky_full_v2.fits, **851GB**)
Duration: **~17 hours**.
Transfer speed: average **~120Mb/s**
Tool: **rsync** `(rsync -e "ssh -i ./path/to/your.pem -p 180" options source destiny)`



	min	max	avg	current
eth0_transmit	2.54 kb/s	1.83 Mb/s	483.85 kb/s	3.26 kb/s
eth0_receive	1.07 kb/s	268.15 Mb/s	111.86 Mb/s	1.82 kb/s
eth1_receive	803.87 b/s	391.63 kb/s	150.62 kb/s	1.07 kb/s
eth1_transmit	432.40 b/s	285.23 Mb/s	110.84 Mb/s	524.13 b/s

eth0: external link
eth1: internal link

Test A1: Upload the SDC2 dataset to the Rucio Data Lake

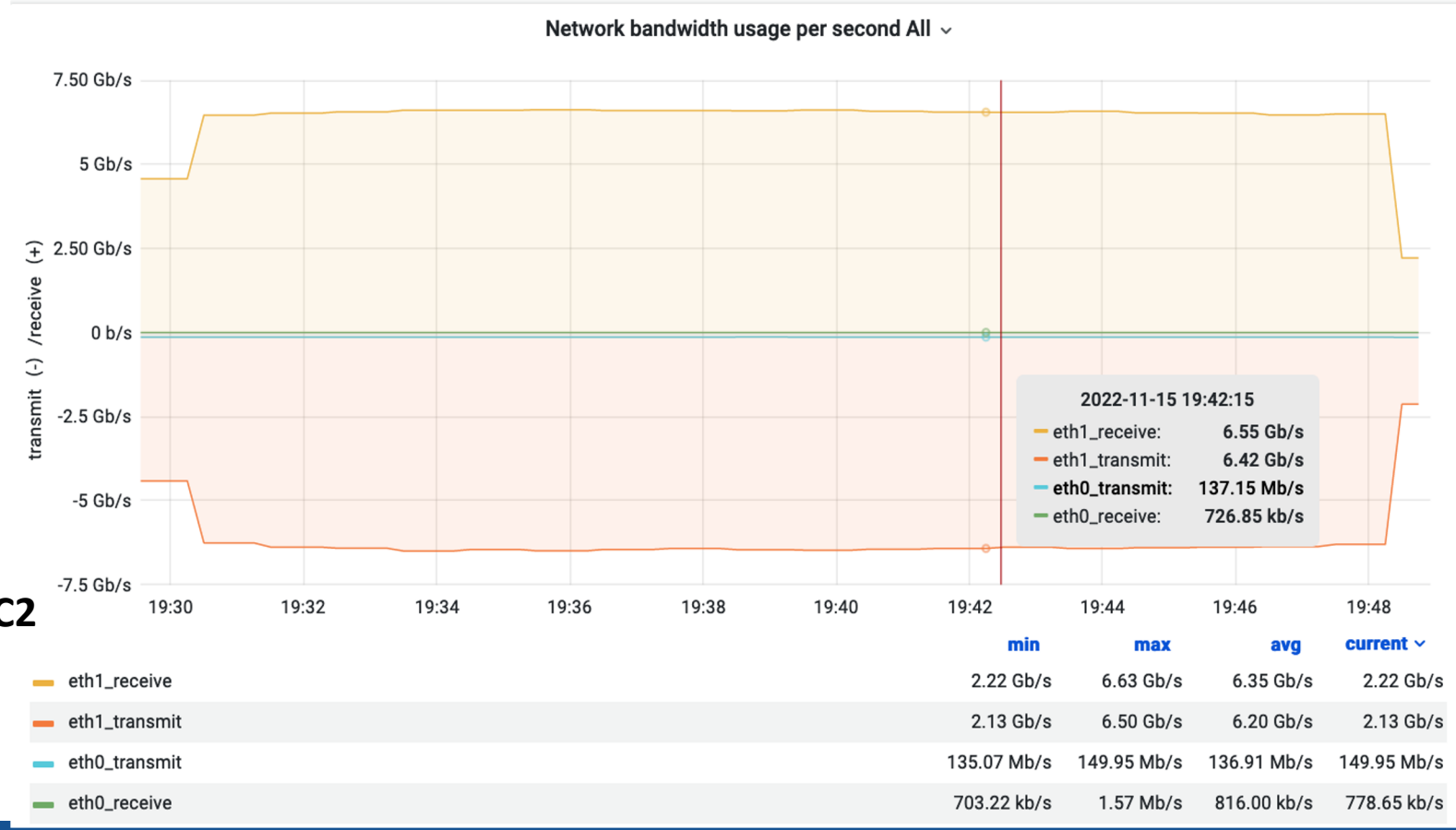
- Dataset was downloaded (rsync) to the storage volume of the RSE
- The RSE is deterministic → no possible just “register” the dataset in the Rucio namespace
- Rucio should copy the file in an specific local path

```
[centos@spsrc22t03-rse-rucio-centos7 dev]$ ls /storage/dteam/disk/dev/deterministic/coral/
00  07  10  16  1e  25  2c  34  3a  42  49  4f  55  61  68  72  78  7f  88  92  9a  a3  a8
01  09  11  17  1f  26  2d  35  3b  43  4a  50  56  62  6a  73  79  81  89  93  9d  a4  aa
03  0b  12  19  20  27  30  36  3c  45  4c  52  59  63  6e  74  7a  82  8d  95  9e  a5  ab
04  0d  14  1a  21  2a  31  37  3e  47  4d  53  5c  65  6f  75  7b  84  8f  96  a0  a6  ac
05  0f  15  1b  23  2b  32  39  3f  48  4e  54  5e  67  71  77  7e  87  91  98  a2  a7  ad
```

Test A1: Upload the SDC2 dataset to the Rucio Data Lake

- Dataset was downloaded (rsync) to the storage volume of the RSE
- The RSE is deterministic → no possible just “register” the dataset in the Rucio namespace
- Rucio should copy the file in an specific local path
- When uploading the file to Rucio DL, we expected a performance similar to when copying the file to a different path:

eth0: external link
eth1: internal link



MAKING A COPY OF THE SDC2 DATASET (862GB)

Time elapsed: ~19 m

Data rate avg : 6.35Gb/s

Test A1: Upload the SDC2 dataset to the Rucio Data Lake

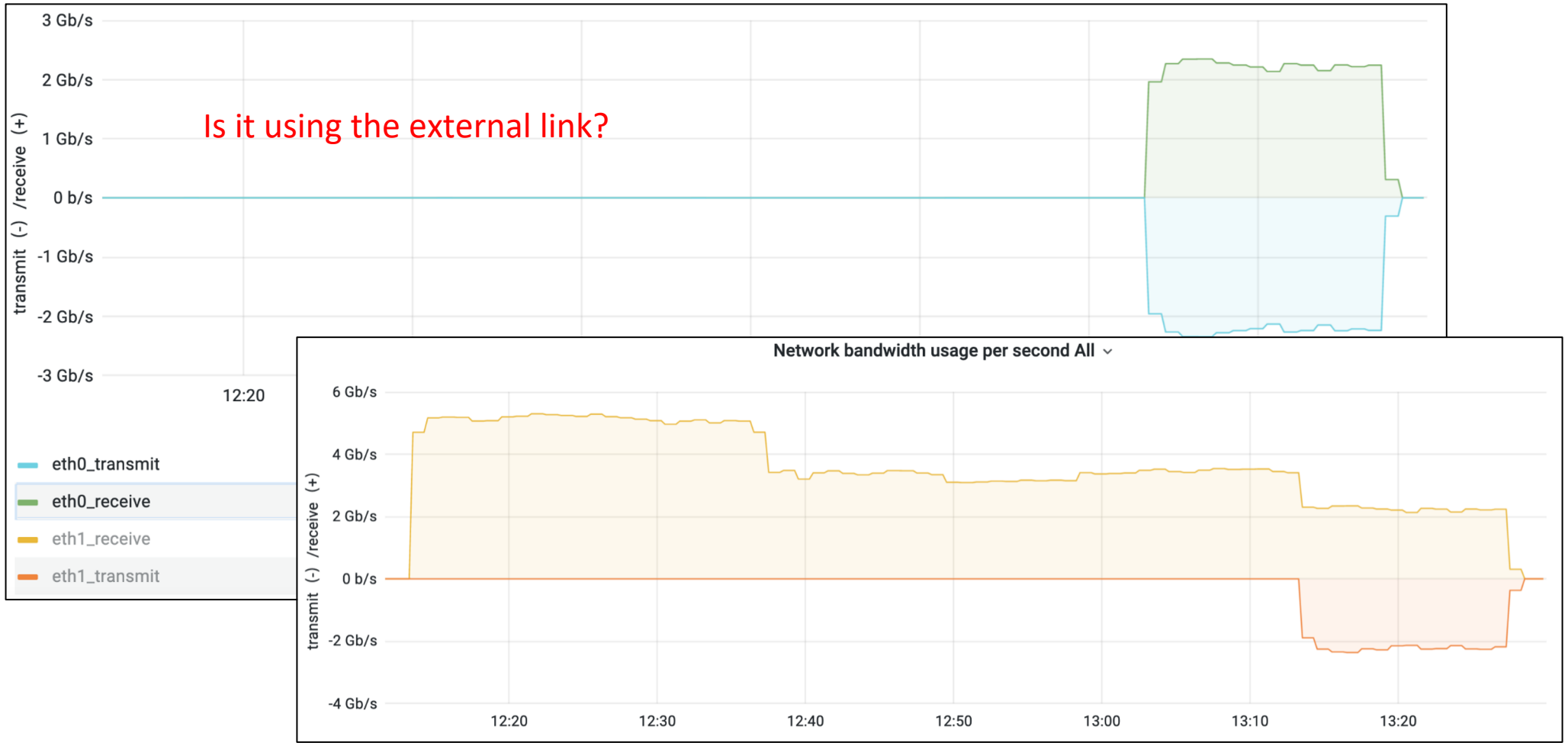
UPLOADING A 851 GB FILE TO RUCIO/SPSRC_STORM (SDC2 dataset)

Time elapsed: ~80 m - **Unsuccessful**

rucio upload --rse SPSRC_STORM --scope testing --register-after-upload /mnt/sky_full_v2.fits


```
2022-11-10 12:12:27,663 INFO    Trying upload with https to SPSRC_STORM
2022-11-10 12:26:28,380 WARNING Upload attempt failed
2022-11-10 12:26:28,381 INFO    Exception: RSE does not support requested operation.
Details: The requested service is not available at the moment.
Details: An unknown exception occurred.
Details: TRANSFER  ERROR: Copy failed with mode streamed, with error: (Neon): 404 Not Found
Traceback (most recent call last):
  File "/usr/local/lib/python3.6/site-packages/rucio/rse/protocols/gfal.py", line 484, in __gfal2_copy
    ret = ctx.filecopy(params, str(src), str(dest))
gfal2.GError: TRANSFER  ERROR: Copy failed with mode streamed, with error: (Neon): 404 Not Found
```

UTC



Rucio uses the external link to upload the file….

…. Is the data transfer using our Local Network?

eth0: external link
eth1: internal link

UPLOADING A 851 GB FILE TO RUCIO/SPSRC_STORM (SDC2 dataset)

Time elapsed: ~80 m - Data rate avg : 3,22Gb/s - **Unsuccessful**

	min	max	avg	current
eth0_transmit	2.65 kb/s	2.35 Gb/s	367.83 Mb/s	2.68 kb/s
eth0_receive	1.10 kb/s	2.35 Gb/s	367.88 Mb/s	1.16 kb/s
eth1_receive	1.07 kb/s	5.30 Gb/s	3.22 Gb/s	1.15 kb/s
eth1_transmit	432.40 b/s	2.36 Gb/s	368.58 Mb/s	644.00 b/s

eth1_receive

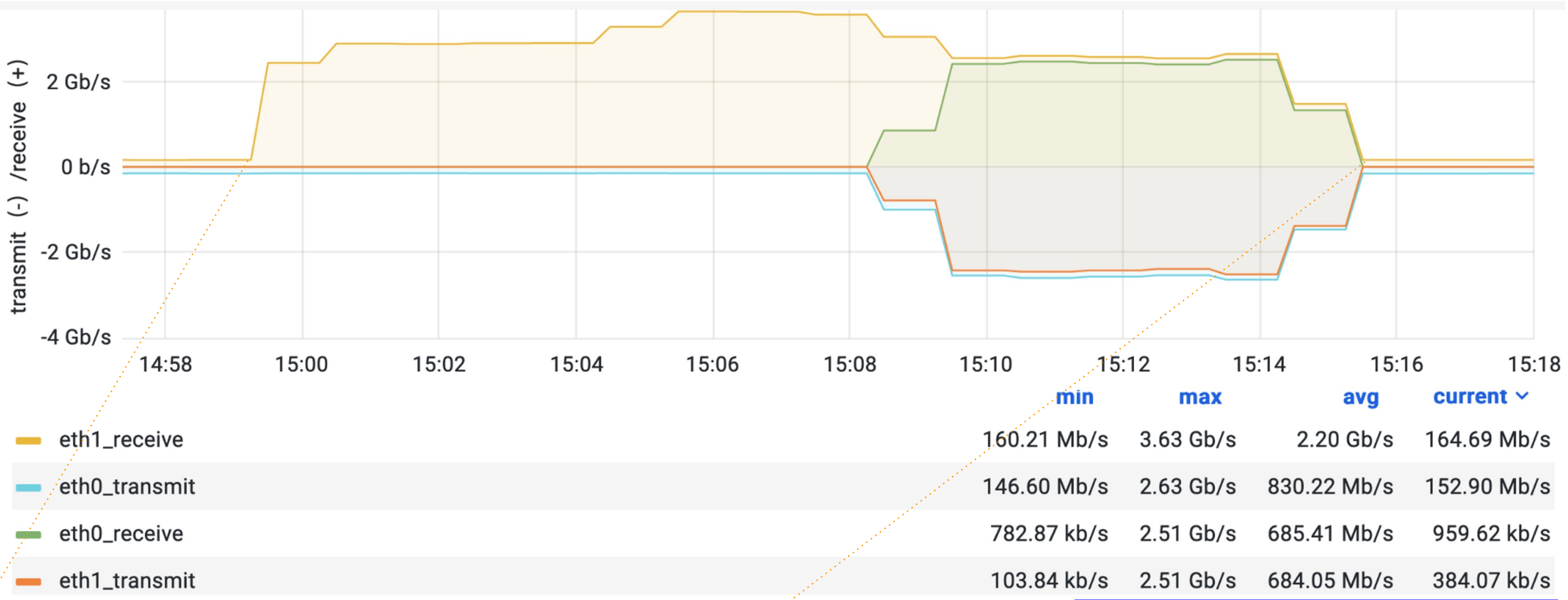
Time elapsed (s)	Data rate (Gb/s)	Total (GB)
4800	3,22	1932

eth0: external link
eth1: internal link

UPLOADING A 100GB FILE TO RUCIO/SPSRC_STORM

Time elapsed: ~15 m - **Successful**

Test A2



	min	max	avg	current
eth1_receive	160.21 Mb/s	3.63 Gb/s	2.20 Gb/s	164.69 Mb/s
eth0_transmit	146.60 Mb/s	2.63 Gb/s	830.22 Mb/s	152.90 Mb/s
eth0_receive	782.87 kb/s	2.51 Gb/s	685.41 Mb/s	959.62 kb/s
eth1_transmit	103.84 kb/s	2.51 Gb/s	684.05 Mb/s	384.07 kb/s

eth1_receive

Time elapsed (s)	Data rate (Gb/s)	Total (GB)
900	2,9	326,25

eth0: external link
eth1: internal link

3 times larger than data file size!!

Test A*x*

```
2022-11-23 13:59:47,813 INFO    Trying upload with https to STFC_STORM
2022-11-23 13:59:47,814 DEBUG   uploadclient.py Processing upload with the domain: wan
2022-11-23 13:59:47,814 DEBUG   logging.py      gfal.Default: connecting to storage
2022-11-23 13:59:47,817 DEBUG   uploadclient.py The PFN created from the LFN: https://srcdev.skatelescope.org:443/storm/sa/test_rse/dev/deterministic
/testing/08/4e/sky_full_v2_204G.fits
2022-11-23 13:59:47,817 DEBUG   logging.py      gfal.Default: checking if file exists https://srcdev.skatelescope.org:443/storm/sa/test_rse/dev/deter
ministic/testing/08/4e/sky_full_v2_204G.fits
2022-11-23 13:59:48,316 DEBUG   logging.py      gfal.Default: checking if file exists https://srcdev.skatelescope.org:443/storm/sa/test_rse/dev/deter
ministic/testing/08/4e/sky_full_v2_204G.fits.rucio.upload
2022-11-23 13:59:48,628 DEBUG   utils.py        put: Attempt 1
2022-11-23 13:59:48,628 DEBUG   logging.py      gfal.Default: uploading file from sky_full_v2_204G.fits to https://srcdev.skatelescope.org:443/storm/
sa/test_rse/dev/deterministic/testing/08/4e/sky_full_v2_204G.fits.rucio.upload
2022-11-23 14:06:48,631 DEBUG   logging.py      gfal.Default: gfal: cancelling all operations
2022-11-23 14:06:48,980 DEBUG   utils.py        put: Attempt failed 1
2022-11-23 14:06:48,980 DEBUG   utils.py        The requested service is not available at the moment.
Details: An unknown exception occurred.
Details: TRANSFER  ERROR: Copy failed with mode streamed, with error: (Neon): 404 Not Found
2022-11-23 14:06:48,980 DEBUG   logging.py      gfal.Default: uploading file from sky_full_v2_204G.fits to https://srcdev.skatelescope.org:443/storm/
sa/test_rse/dev/deterministic/testing/08/4e/sky_full_v2_204G.fits.rucio.upload
2022-11-23 14:13:48,982 DEBUG   logging.py      gfal.Default: gfal: cancelling all operations
2022-11-23 14:13:49,310 WARNING Upload attempt failed
2022-11-23 14:13:49,310 INFO    Exception: RSE does not support requested operation.
Details: The requested service is not available at the moment.
Details: An unknown exception occurred.
Details: TRANSFER  ERROR: Copy failed with mode streamed, with error: (Neon): 404 Not Found
```

transfer-timeout by default (==360s)

We discovered the effect of the transfer-timeout option

- How this timeout is affecting to the process?
- Not possible to disable this option using the command line API

Test B*x*: Using the Rucio Python API / **rucio-task-manager**

- Developed by SKAO team (R. Barnsley+)
- Automation
- Integration with SKAO Grafana monitoring

https://gitlab.com/ska-telescope/src/ska-rucio-task-manager

Tests to study performance scalability:

- How does data transfer performance scale to large dataset sizes? Are there any specific limitations?
- How does it scale to large number transfers? How does Rucio handle a high number of simultaneous transfers?
- How is the data transfer performance affected by the access protocols offered by the RSEs? FTS performance

Test B1: Uploading files with different size to all RSEs

```yaml
test-upload-dev:
  description: "Test uploading RSEs (SPSRC, KRSRC, JPSRC, IMPERIAL, CNSRC, CASRC) 3 files 200MB, 500MB, 1GB, 1.5GB, 2GB, 3GB, 4GB"
  module_name: "tasks.tests.upload"
  class_name: "TestUpload"
  enabled: true
  args:
  kwargs:
    n_files: 3
    sizes:
      - 200000000  # bytes
      - 500000000  # bytes
      - 1000000000  # bytes
      - 1500000000  # bytes
      - 2000000000  # bytes
      - 3000000000
      - 4000000000  # bytes
    lifetime: 3600 # seconds
    protocols:
      - https
    rses:
      - SPSRC_STORM
      - KRSRC_STORM
      - JPSRC_STORM
      - IMPERIAL
      - CNSRC_STORM
      - CNAF
      - CHSRC_XRD
      - CASRC_XRD
    scope: testing
    output:
      databases:
        - type: es
          uri:  https://monit.srcdev.skao.int/elastic/
          index: rucio-task-manager.skao-dev.tasks.tests.coral-3f-200mb-500mb-1gb-1.5gb-2gb-3gb-4gb-spsrc-krsrc-jpsrc-imperial-cnsrc-cnaf-
          chsrc-casrc_ok
          auth:
```

Test B1: Uploading files with different size to all RSEs

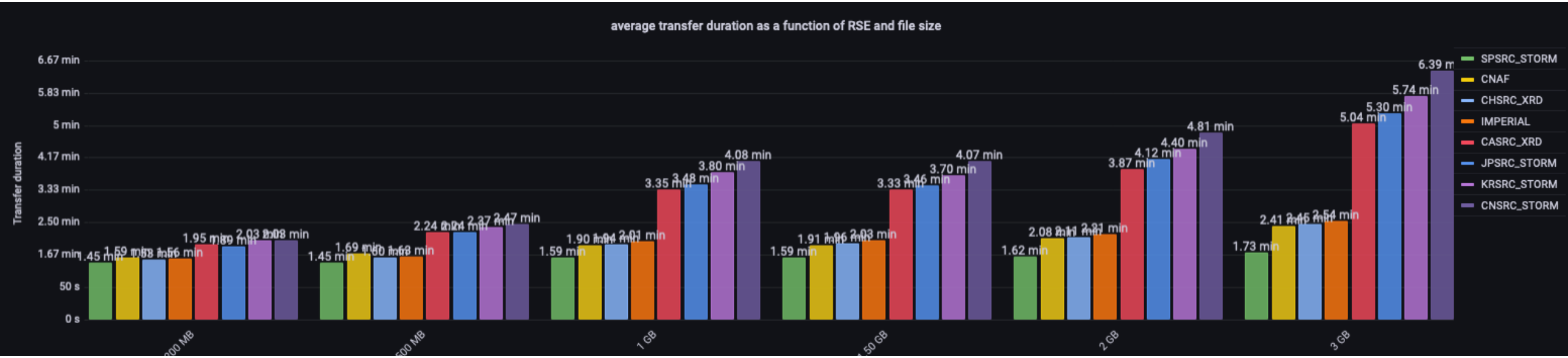
Average transfer rate as a function of RSE and file size

average transfer rate as a function of RSE and file size

file size \ RSE	SPSRC_STORM	KRSRC_STORM	JPSRC_STORM	IMPERIAL	CNSRC_STORM	CNAF	CHSRC_XRD	CASRC_XRD
200 MB	2.30 MB/s	1.65 MB/s	1.76 MB/s	2.14 MB/s	1.65 MB/s	2.09 MB/s	2.17 MB/s	1.71 MB/s
500 MB	5.73 MB/s	3.52 MB/s	3.73 MB/s	5.11 MB/s	3.38 MB/s	4.97 MB/s	5.20 MB/s	3.72 MB/s
1 GB	10.6 MB/s	4.83 MB/s	5.25 MB/s	8.56 MB/s	4.60 MB/s	8.96 MB/s	8.84 MB/s	5.41 MB/s
1.50 GB	15.7 MB/s	6.75 MB/s	7.23 MB/s	12.3 MB/s	6.15 MB/s	13.1 MB/s	12.8 MB/s	7.50 MB/s
2 GB	20.6 MB/s	7.58 MB/s	8.09 MB/s	15.1 MB/s	6.93 MB/s	16.0 MB/s	15.8 MB/s	8.61 MB/s
3 GB	28.9 MB/s	8.71 MB/s	9.43 MB/s	19.7 MB/s	7.83 MB/s	20.8 MB/s	20.5 MB/s	9.92 MB/s
Mean	14.0 MB/s	5.51 MB/s	5.92 MB/s	10.5 MB/s	5.09 MB/s	11.0 MB/s	10.9 MB/s	6.15 MB/s

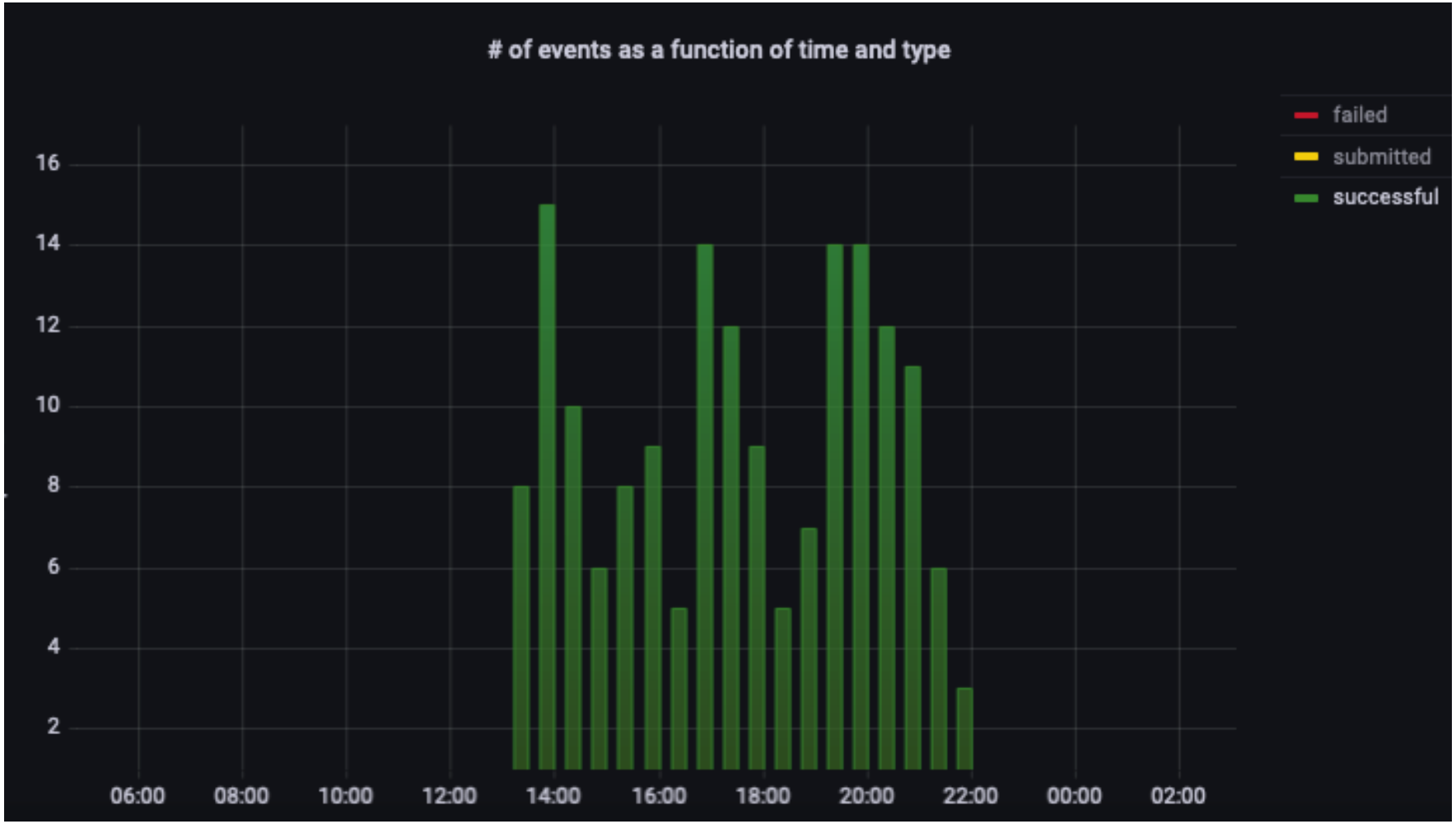
Test B1: Uploading files with different size to all RSEs

Average transfer duration as a function of RSE and file size

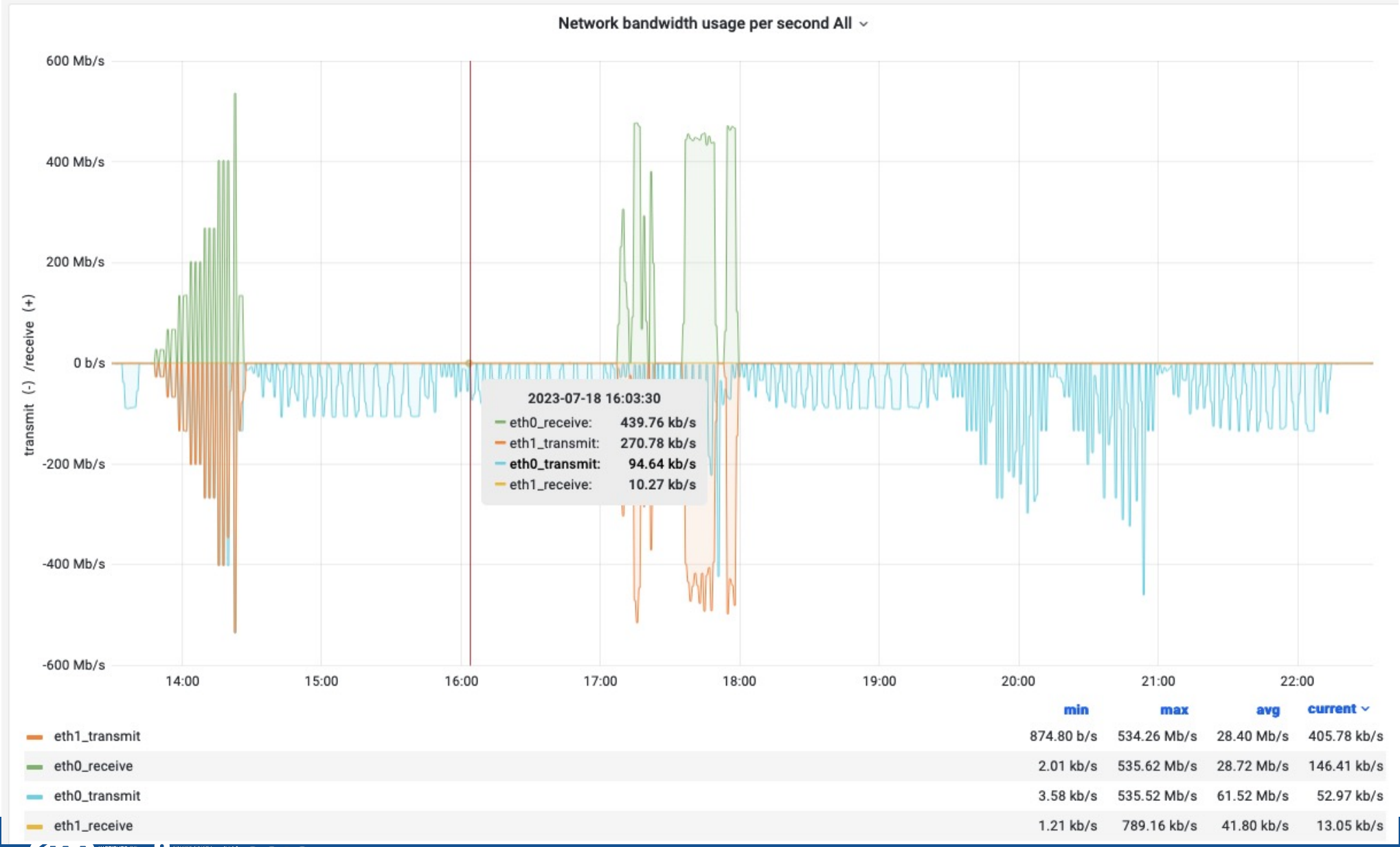


Test B1: Uploading files with different size to all RSEs

Number of events as a function of time and type



Test B1: Uploading files with different size to all RSEs



Ongoing / Next Tests ”B”

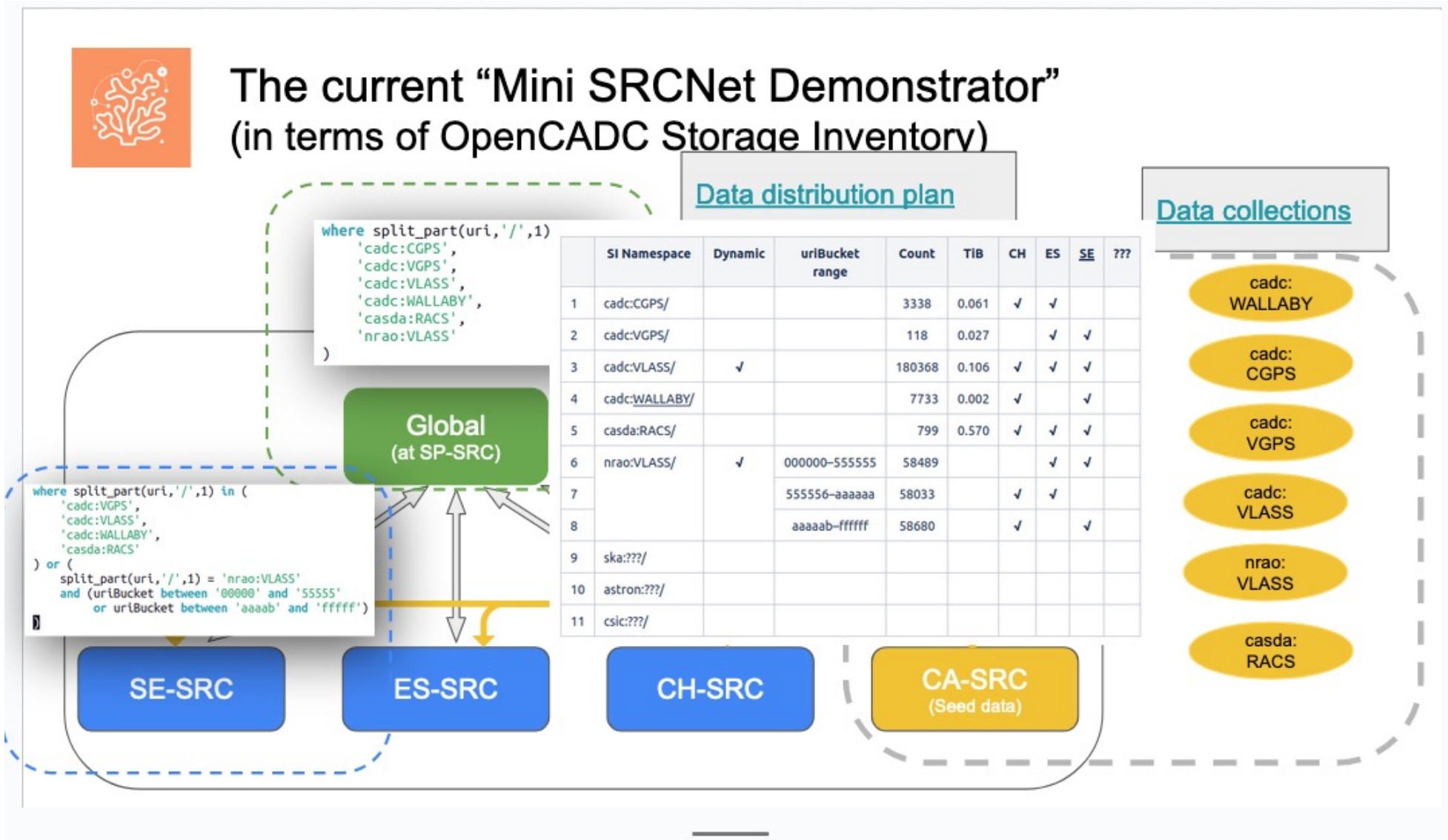
- Size scalability (file sizes in the scale of hundreds of GBs) with **replication rules**

  →Avoiding user token expiration (previous tests executed with 12h tokens)

- Scalability in the number of files

- Transfers between RSEs with different access protocols

OpenCADC Storage Inventory



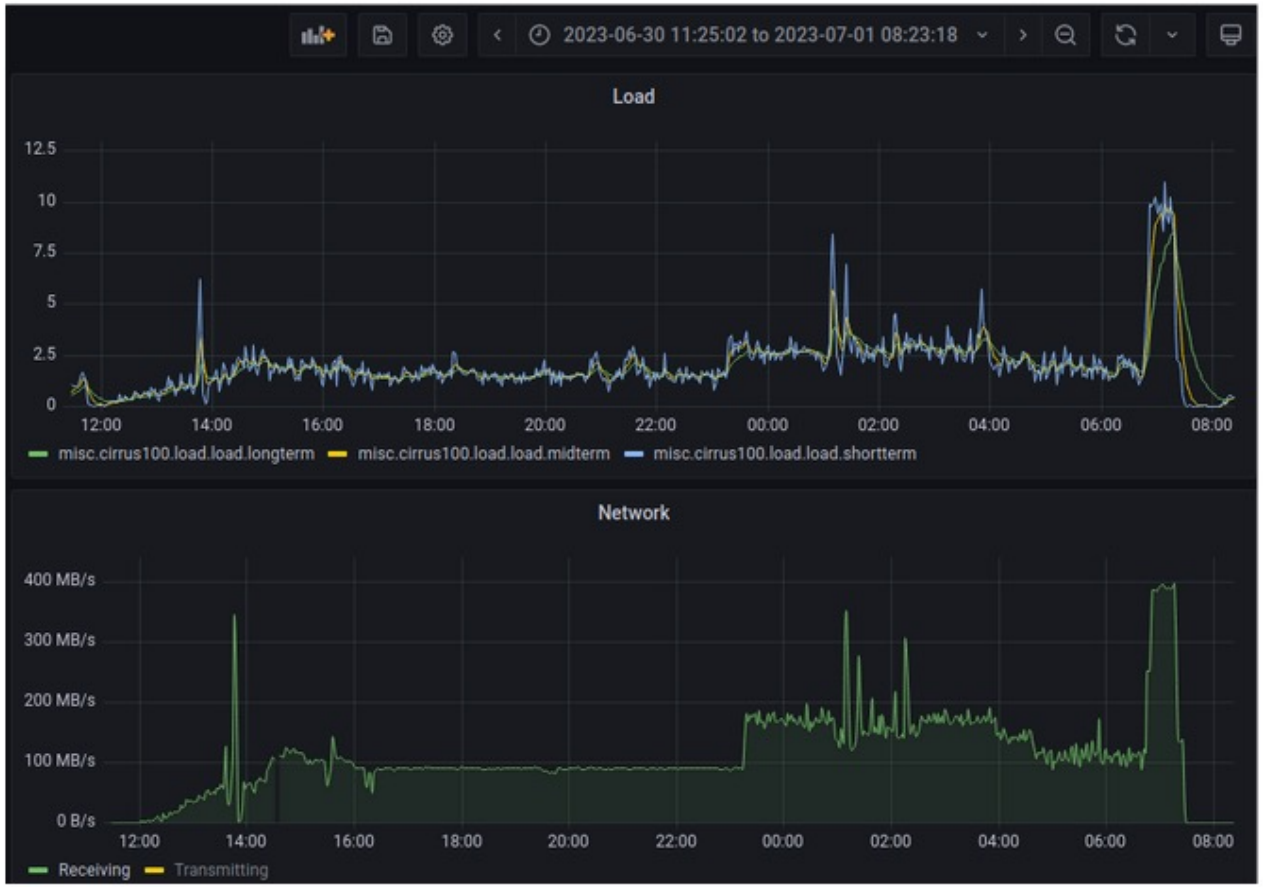
	SI Namespace	Dynamic	uriBucket range	Count	TiB	CH	ES	SE	???
1	cadc:CGPS/			3338	0.061	√	√		
2	cadc:VGPS/			118	0.027		√	√	
3	cadc:VLASS/	√		180368	0.106	√	√	√	
4	cadc:WALLABY/			7733	0.002	√		√	
5	casda:RACS/			799	0.570	√	√	√	
6	nrao:VLASS/	√	000000–555555	58489			√	√	
7			555556–aaaaaa	58033		√	√		
8			aaaaab–ffffff	58680		√		√	
9	ska:???/								
10	astron:???/								
11	csic:???/								

Credit: Franz Kirsten

OpenCADC Storage Inventory

Some stats on data volumes/transfer rates/system load

Full data set: 10.471 TB (368051 files)
Spain: 7.227 TB (301586 files)
Switzerland: 7.213 TB (309367 files)
Sweden: 7.195 TB (306609 files)



Credit: Franz Kirsten

Questions

- **Deterministic Rucio Storage Endpoint (RSE)**
  - Disadvantage: Registry of files not possible
  - Advantages: ??

- **Low transfers SPSRC – SPSRC:**
  - Rucio issue → Use non-deterministc RSE
  - Not Rucio issue → identify the problem in our internal network

- **Is a "Virtual DTN " a good option**?
  - If not, how to connect the physical DTN with the openstack platform?

- **Is there a bottleneck in our *virtual* Openstack network?**

- **Learn how to improve our network configuration**

Rucio Tests

Network Tests

Thank you!