

# LTDSF S3



Poznan Supercomputing  
and Networking Center

# What is S3? - HTTP Object Storage API

1. S3 - Simple Storage Service
2. Created by AWS and one of their most successful products
3. Object storage not a filesystem
  - a. Files represented as Key Value Pairs
4. Ideal for data storage and transfer
5. PSNC uses open source S3 API implementation (Ceph)
6. LTDSF S3 is hosted in Poznan Poland
  - a. <https://eurofusion-2.s3.storage.psnc.pl>



Poznan Supercomputing  
and Networking Center

# S3 Basics

## 1. Users

- a. Person uploading/downloading data

## 2. Buckets

- a. Place to store our files

## 3. Objects

- a. files stored in a bucket



Poznan Supercomputing  
and Networking Center

# S3 Basics - Bucket

1. Bucket is like a huge directory
  - a. It stores our files/objects
  - b. Contains list of object keys
  - c. It can have custom access permissions (advanced usage)
  - d. Bucket names are unique to s3 instance



Poznan Supercomputing  
and Networking Center

## S3 Basics - object

1. Key-Value pair in S3 bucket
2. Key is like filepath in a normal filesystem
3. Value is the file content



Poznan Supercomputing  
and Networking Center

# S3 basics - object key explained

1. Example object
  - a. `s3://eprycka/dir1/dir13/dir131/file-0`
2. Let's break this down
3. “s3://” - prefix used by s3cmd client to signify it's s3 path
4. “eprycka” - is a name of s3 bucket our object is in
5. “dir1/dir13/dir131/file-0” - is the Key of our object
6. “dir1/dir13/dir131/” - is an object prefix (acts like directory)

Notice that Object Key has “/” in it's path.

These are part of the string, not a real directories.



Poznan Supercomputing  
and Networking Center

# S3 basics - object prefixes

1. Our object from previous example looked like this  
“s3://eprycka/dir1/dir13/dir131/file-0”
2. “dir1/dir13/dir131/” wasn’t made out of directories but it acts like a directory
3. We call these pseudo-directories or object prefixes
4. Object prefixes can be listed like directories
5. But they disappear once you delete the object inside them
6. There are no empty directories in S3 bucket
7. We will show examples later.



Poznan Supercomputing  
and Networking Center

# S3 basics - S3 clients

1. API standard published by Amazon Web Services
2. S3 API client libraries available for all popular programming languages
3. Many popular clients capable of working with data on S3
4. Clients we recommend:
  - a. s3cmd ( good for every day use, we will use it later )
  - b. rclone ( good for bulk data transfer, examples in PDF)



Poznan Supercomputing  
and Networking Center

# S3 basics - workflow examples

1. Archiving data
  - a. Uploading older data to s3 bucket
2. Backing up your home directory
3. Storing results on s3
  - a. Upload command at the end of your HPC job can upload results to S3
4. Storing datasets in S3
  - a. Storing larger datasets on s3
  - b. Downloading part of larger dataset to a local filesystem when needed
  - c. Could be done at the beginning of your HPC job
5. Native S3 data access
  - a. You can write application which accesses data from s3 directly (s3 API libraries)
  - b. May not be suitable for certain workload because of IOPS and BANDWIDTH limitations
  - c. It's Long Term Data Storage not scratch storage ;)

# S3 and differences from POSIX filesystem



Poznan Supercomputing  
and Networking Center

## 1. S3 is not a filesystem

- a. No file locking
- b. No partial writes
- c. No partial file overwrites
- d. No -rwxrwxrwx file permissions
- e. No POSIX

## 2. You shouldn't use clients which mount s3 buckets as filesystem

- a. Some clients implement it but using them is not a good idea.
- b. You may loose data if client crashes.
- c. Your client may get stuck if you execute shell `ls` on a very large bucket
- d. We do not support any this use case.

# S3 Advanced usage (not in scope of this meeting)

1. **Bucket policies** can make s3 data publicly readable (public file hosting)
2. **Bucket policies** can grant other user access to your bucket  
(sharing data between users or teams)
3. **Versioned buckets** protect against accidental data deletion.
4. **Object Lock** can make sure noone will be able to delete your data (**Ransomware protection**)
5. **Bucket Lifecycles** can delete objects in parts of buckets after X days  
( Great for cleanup of temporary files or older versions of objects )
6. S3 can also be used to host static websites



Poznan Supercomputing  
and Networking Center