C.E.C.I

Consortium des Equipements
de Calcul Intensif
en Fédération Wallonie-Bruxelles

# Introduction to
# Scientific Data Management

damien.francois@uclouvain.be

November 2020

http://www.ceci-hpc.be/training.html

UCL Université catholique de Louvain

INSTITUT DE CALCUL INTENSIF ET DE STOCKAGE DE MASSE

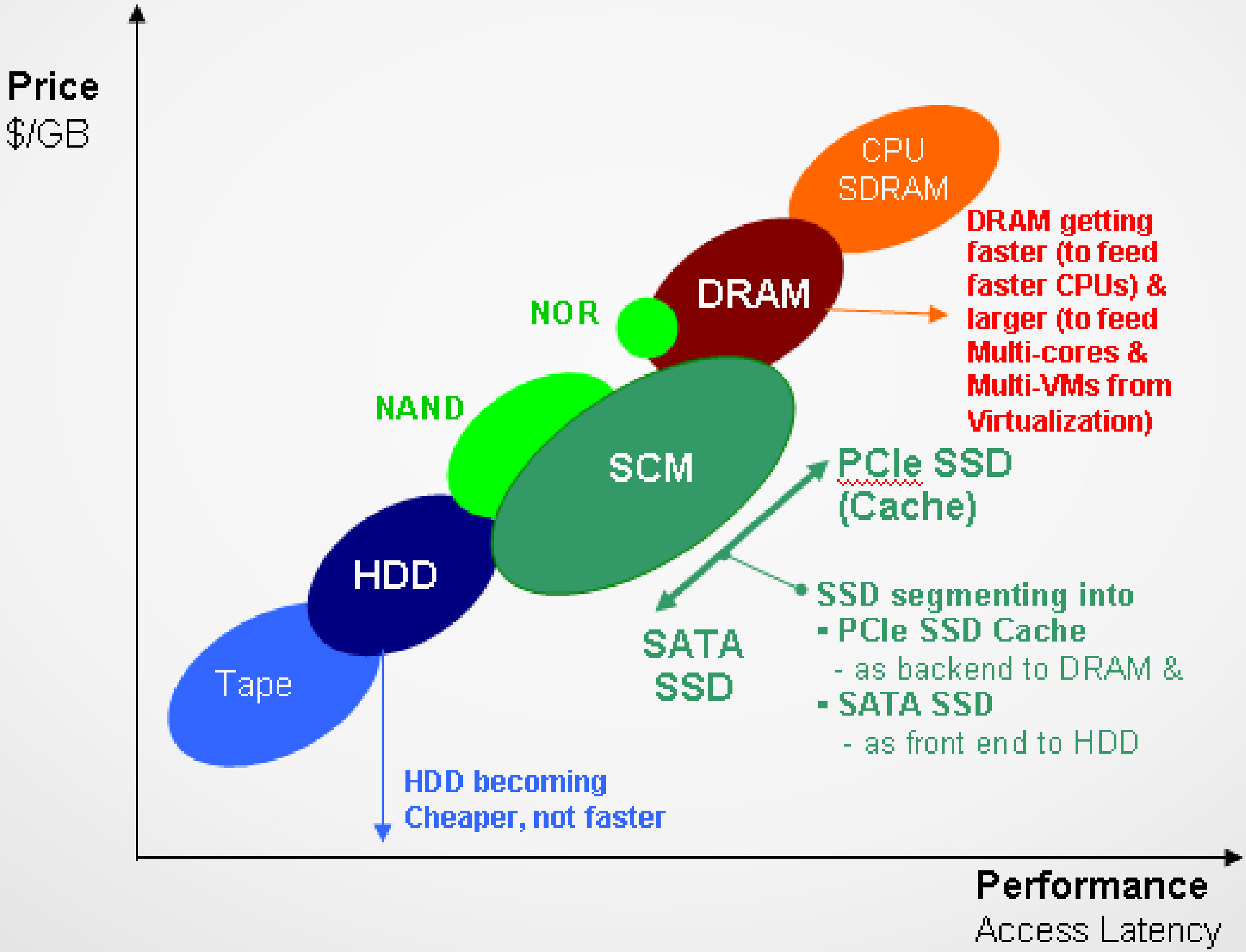
# Goal of this session:

"Share tools, tips and tricks related to the
**storage**, **transfer**, and **sharing**
of scientific data"

# 1. Data storage

Filesystems – Object stores – Databases

# Storage Technologies



DRAM getting faster (to feed faster CPUs) & larger (to feed Multi-cores & Multi-VMs from Virtualization)

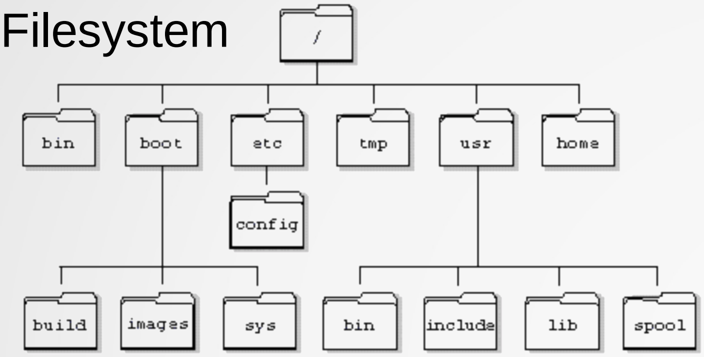
SSD segmenting into
- PCIe SSD Cache
  - as backend to DRAM &
- SATA SSD
  - as front end to HDD

HDD becoming Cheaper, not faster

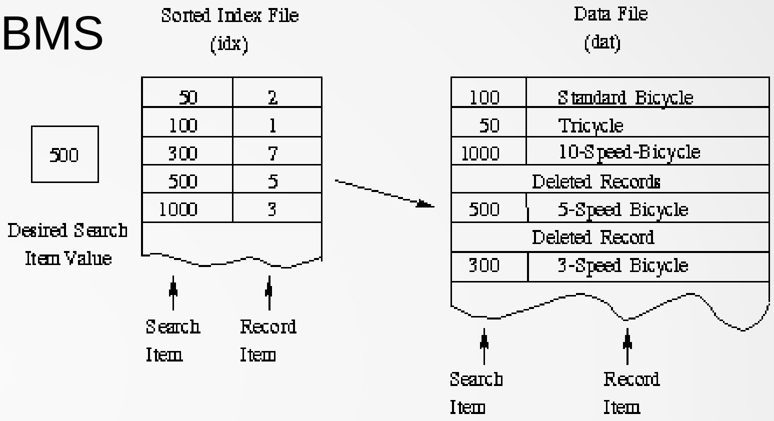
http://www.slideshare.net/IMEXresearch/ss-ds-ready-for-enterprise-cloud

# Storage paradigms

## Filesystem



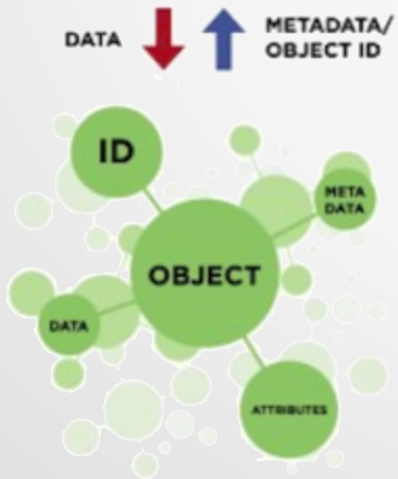
## RDBMS



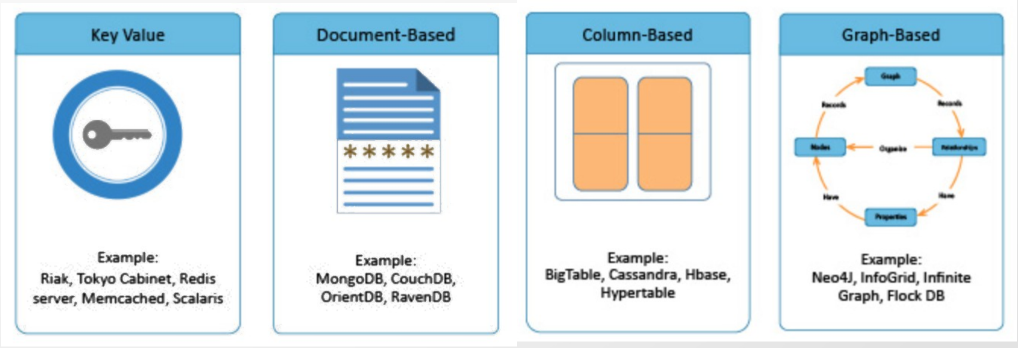
| | |
|---|---|
| 50 | 2 |
| 100 | 1 |
| 300 | 7 |
| 500 | 5 |
| 1000 | 3 |

| | |
|---|---|
| 100 | Standard Bicycle |
| 50 | Tricycle |
| 1000 | 10-Speed-Bicycle |
| Deleted Records | |
| 500 | 5-Speed Bicycle |
| Deleted Record | |
| 300 | 3-Speed Bicycle |

## Objects store

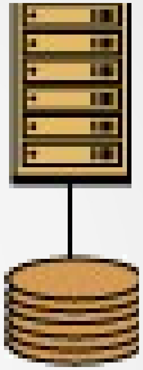


## NoSQL


Key Value
Example:
Riak, Tokyo Cabinet, Redis server, Memcached, Scalaris
Document-Based
Example:
MongoDB, CouchDB, OrientDB, RavenDB
Column-Based
Example:
BigTable, Cassandra, Hbase, Hypertable
Graph-Based
Example:
Neo4J, InfoGrid, Infinite Graph, Flock DB

# 1.1 Filesystems

# (local) Filesystems



**Generation 0:** No system at all. There was just an arbitrary stream of data. Think punchcards, data on audiocassette, Atari 2600 ROM carts.

**Generation 1:** Early random access. Here, there are multiple named files on one device with no folders or other metadata. Think Apple ][ DOS (but not ProDOS!) as one example.

**Generation 2:** Early organization (aka folders). When devices became capable of holding hundreds of files, better organization became necessary. We're referring to TRS-DOS, Apple //c ProDOS, MS-DOS FAT/FAT32, etc.

**Generation 3:** Metadata—ownership, permissions, etc. As the user count on machines grew higher, the ability to restrict and control access became necessary. This includes AT&T UNIX, Netware, early NTFS, etc.

**Generation 4:** Journaling! This is the killer feature defining all current, modern filesystems—ext4, modern NTFS, UFS2, XFS, you name it. Journaling keeps the filesystem from becoming inconsistent in the event of a crash, making it much less likely that you'll lose data, or even an entire disk, when the power goes off or the kernel crashes.

**Generation 5:** Copy on Write snapshots, Per-block checksumming, Volume management, Far-future scalability, Asynchronous incremental replication, Online compression. Generation 5 filesystems are Btrfs and ZFS.

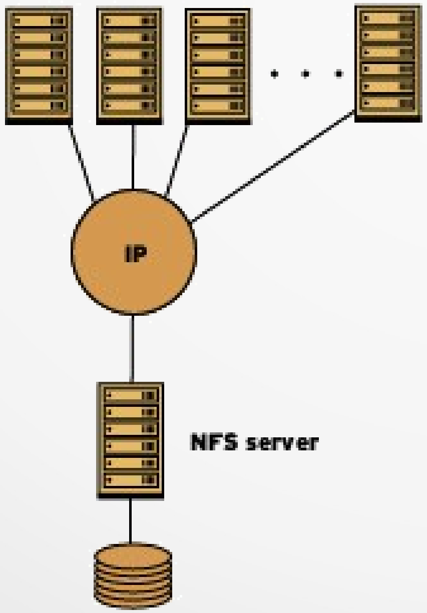
Typical usage: Operating system & Local scratch space

http://arstechnica.com/information-technology/2014/01/bitrot-and-atomic-cows-inside-next-gen-filesystems/

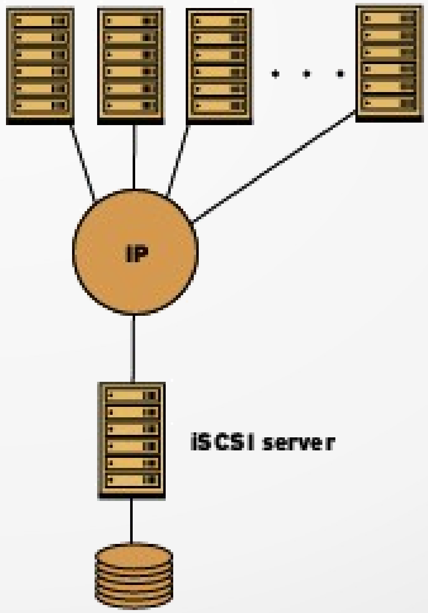
# Network filesystem

One source many consumers

NAS: ex. NFS



SAN: ex. GFS2

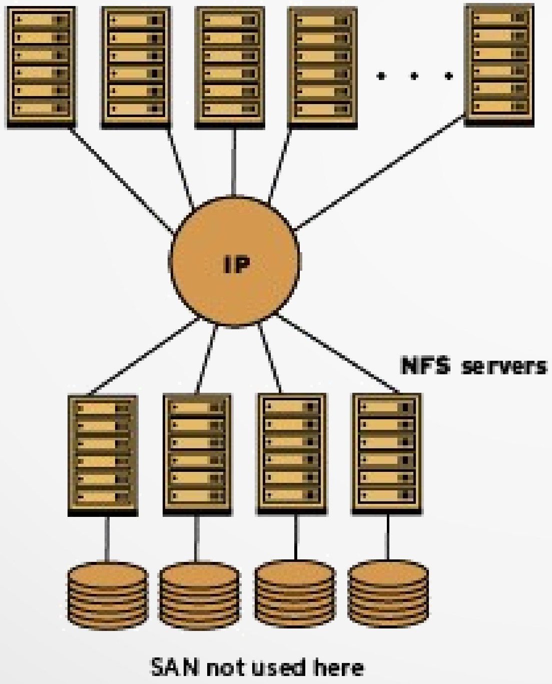


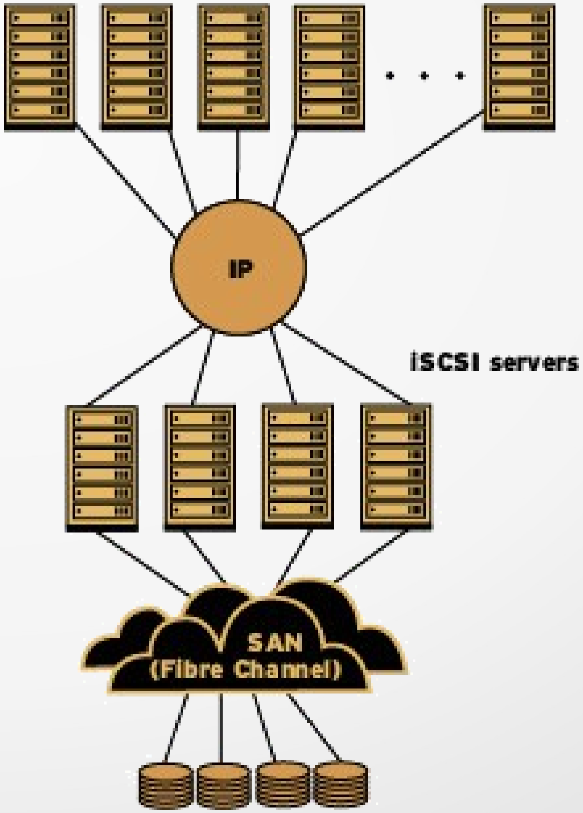
Typical usage: Home directories, Mass storage

Pictures from https://www.redhat.com/magazine/008jun05/features/gfs_nfs/

# Parallel / distributed filesystem

Many sources many consumers
ex: Lustre, GPFS, BeeGFS, GlusterFS

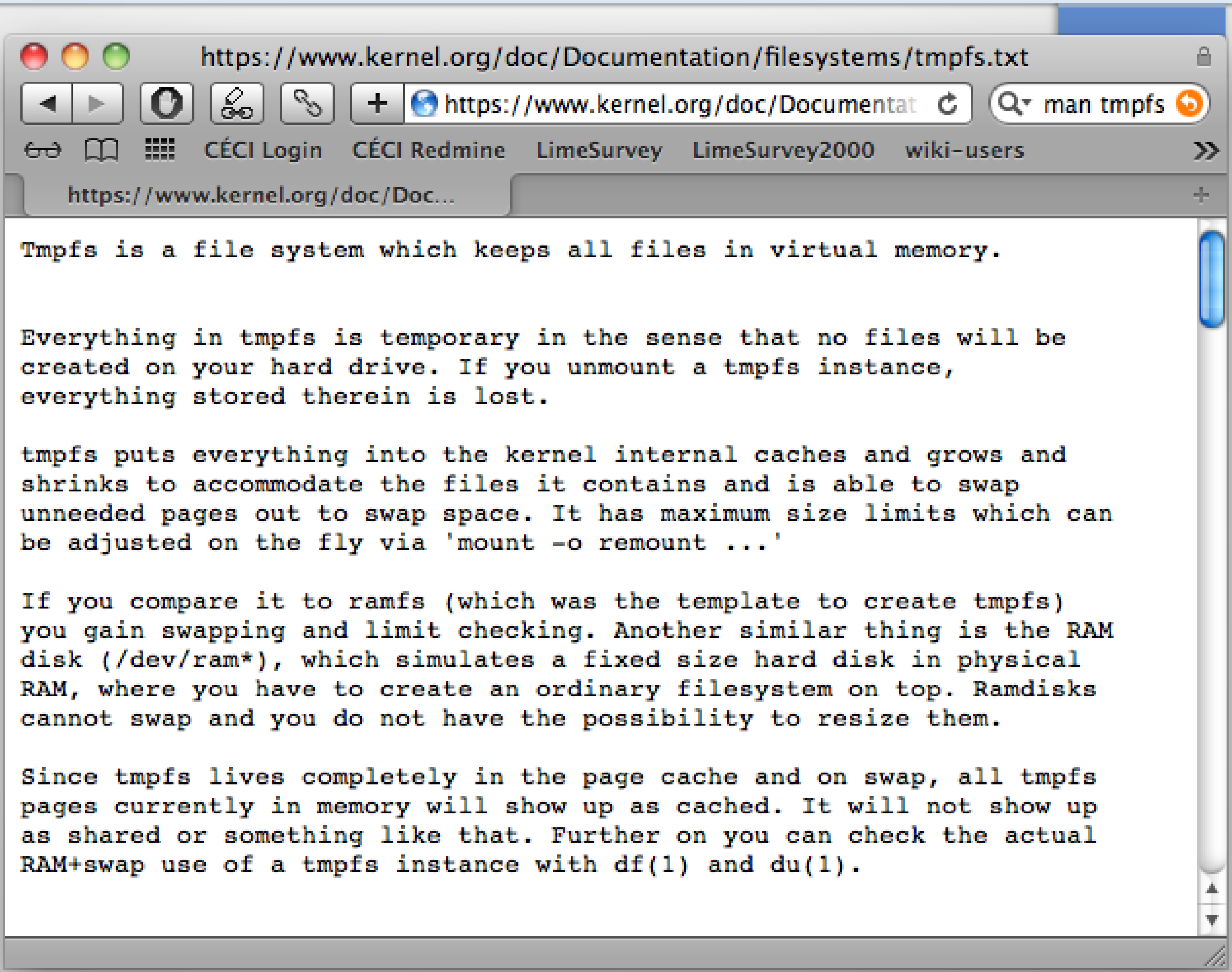




Typical usage: Global Scratch space

Pictures from https://www.redhat.com/magazine/008jun05/features/gfs_nfs/

# Special filesystems – in memory


https://www.kernel.org/doc/Documentation/filesystems/tmpfs.txt
https://www.kernel.org/doc/Documentat
man tmpfs
CÉCI Login
CÉCI Redmine
LimeSurvey
LimeSurvey2000
wiki-users
https://www.kernel.org/doc/Doc...

Tmpfs is a file system which keeps all files in virtual memory.

Everything in tmpfs is temporary in the sense that no files will be created on your hard drive. If you unmount a tmpfs instance, everything stored therein is lost.

tmpfs puts everything into the kernel internal caches and grows and shrinks to accommodate the files it contains and is able to swap unneeded pages out to swap space. It has maximum size limits which can be adjusted on the fly via 'mount -o remount ...'

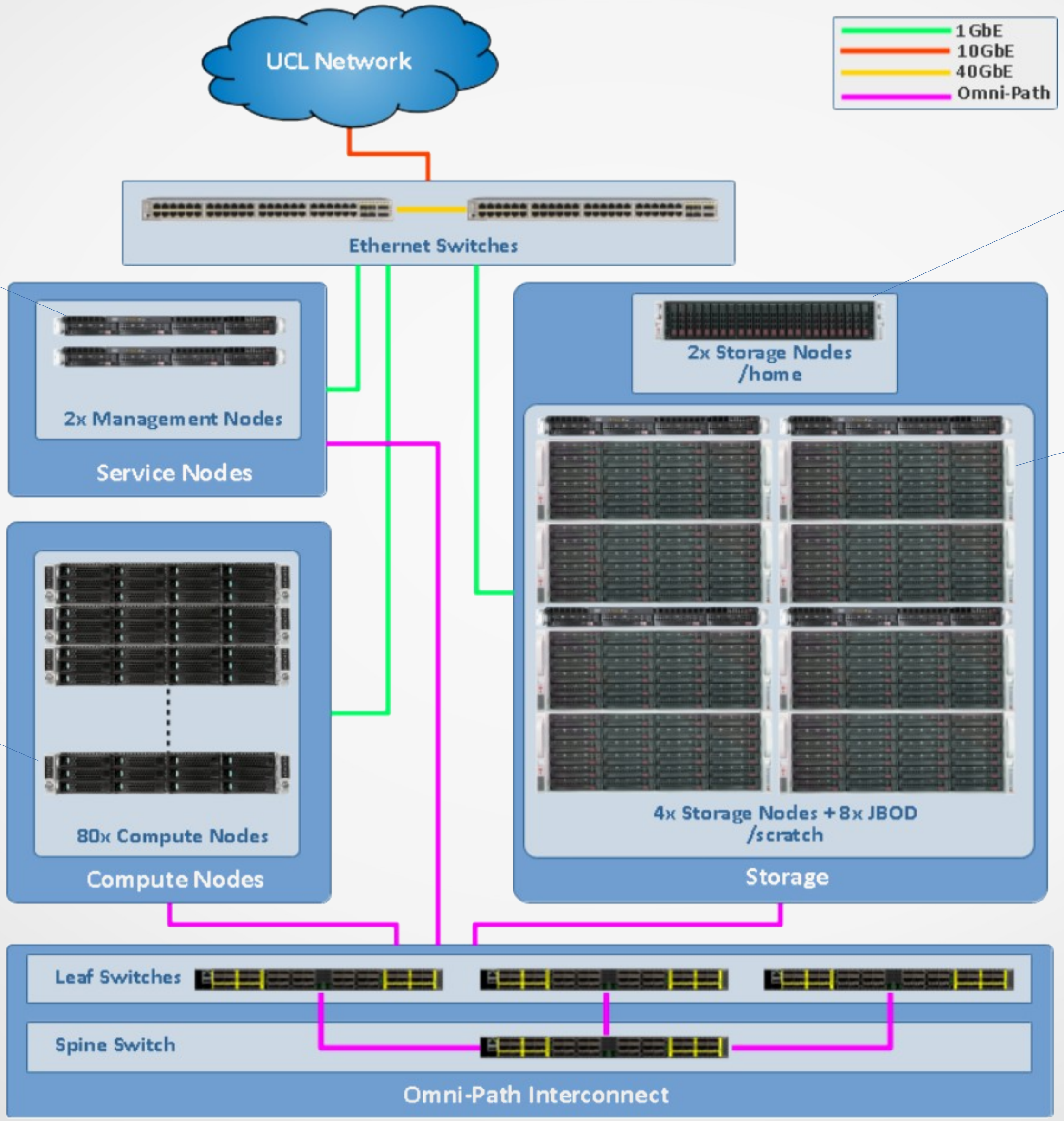
If you compare it to ramfs (which was the template to create tmpfs) you gain swapping and limit checking. Another similar thing is the RAM disk (/dev/ram*), which simulates a fixed size hard disk in physical RAM, where you have to create an ordinary filesystem on top. Ramdisks cannot swap and you do not have the possibility to resize them.

Since tmpfs lives completely in the page cache and on swap, all tmpfs pages currently in memory will show up as cached. It will not show up as shared or something like that. Further on you can check the actual RAM+swap use of a tmpfs instance with df(1) and du(1).

Typical usage: Temporary filesystems

# Filesystems on Lemaitre3

XFS
2TB
DRBD



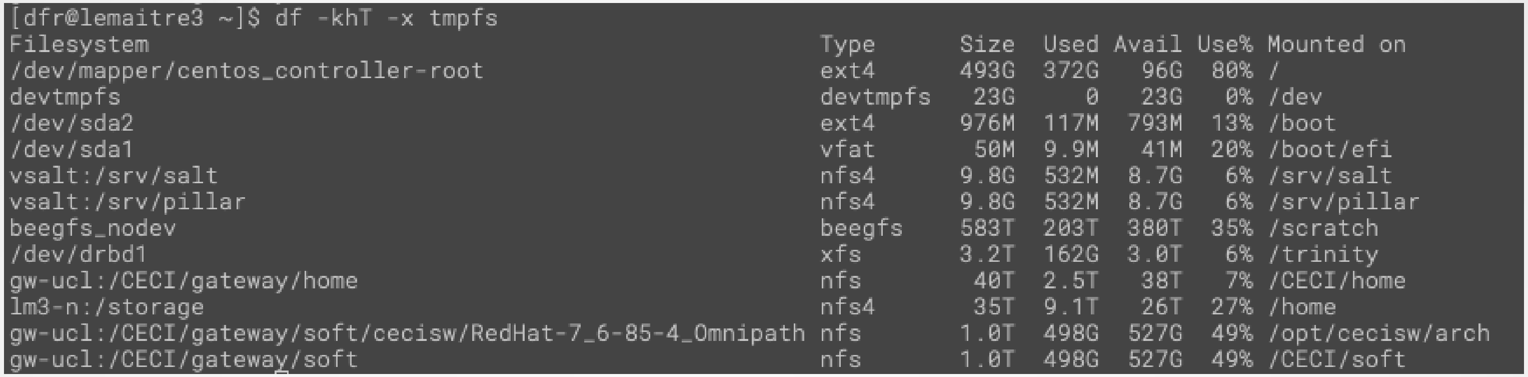
NFS
35TB
ZRAID2

BeeGFS
580TB
RAID6

XFS
220GB

# Filesystems on Lemaitre3

Volume

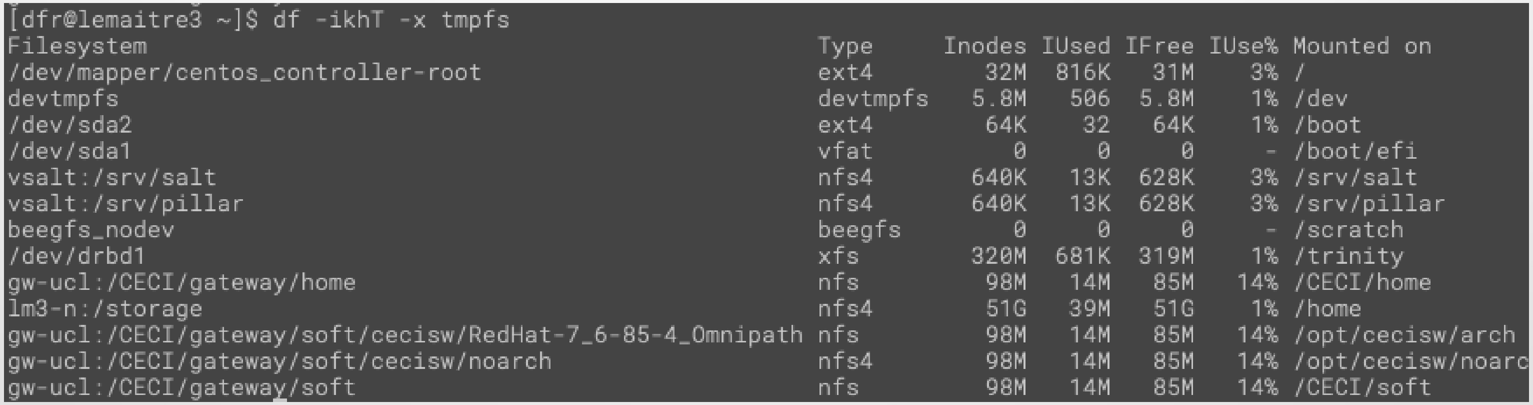


```
[dfr@lemaitre3 ~]$ df -khT -x tmpfs
Filesystem                                           Type      Size  Used Avail Use% Mounted on
/dev/mapper/centos_controller-root                   ext4      493G  372G   96G  80% /
devtmpfs                                             devtmpfs   23G     0   23G   0% /dev
/dev/sda2                                            ext4      976M  117M  793M  13% /boot
/dev/sda1                                            vfat       50M  9.9M   41M  20% /boot/efi
vsalt:/srv/salt                                      nfs4      9.8G  532M  8.7G   6% /srv/salt
vsalt:/srv/pillar                                    nfs4      9.8G  532M  8.7G   6% /srv/pillar
beegfs_nodev                                         beegfs    583T  203T  380T  35% /scratch
/dev/drbd1                                           xfs       3.2T  162G  3.0T   6% /trinity
gw-ucl:/CECI/gateway/home                            nfs        40T  2.5T   38T   7% /CECI/home
lm3-n:/storage                                       nfs4       35T  9.1T   26T  27% /home
gw-ucl:/CECI/gateway/soft/cecisw/RedHat-7_6-85-4_Omnipath nfs  1.0T  498G  527G  49% /opt/cecisw/arch
gw-ucl:/CECI/gateway/soft                            nfs       1.0T  498G  527G  49% /CECI/soft
```

Source:
- /dev/sd… → local disk
- /dev/mapper… → LVM
- <machine>:<path> → NFS
- other (e.g. beegfs_nodev) → specific filesystem

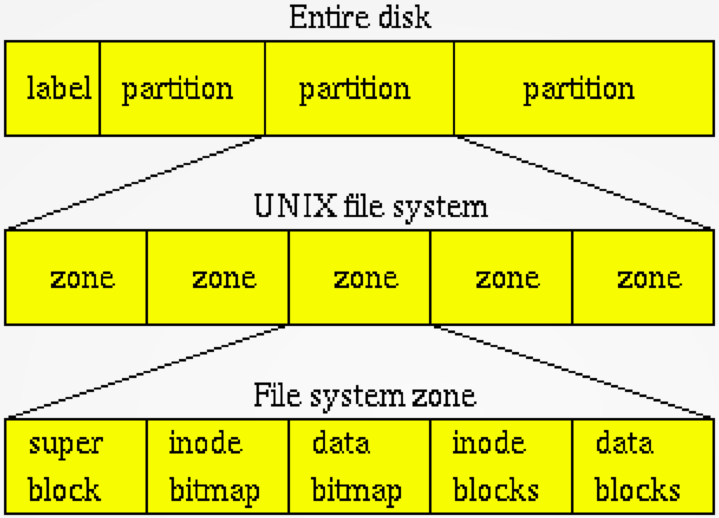
Local mount point

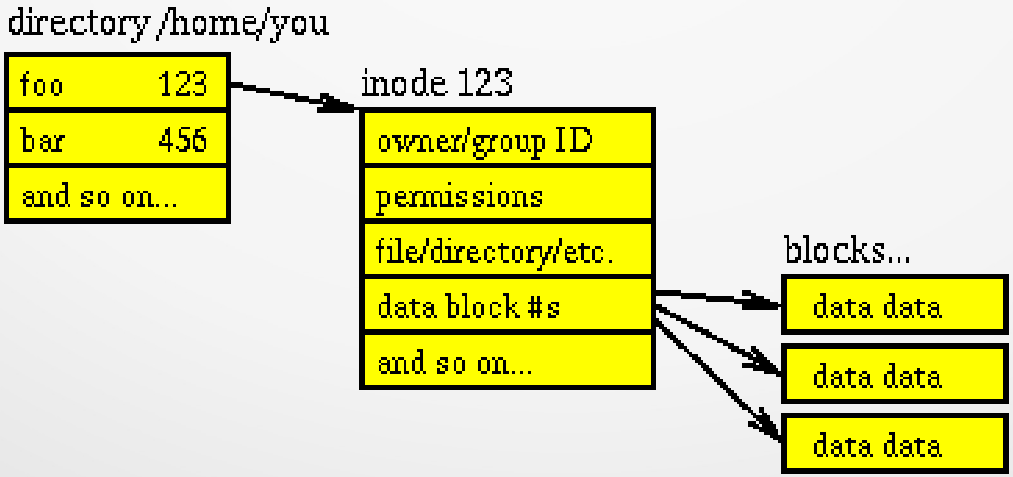
Inodes



```
[dfr@lemaitre3 ~]$ df -ikhT -x tmpfs
Filesystem                                           Type     Inodes IUsed IFree IUse% Mounted on
/dev/mapper/centos_controller-root                   ext4        32M  816K   31M    3% /
devtmpfs                                             devtmpfs   5.8M   506  5.8M    1% /dev
/dev/sda2                                            ext4        64K    32   64K    1% /boot
/dev/sda1                                            vfat          0     0     0     - /boot/efi
vsalt:/srv/salt                                      nfs4       640K   13K  628K    3% /srv/salt
vsalt:/srv/pillar                                    nfs4       640K   13K  628K    3% /srv/pillar
beegfs_nodev                                         beegfs        0     0     0     - /scratch
/dev/drbd1                                           xfs        320M  681K  319M    1% /trinity
gw-ucl:/CECI/gateway/home                            nfs         98M   14M   85M   14% /CECI/home
lm3-n:/storage                                       nfs4        51G   39M   51G    1% /home
gw-ucl:/CECI/gateway/soft/cecisw/RedHat-7_6-85-4_Omnipath nfs    98M   14M   85M   14% /opt/cecisw/arch
gw-ucl:/CECI/gateway/soft/cecisw/noarch              nfs4        98M   14M   85M   14% /opt/cecisw/noarc
gw-ucl:/CECI/gateway/soft                            nfs         98M   14M   85M   14% /CECI/soft
```

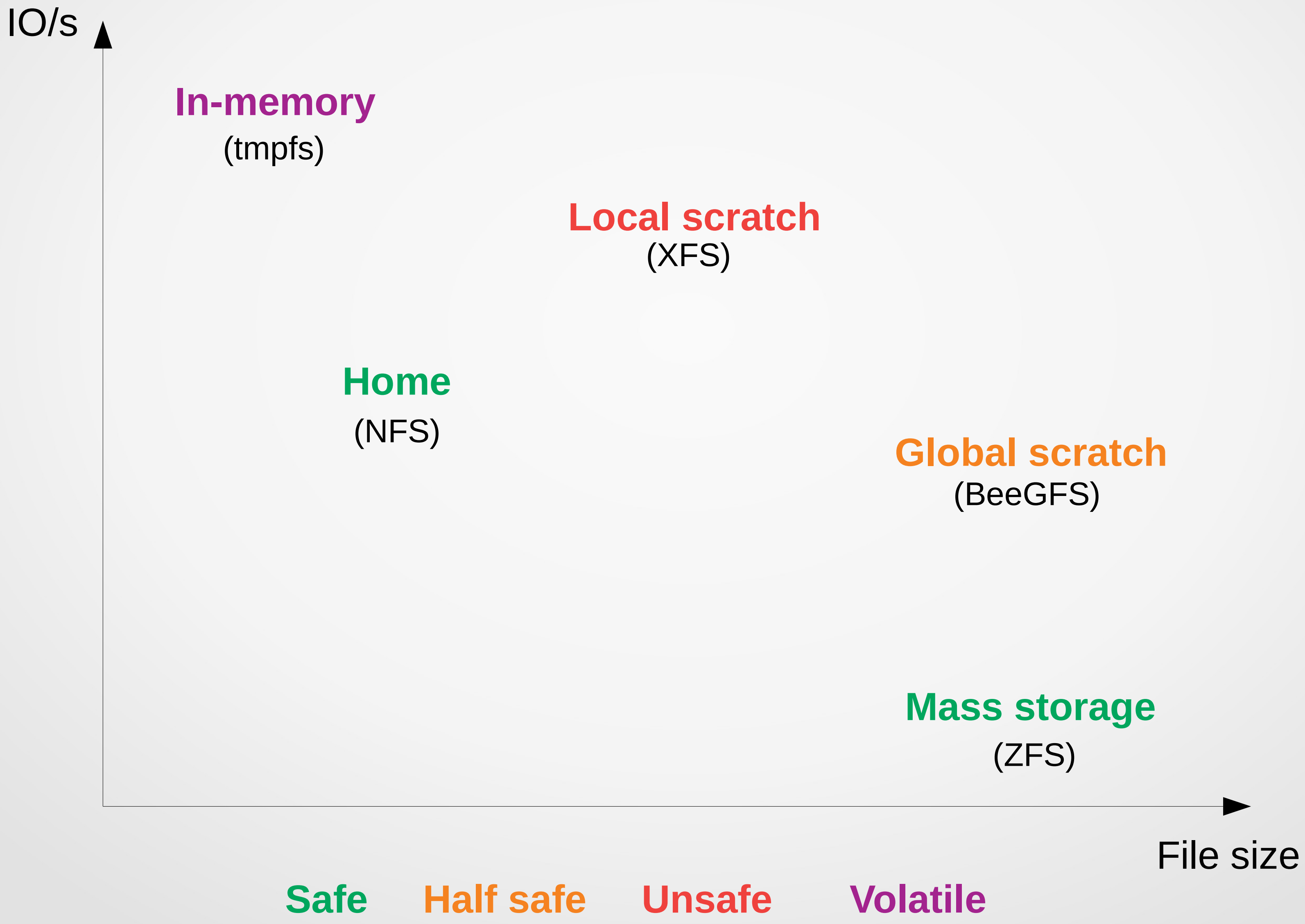
# A word about inodes (simplified)





http://www.porcupine.org/forensics/forensic-discovery/chapter3.html

# What filesystem for what usage



IO/s

**In-memory**
(tmpfs)

**Local scratch**
(XFS)

**Home**
(NFS)

**Global scratch**
(BeeGFS)

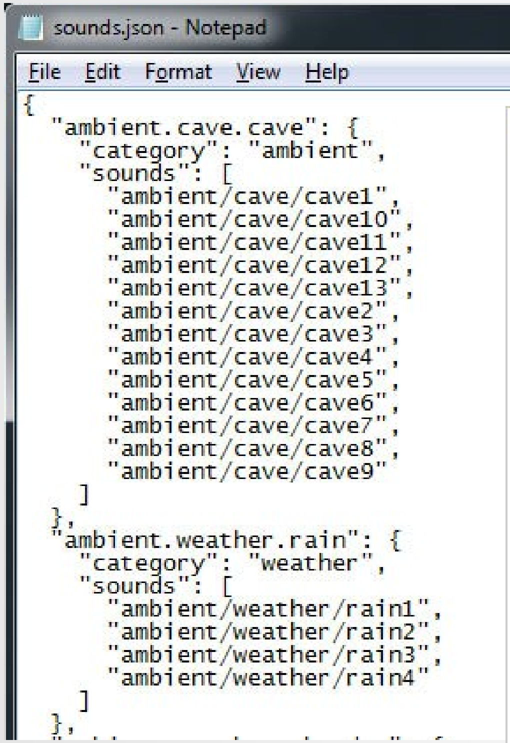
**Mass storage**
(ZFS)

File size

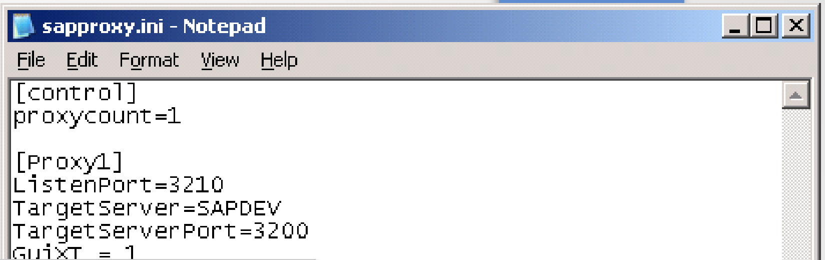
**Safe** **Half safe** **Unsafe** **Volatile**

# File formats

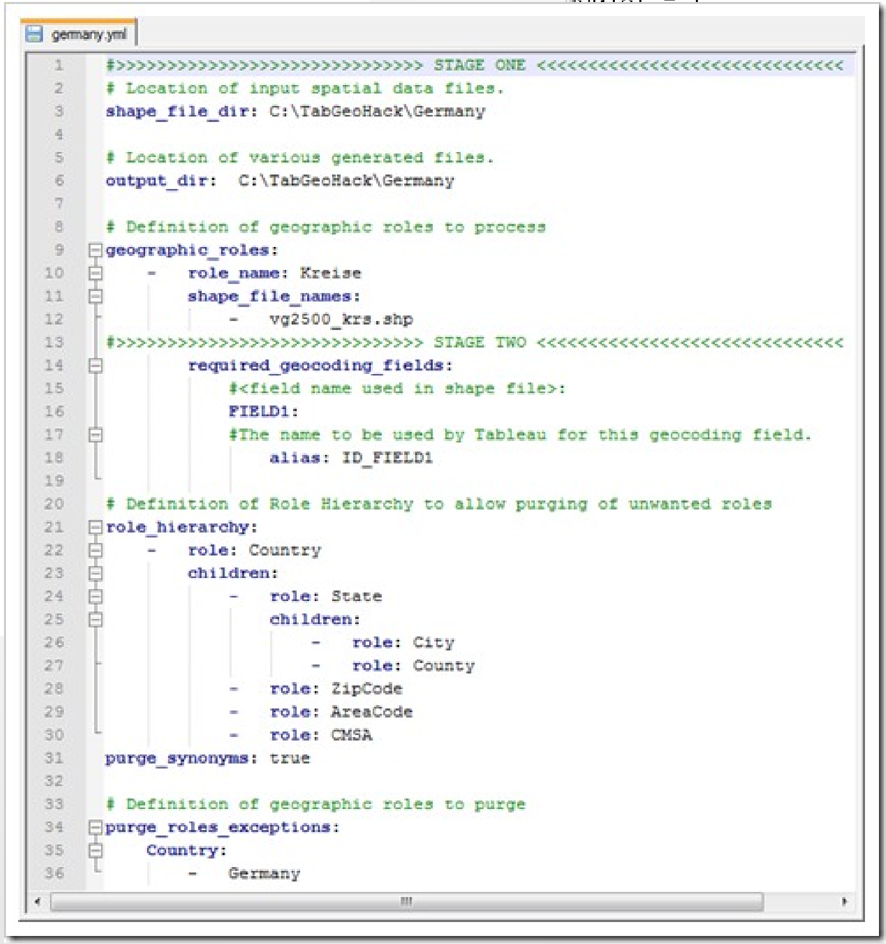
# Text File Formats – JSON, YML, XML, INI



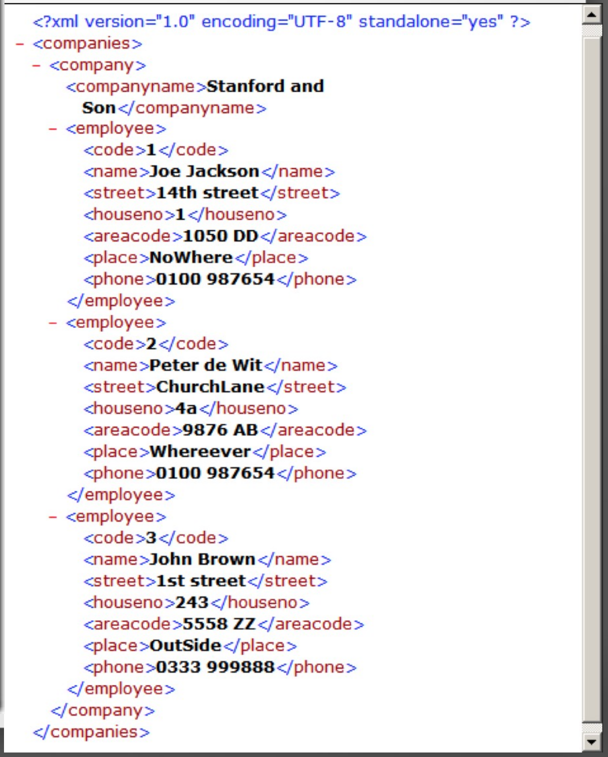
```
{
  "ambient.cave.cave": {
    "category": "ambient",
    "sounds": [
      "ambient/cave/cave1",
      "ambient/cave/cave10",
      "ambient/cave/cave11",
      "ambient/cave/cave12",
      "ambient/cave/cave13",
      "ambient/cave/cave2",
      "ambient/cave/cave3",
      "ambient/cave/cave4",
      "ambient/cave/cave5",
      "ambient/cave/cave6",
      "ambient/cave/cave7",
      "ambient/cave/cave8",
      "ambient/cave/cave9"
    ]
  },
  "ambient.weather.rain": {
    "category": "weather",
    "sounds": [
      "ambient/weather/rain1",
      "ambient/weather/rain2",
      "ambient/weather/rain3",
      "ambient/weather/rain4"
    ]
  },
```



```
[control]
proxycount=1

[Proxy1]
ListenPort=3210
TargetServer=SAPDEV
TargetServerPort=3200
GuiXT = 1
```

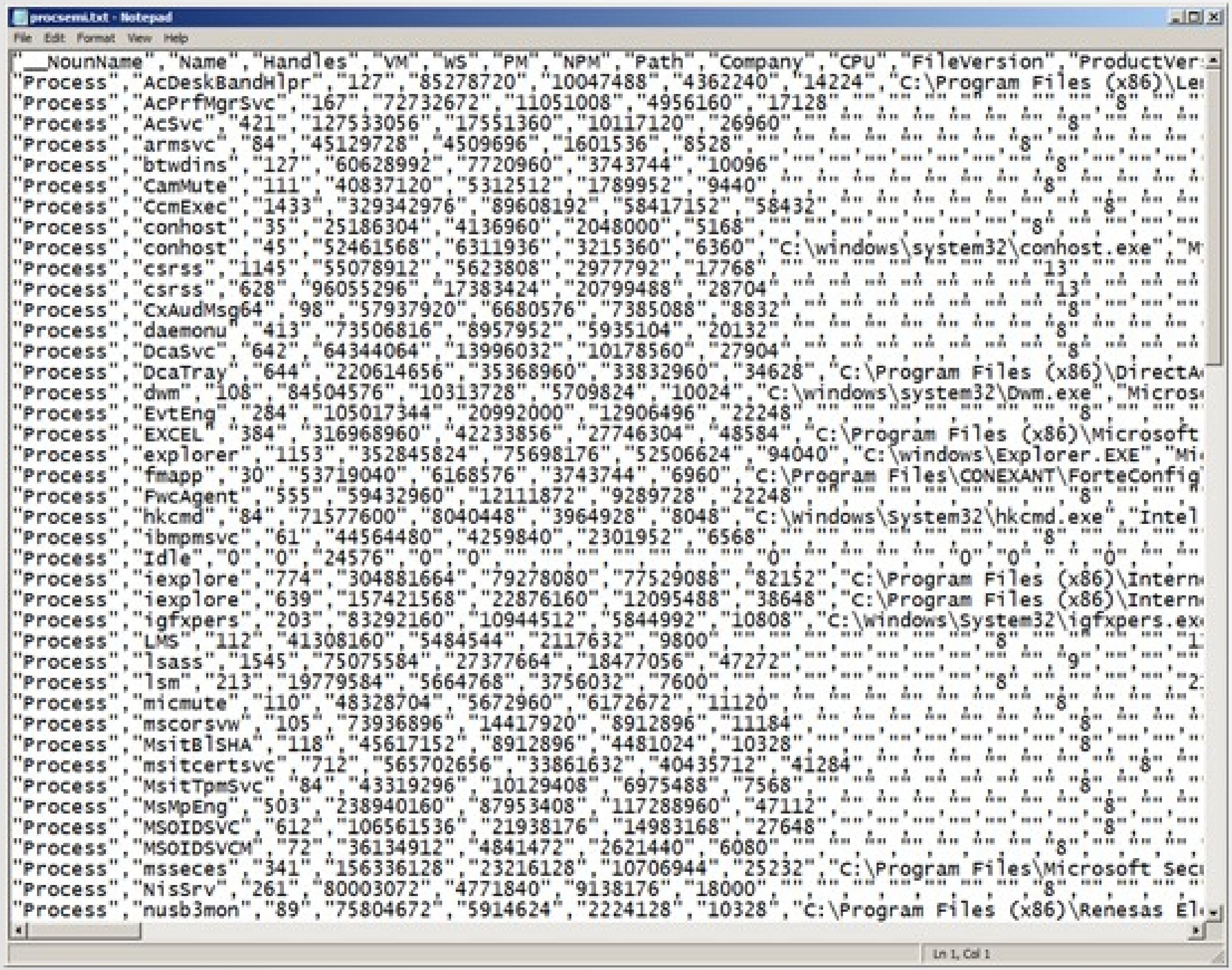


```
#>>>>>>>>>>>>>>>>>>>>>>>>>>>>>> STAGE ONE <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
# Location of input spatial data files.
shape_file_dir: C:\TabGeoHack\Germany

# Location of various generated files.
output_dir:  C:\TabGeoHack\Germany

# Definition of geographic roles to process
geographic_roles:
    -   role_name: Kreise
        shape_file_names:
            -   vg2500_krs.shp
#>>>>>>>>>>>>>>>>>>>>>>>>>>>>>> STAGE TWO <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
        required_geocoding_fields:
            #<field name used in shape file>:
            FIELD1:
            #The name to be used by Tableau for this geocoding field.
                alias: ID_FIELD1

# Definition of Role Hierarchy to allow purging of unwanted roles
role_hierarchy:
    -   role: Country
        children:
            -   role: State
                children:
                    -   role: City
                    -   role: County
            -   role: ZipCode
            -   role: AreaCode
            -   role: CMSA
purge_synonyms: true

# Definition of geographic roles to purge
purge_roles_exceptions:
    Country:
        -   Germany
```



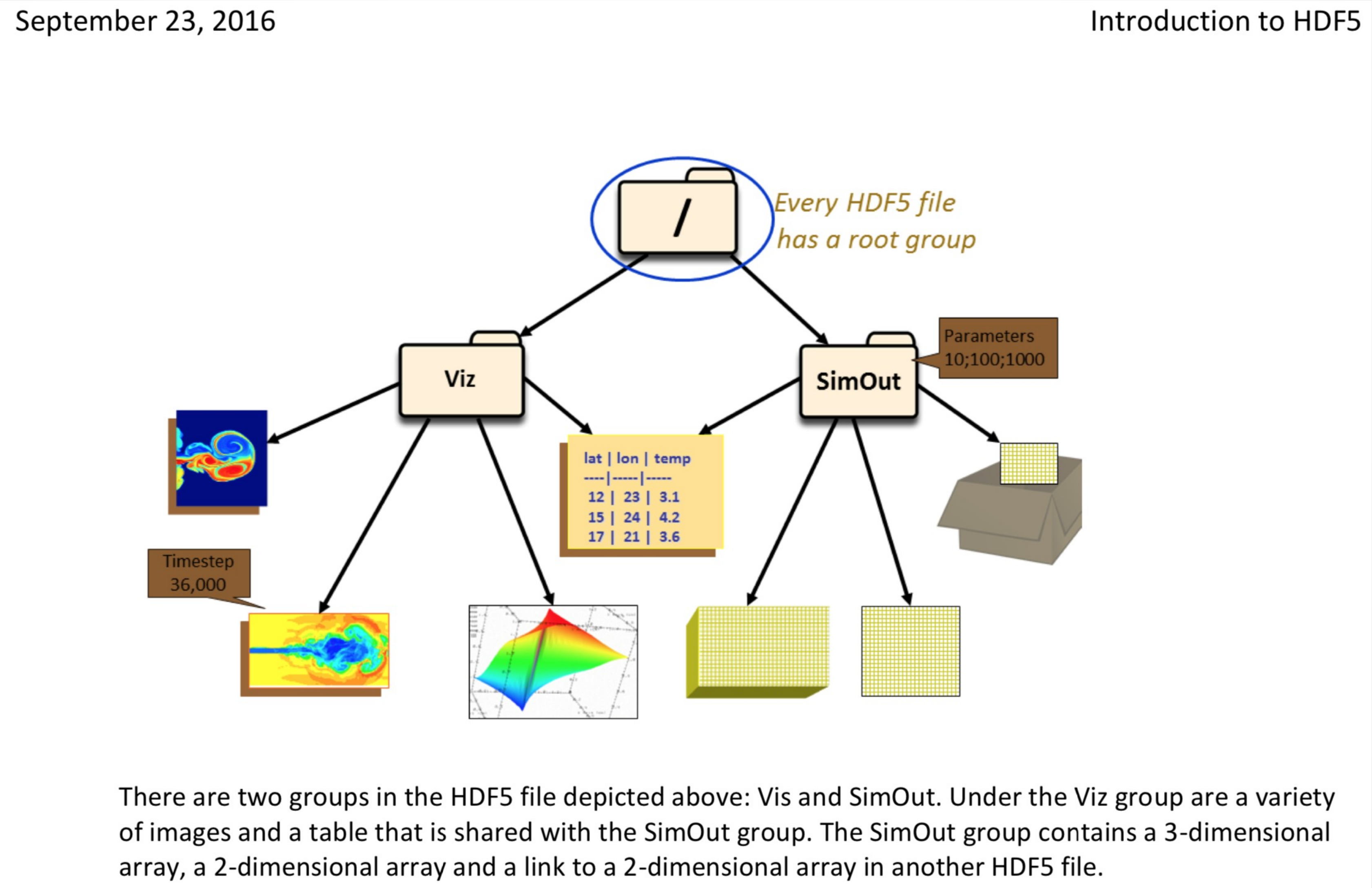
```
  <?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
- <companies>
  - <company>
      <companyname>Stanford and
        Son</companyname>
    - <employee>
        <code>1</code>
        <name>Joe Jackson</name>
        <street>14th street</street>
        <houseno>1</houseno>
        <areacode>1050 DD</areacode>
        <place>NoWhere</place>
        <phone>0100 987654</phone>
      </employee>
    - <employee>
        <code>2</code>
        <name>Peter de Wit</name>
        <street>ChurchLane</street>
        <houseno>4a</houseno>
        <areacode>9876 AB</areacode>
        <place>Whereever</place>
        <phone>0100 987654</phone>
      </employee>
    - <employee>
        <code>3</code>
        <name>John Brown</name>
        <street>1st street</street>
        <houseno>243</houseno>
        <areacode>5558 ZZ</areacode>
        <place>OutSide</place>
        <phone>0333 999888</phone>
      </employee>
    </company>
  </companies>
```

# Text File Formats – CSV



```
"__NounName","Name","Handles","VM","WS","PM","NPM","Path","Company","CPU","FileVersion","ProductVer
"Process","AcDeskBandHlpr","127","85278720","10047488","4362240","14224","C:\Program Files (x86)\Le
"Process","AcPrfMgrSvc","167","72732672","11051008","4956160","17128","","","","","","","","8","",""
"Process","AcSvc","421","127533056","17551360","10117120","26960","","","","","","","","8","","",""
"Process","armsvc","84","45129728","4509696","1601536","8528","","","","","","","","8","","","",""
"Process","btwdins","127","60628992","7720960","3743744","10096","","","","","","","","8","","",""
"Process","CamMute","111","40837120","5312512","1789952","9440","","","","","","","","8","","","",""
"Process","CcmExec","1433","329342976","89608192","58417152","58432","","","","","","","","8","",""
"Process","conhost","35","25186304","4136960","2048000","5168","","","","","","","","8","","","",""
"Process","conhost","45","52461568","6311936","3215360","6360","C:\windows\system32\conhost.exe","M
"Process","csrss","1145","55078912","5623808","2977792","17768","","","","","","","","13","","",""
"Process","csrss","628","96055296","17383424","20799488","28704","","","","","","","","13","","",""
"Process","CxAudMsg64","98","57937920","6680576","7385088","8832","","","","","","","","8","","",""
"Process","daemonu","413","73506816","8957952","5935104","20132","","","","","","","","8","","",""
"Process","DcaSvc","642","64344064","13996032","10178560","27904","","","","","","","","8","","",""
"Process","DcaTray","644","220614656","35368960","33832960","34628","C:\Program Files (x86)\DirectA
"Process","dwm","108","84504576","10313728","5709824","10024","C:\windows\system32\Dwm.exe","Micros
"Process","EvtEng","284","105017344","20992000","12906496","22248","","","","","","","","8","","",""
"Process","EXCEL","384","316968960","42233856","27746304","48584","C:\Program Files (x86)\Microsoft
"Process","explorer","1153","352845824","75698176","52506624","94040","C:\windows\Explorer.EXE","Mi
"Process","fmapp","30","53719040","6168576","3743744","6960","C:\Program Files\CONEXANT\ForteConfig
"Process","FwcAgent","555","59432960","12111872","9289728","22248","","","","","","","","8","","",""
"Process","hkcmd","84","71577600","8040448","3964928","8048","C:\Windows\System32\hkcmd.exe","Intel
"Process","ibmpmsvc","61","44564480","4259840","2301952","6568","","","","","","","","8","","","",""
"Process","Idle","0","0","24576","0","0","","","","","","","","0","","","","","0","0","","","0","",""
"Process","iexplore","774","304881664","79278080","77529088","82152","C:\Program Files (x86)\Intern
"Process","iexplore","639","157421568","22876160","12095488","38648","C:\Program Files (x86)\Intern
"Process","igfxpers","203","83292160","10944512","5844992","10808","C:\Windows\System32\igfxpers.ex
"Process","LMS","112","41308160","5484544","2117632","9800","","","","","","","","8","","","","","1
"Process","lsass","1545","75075584","27377664","18477056","47272","","","","","","","","9","","",""
"Process","lsm","213","19779584","5664768","3756032","7600","","","","","","","","8","","","","","2
"Process","micmute","110","48328704","5672960","6172672","11120","","","","","","","","8","","",""
"Process","mscorsvw","105","73936896","14417920","8912896","11184","","","","","","","","8","","",""
"Process","MsitBlSHA","118","45617152","8912896","4481024","10328","","","","","","","","8","","",""
"Process","msitcertsvc","712","565702656","33861632","40435712","41284","","","","","","","","8",""
"Process","MsitTpmSvc","84","43319296","10129408","6975488","7568","","","","","","","","8","","",""
"Process","MsMpEng","503","238940160","87953408","117288960","47112","","","","","","","","8","",""
"Process","MSOIDSVC","612","106561536","21938176","14983168","27648","","","","","","","","8","",""
"Process","MSOIDSVCM","72","36134912","4841472","2621440","6080","","","","","","","","8","","",""
"Process","msseces","341","156336128","23216128","10706944","25232","C:\Program Files\Microsoft Sec
"Process","NisSrv","261","80003072","4771840","9138176","18000","","","","","","","","8","","","",""
"Process","nusb3mon","89","75804672","5914624","2224128","10328","C:\Program Files (x86)\Renesas El
```

# Binary File Formats – CDF, HDF



September 23, 2016

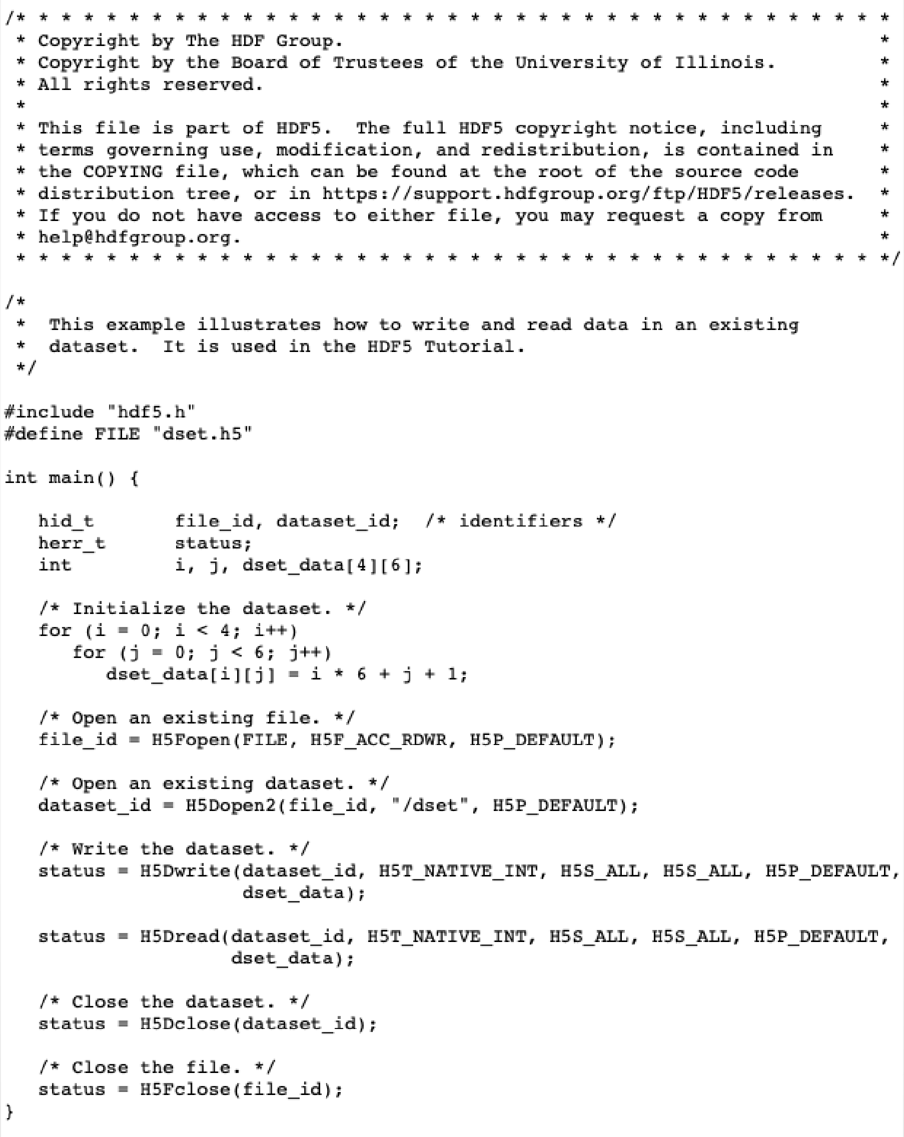
Introduction to HDF5

| lat | lon | temp |
|-----|-----|------|
| 12 | 23 | 3.1 |
| 15 | 24 | 4.2 |
| 17 | 21 | 3.6 |

There are two groups in the HDF5 file depicted above: Vis and SimOut. Under the Viz group are a variety of images and a table that is shared with the SimOut group. The SimOut group contains a 3-dimensional array, a 2-dimensional array and a link to a 2-dimensional array in another HDF5 file.

https://support.hdfgroup.org/HDF5/Tutor/HDF5Intro.pdf

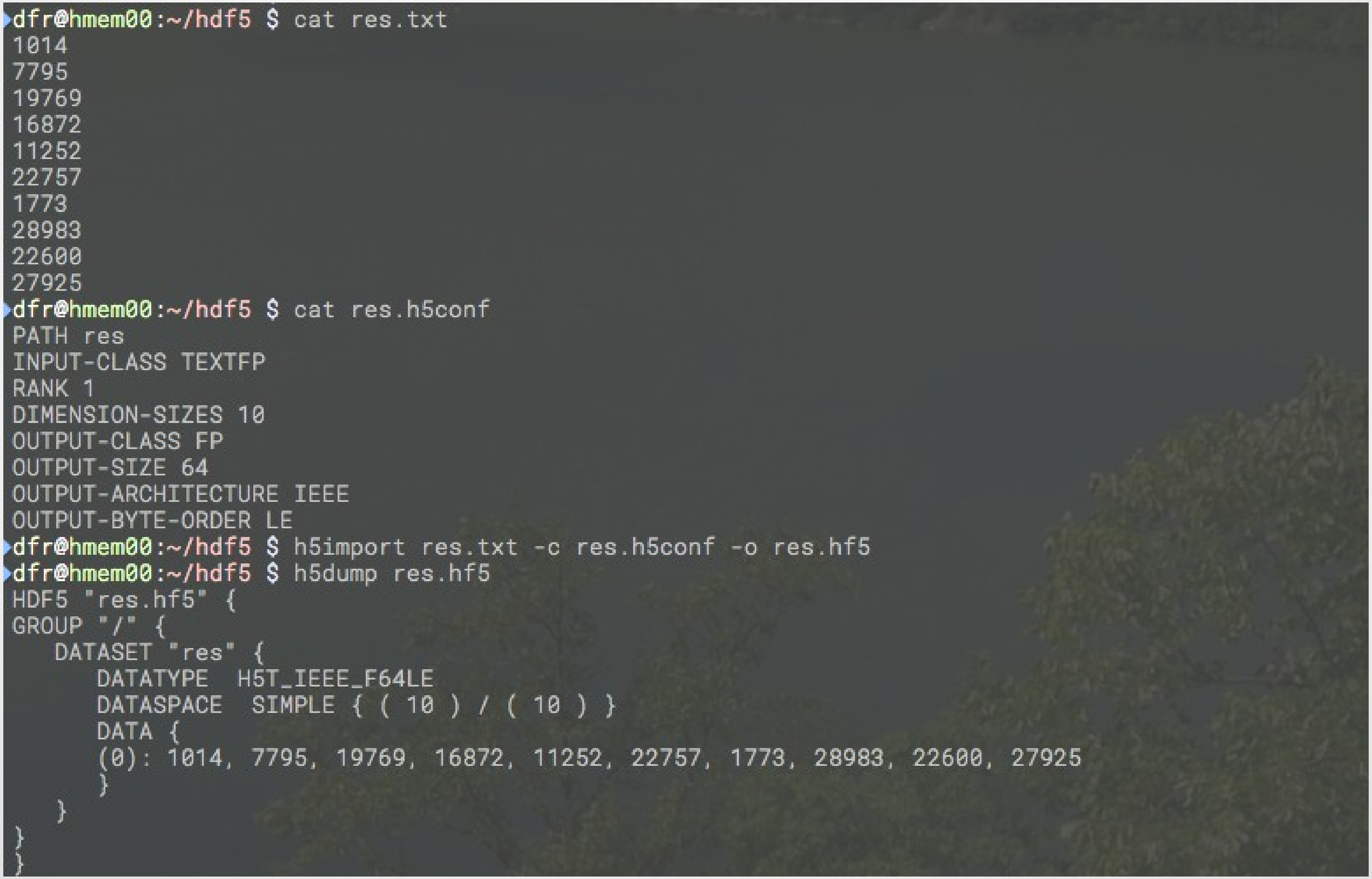
# Binary File Formats – CDF, HDF



```
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 * Copyright by The HDF Group.                                               *
 * Copyright by the Board of Trustees of the University of Illinois.         *
 * All rights reserved.                                                      *
 *                                                                           *
 * This file is part of HDF5.  The full HDF5 copyright notice, including     *
 * terms governing use, modification, and redistribution, is contained in    *
 * the COPYING file, which can be found at the root of the source code       *
 * distribution tree, or in https://support.hdfgroup.org/ftp/HDF5/releases.  *
 * If you do not have access to either file, you may request a copy from     *
 * help@hdfgroup.org.                                                        *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */

/*
 *  This example illustrates how to write and read data in an existing
 *  dataset.  It is used in the HDF5 Tutorial.
 */

#include "hdf5.h"
#define FILE "dset.h5"

int main() {

   hid_t       file_id, dataset_id;  /* identifiers */
   herr_t      status;
   int         i, j, dset_data[4][6];

   /* Initialize the dataset. */
   for (i = 0; i < 4; i++)
      for (j = 0; j < 6; j++)
         dset_data[i][j] = i * 6 + j + 1;

   /* Open an existing file. */
   file_id = H5Fopen(FILE, H5F_ACC_RDWR, H5P_DEFAULT);

   /* Open an existing dataset. */
   dataset_id = H5Dopen2(file_id, "/dset", H5P_DEFAULT);

   /* Write the dataset. */
   status = H5Dwrite(dataset_id, H5T_NATIVE_INT, H5S_ALL, H5S_ALL, H5P_DEFAULT,
                     dset_data);

   status = H5Dread(dataset_id, H5T_NATIVE_INT, H5S_ALL, H5S_ALL, H5P_DEFAULT,
                    dset_data);

   /* Close the dataset. */
   status = H5Dclose(dataset_id);

   /* Close the file. */
   status = H5Fclose(file_id);
}
```

https://support.hdfgroup.org/ftp/HDF5/current/src/unpacked/examples/h5_rdwt.c

# Binary File Formats – CDF, HDF



```
dfr@hmem00:~/hdf5 $ cat res.txt
1014
7795
19769
16872
11252
22757
1773
28983
22600
27925
dfr@hmem00:~/hdf5 $ cat res.h5conf
PATH res
INPUT-CLASS TEXTFP
RANK 1
DIMENSION-SIZES 10
OUTPUT-CLASS FP
OUTPUT-SIZE 64
OUTPUT-ARCHITECTURE IEEE
OUTPUT-BYTE-ORDER LE
dfr@hmem00:~/hdf5 $ h5import res.txt -c res.h5conf -o res.hf5
dfr@hmem00:~/hdf5 $ h5dump res.hf5
HDF5 "res.hf5" {
GROUP "/" {
   DATASET "res" {
      DATATYPE  H5T_IEEE_F64LE
      DATASPACE  SIMPLE { ( 10 ) / ( 10 ) }
      DATA {
      (0): 1014, 7795, 19769, 16872, 11252, 22757, 1773, 28983, 22600, 27925
      }
   }
}
}
```

# What file format for what usage

- Meta data
  - Configuration file: INI, YAML
  - Result with context information: JSON
- Data
  - Small data (kBs): CSV
  - Medium data (MBs): compressed CSV
  - Large data (GBs): netCDF, HDF5, DXMF
  - Huge data (TBs): Database, Object store

(“loss of innocence”)

Use dedicated libraries to write and read them

# 1.2 Object storage

# Object storage

- Object: data (e.g. file) + **custom** meta data

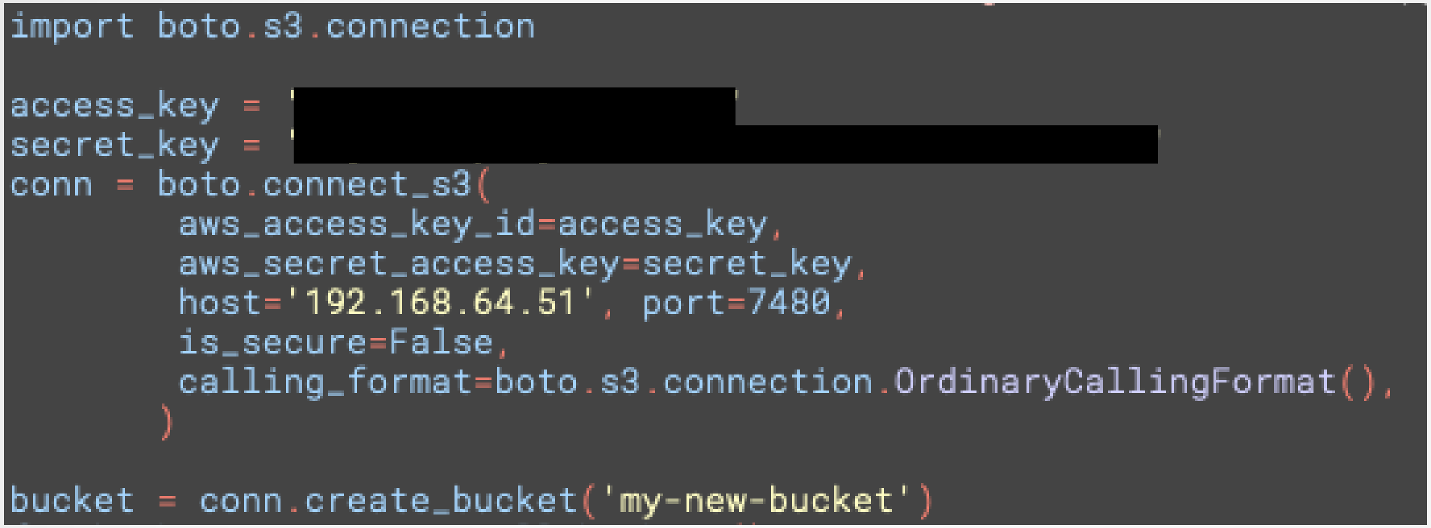
SWIFT
*an OpenStack Community Project*

S3

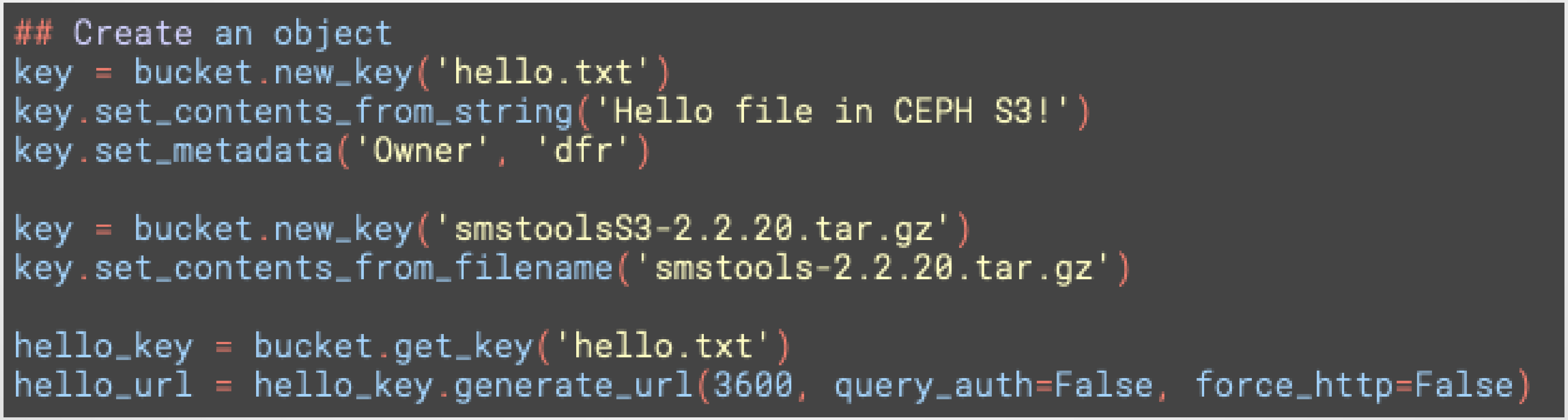
ceph

- Often built on erasure coding (“software RAID”)
- Scale out easily
- Useful for web applications but coming to scientific world
- Access with REST API (through HTTP)

# S3 Python example



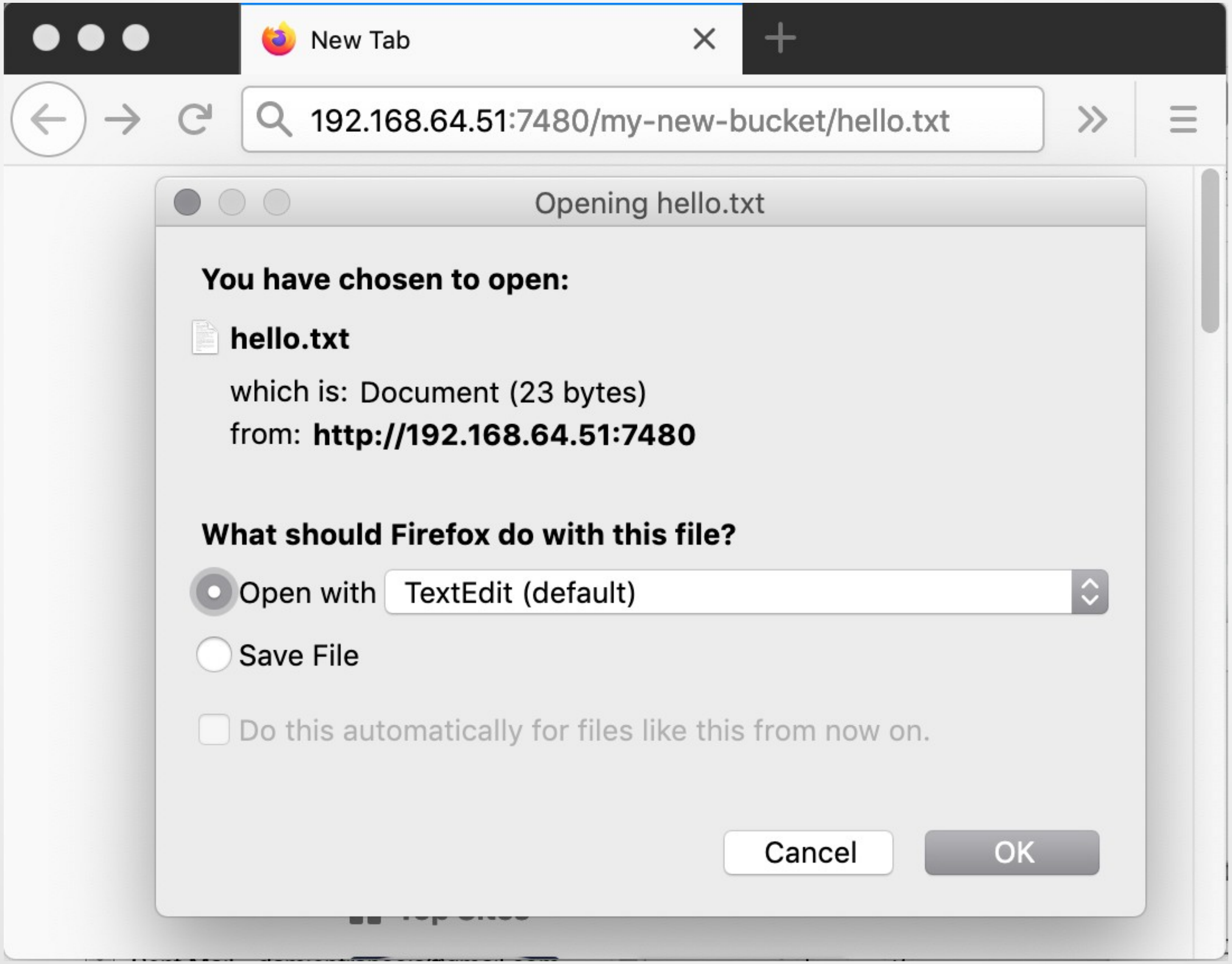
```
import boto.s3.connection

access_key = '[redacted]
secret_key = '[redacted]
conn = boto.connect_s3(
        aws_access_key_id=access_key,
        aws_secret_access_key=secret_key,
        host='192.168.64.51', port=7480,
        is_secure=False,
        calling_format=boto.s3.connection.OrdinaryCallingFormat(),
       )

bucket = conn.create_bucket('my-new-bucket')
```



```
## Create an object
key = bucket.new_key('hello.txt')
key.set_contents_from_string('Hello file in CEPH S3!')
key.set_metadata('Owner', 'dfr')

key = bucket.new_key('smstoolsS3-2.2.20.tar.gz')
key.set_contents_from_filename('smstools-2.2.20.tar.gz')

hello_key = bucket.get_key('hello.txt')
hello_url = hello_key.generate_url(3600, query_auth=False, force_http=False)
```

https://pypi.org/project/boto/

# S3 Python example



192.168.64.51:7480/my-new-bucket/hello.txt

Opening hello.txt

**You have chosen to open:**

**hello.txt**

which is: Document (23 bytes)

from: **http://192.168.64.51:7480**

**What should Firefox do with this file?**

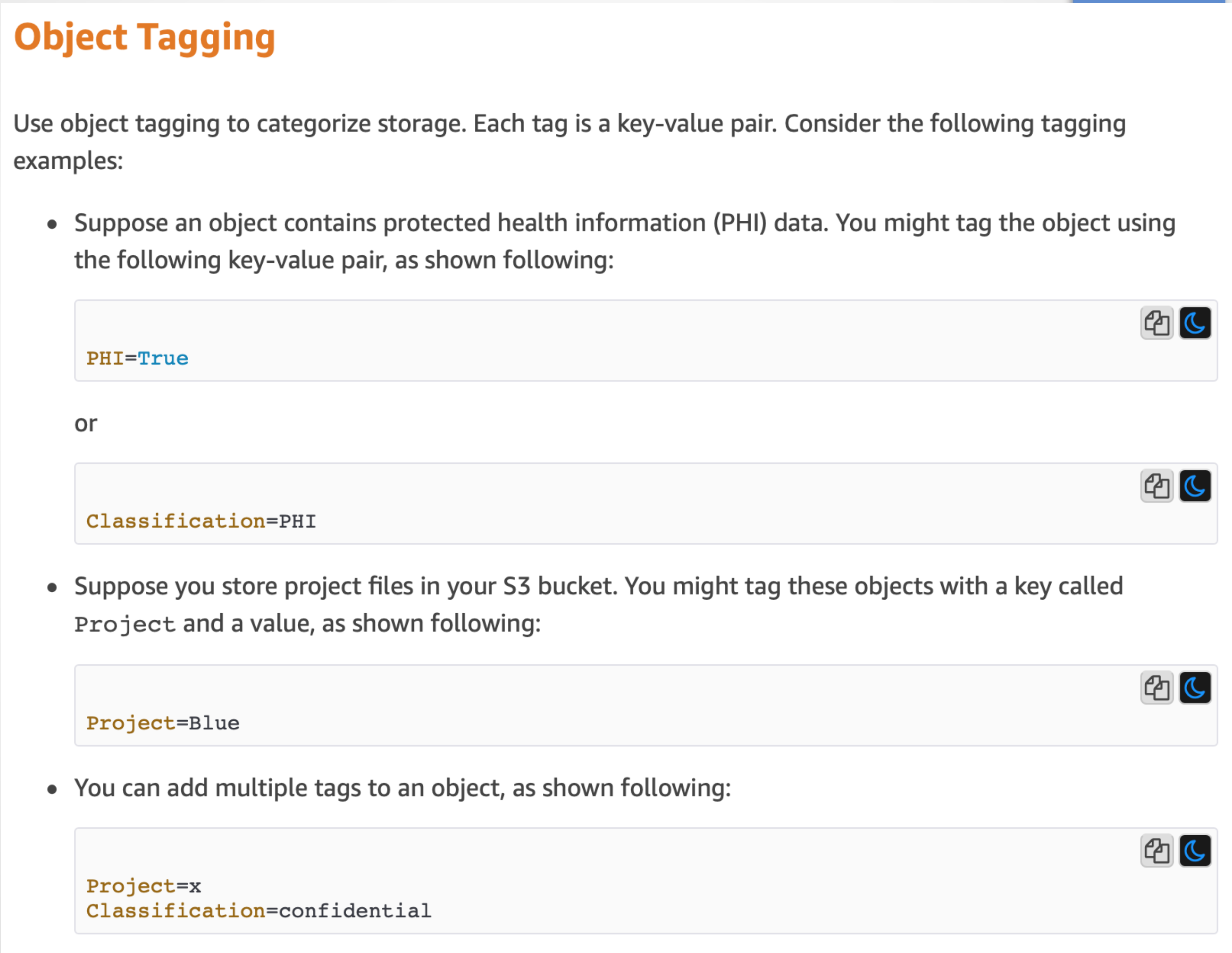
● Open with TextEdit (default)

○ Save File

□ Do this automatically for files like this from now on.

https://pypi.org/project/boto/

# S3 Object tagging (meta data)



## Object Tagging

Use object tagging to categorize storage. Each tag is a key-value pair. Consider the following tagging examples:

- Suppose an object contains protected health information (PHI) data. You might tag the object using the following key-value pair, as shown following:

```
PHI=True
```

or

```
Classification=PHI
```

- Suppose you store project files in your S3 bucket. You might tag these objects with a key called `Project` and a value, as shown following:

```
Project=Blue
```

- You can add multiple tags to an object, as shown following:

```
Project=x
Classification=confidential
```

https://docs.aws.amazon.com/AmazonS3/latest/dev/object-tagging.html

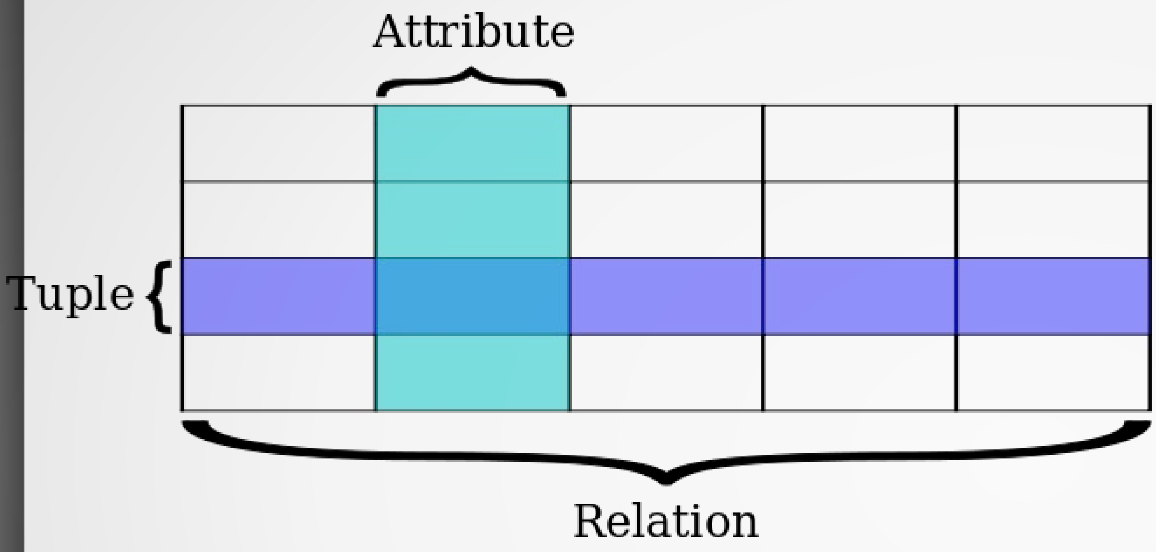
# 1.3 Databases

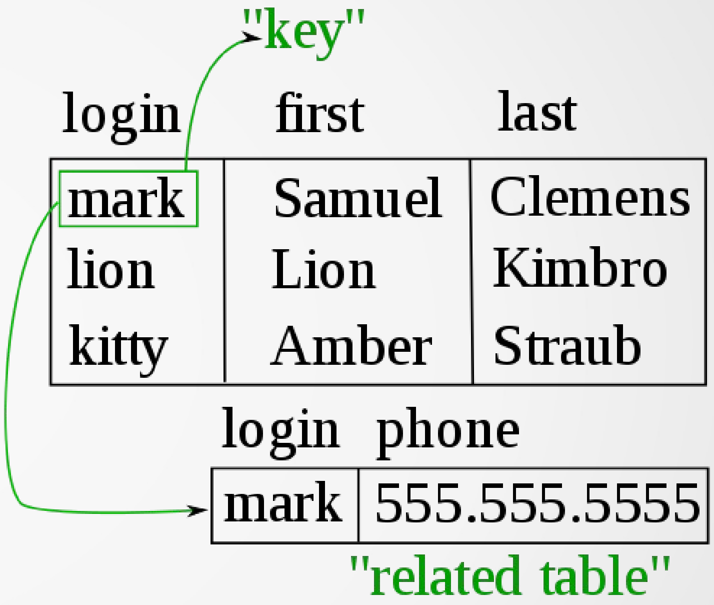
# RDBMS

- Mostly needed for categorical data and alphanumerical data (not suited for matrices, but good for end-results)
- Indexes make finding a data element is very fast (and computing sums, maxima, etc.)
- Encodes relations between data (constraints, etc)
- Atomicity, Consistency, Isolation, and Durability

Pictures from http://www.ibm.com/developerworks/library/x-matters8/

# Tables and SQL





| login | first | last |
|---|---|---|
| mark | Samuel | Clemens |
| lion | Lion | Kimbro |
| kitty | Amber | Straub |

| login | phone |
|---|---|
| mark | 555.555.5555 |

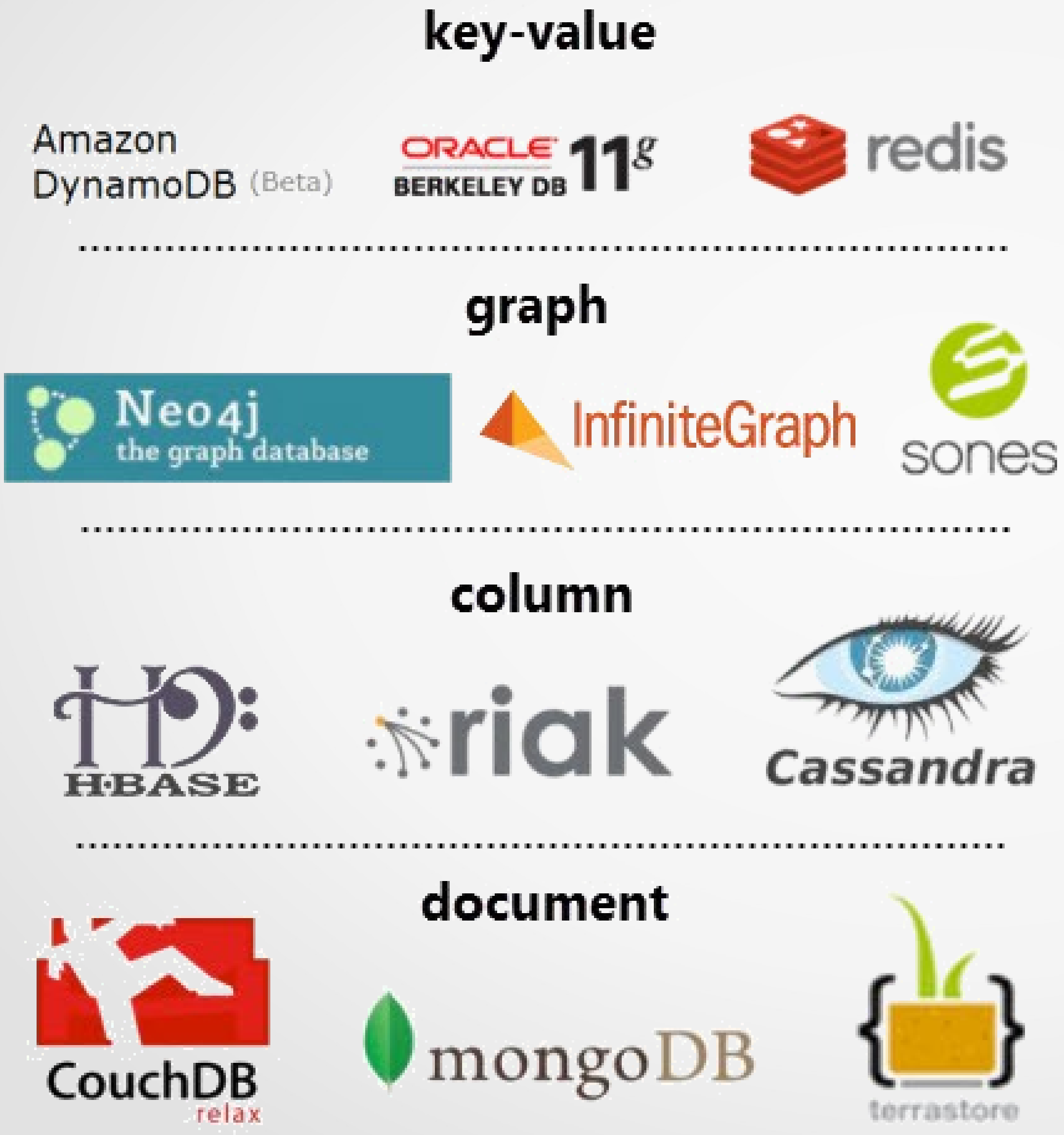
```
create table Users (login varchar(255), first varchar(255), last varchar(255));

insert into Users values (“mark”, ‘Samuel”, “Clemens”);

select first,last from Users where login=’lion’;
select login, phone from Users join PhoneNb on Users.login=Phone.login;
```

# File-based RDBMS

SQLite DuckDB

- The features of a relational database without the need for a complete setup
- Simply based on files
- Command line interface + API (Python, etc.)
- Still much more efficient than writing a million small files

# NoSQL



- Mostly needed for unstructured, semi-structured, and polymorphic data
- Scaling out very easy
- Basic Availability, Soft-state, Eventual consistency

Pictures from http://www.tomsitpro.com/articles/rdbms-sql-cassandra-dba-developer,2-547-2.html

# File-based NoSQL

TinyDB

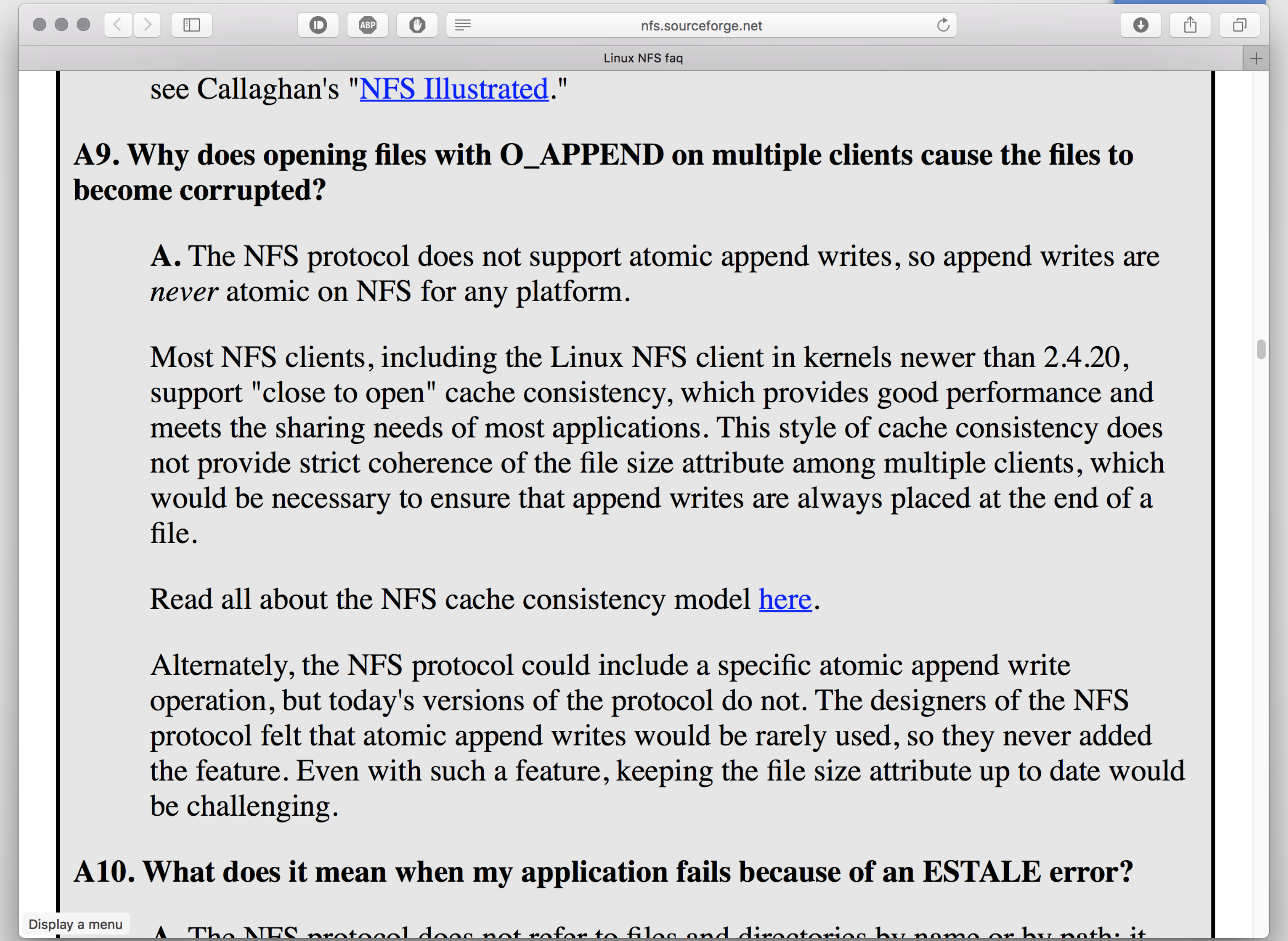
- The features of a relational database without the need for a complete setup
- Simply based on files
- Command line interface + API (Python, etc.)
- Still much more efficient than writing a million small files

https://github.com/msiemens/tinydb

# When to use a database?

- when you have a large number of small files
- when you perform a lot of direct writes in a large file
- when you want to keep structure/relations between data
- when software crashes have a non-negligible probability
- when files are updated by several processes

# Example: Danger of NFS



see Callaghan's "NFS Illustrated."

**A9. Why does opening files with O_APPEND on multiple clients cause the files to become corrupted?**

**A.** The NFS protocol does not support atomic append writes, so append writes are *never* atomic on NFS for any platform.

Most NFS clients, including the Linux NFS client in kernels newer than 2.4.20, support "close to open" cache consistency, which provides good performance and meets the sharing needs of most applications. This style of cache consistency does not provide strict coherence of the file size attribute among multiple clients, which would be necessary to ensure that append writes are always placed at the end of a file.

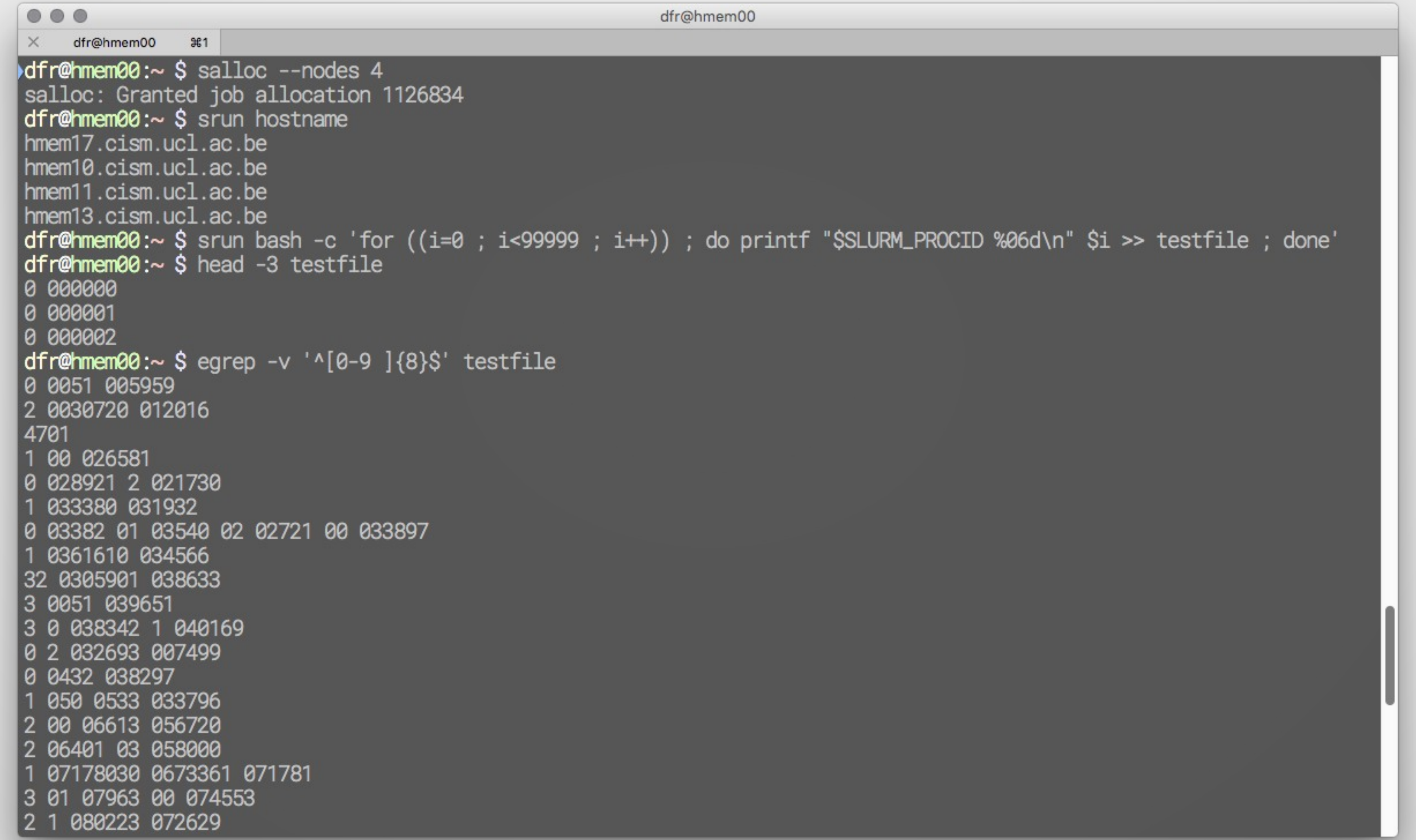
Read all about the NFS cache consistency model here.

Alternately, the NFS protocol could include a specific atomic append write operation, but today's versions of the protocol do not. The designers of the NFS protocol felt that atomic append writes would be rarely used, so they never added the feature. Even with such a feature, keeping the file size attribute up to date would be challenging.

**A10. What does it mean when my application fails because of an ESTALE error?**

**A.** The NFS protocol does not refer to files and directories by name or by path; it

# Example: Danger of NFS



```
dfr@hmem00:~ $ salloc --nodes 4
salloc: Granted job allocation 1126834
dfr@hmem00:~ $ srun hostname
hmem17.cism.ucl.ac.be
hmem10.cism.ucl.ac.be
hmem11.cism.ucl.ac.be
hmem13.cism.ucl.ac.be
dfr@hmem00:~ $ srun bash -c 'for ((i=0 ; i<99999 ; i++)) ; do printf "$SLURM_PROCID %06d\n" $i >> testfile ; done'
dfr@hmem00:~ $ head -3 testfile
0 000000
0 000001
0 000002
dfr@hmem00:~ $ egrep -v '^[0-9 ]{8}$' testfile
0 0051 005959
2 0030720 012016
4701
1 00 026581
0 028921 2 021730
1 033380 031932
0 03382 01 03540 02 02721 00 033897
1 0361610 034566
32 0305901 038633
3 0051 039651
3 0 038342 1 040169
0 2 032693 007499
0 0432 038297
1 050 0533 033796
2 00 06613 056720
2 06401 03 058000
1 07178030 0673361 071781
3 01 07963 00 074553
2 1 080223 072629
```

# When NOT to use a database?

- only sequential access
- simple matrices/vectors, etc.
- direct access on fixed-size records and no structure

# 2.

## Data transfer

# Network technologies

Laptop

100Mbps

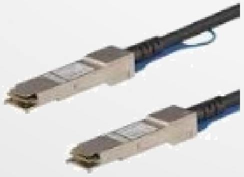
Building switch

10Gbps

Datacenter switch

1Gbps

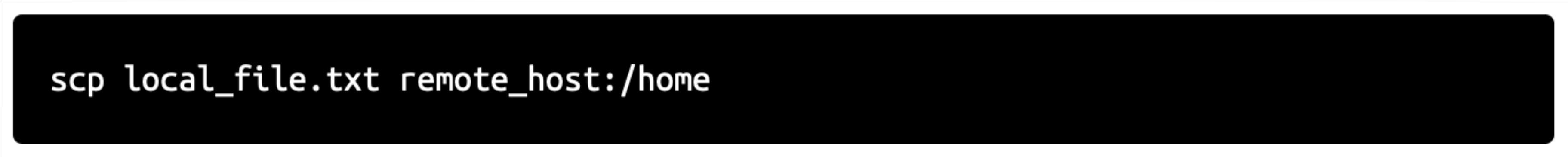
Cluster frontend



100Gbps

Compute node

# SCP

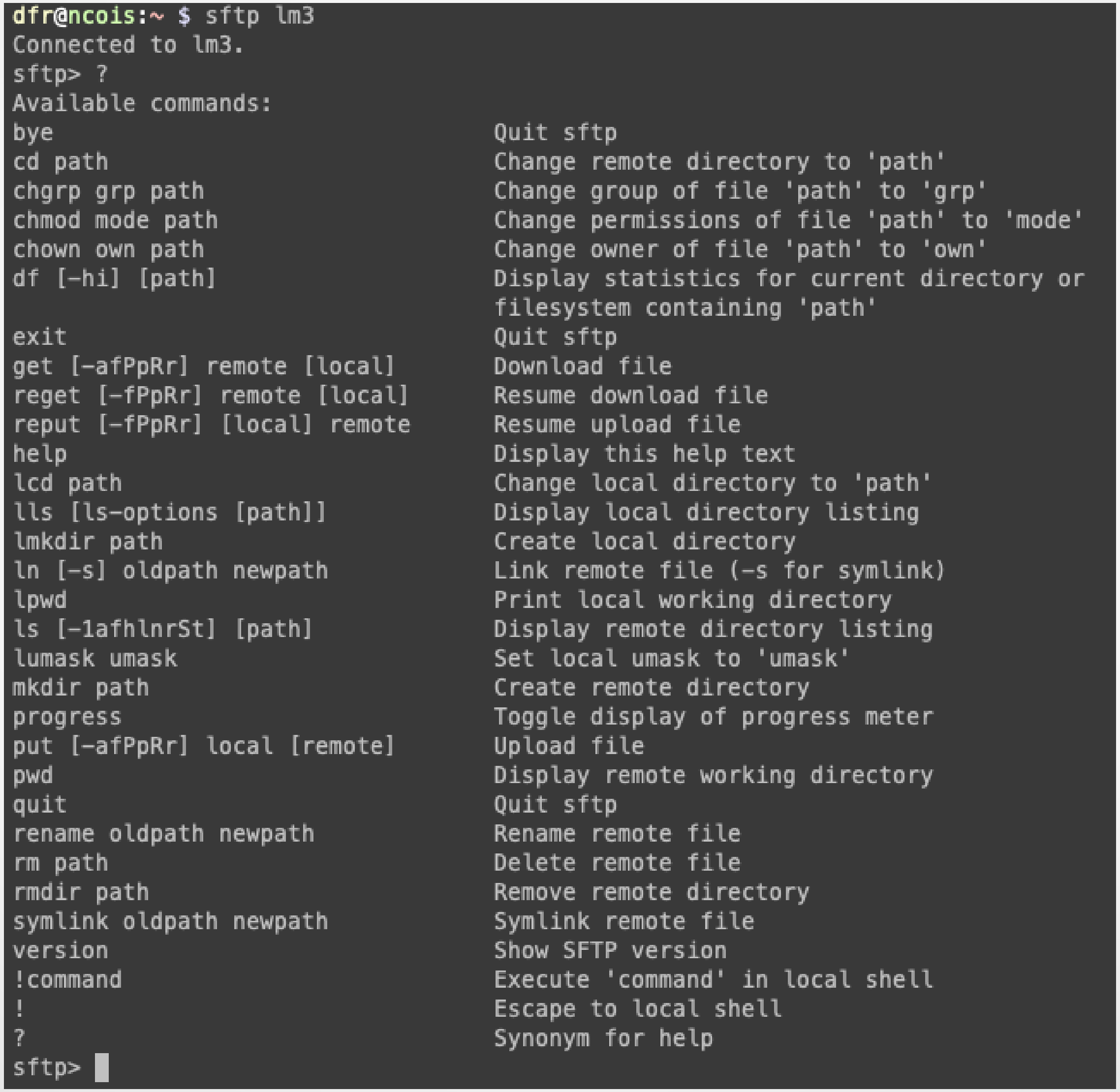


```
scp local_file.txt remote_host:/home
```

- Most direct way to copy data from/to UNIX/Linux machines
- Inefficient (sequential and synchronous)
- “outdated, inflexible and not readily fixed” (OpenSSH 8.0 release notes)

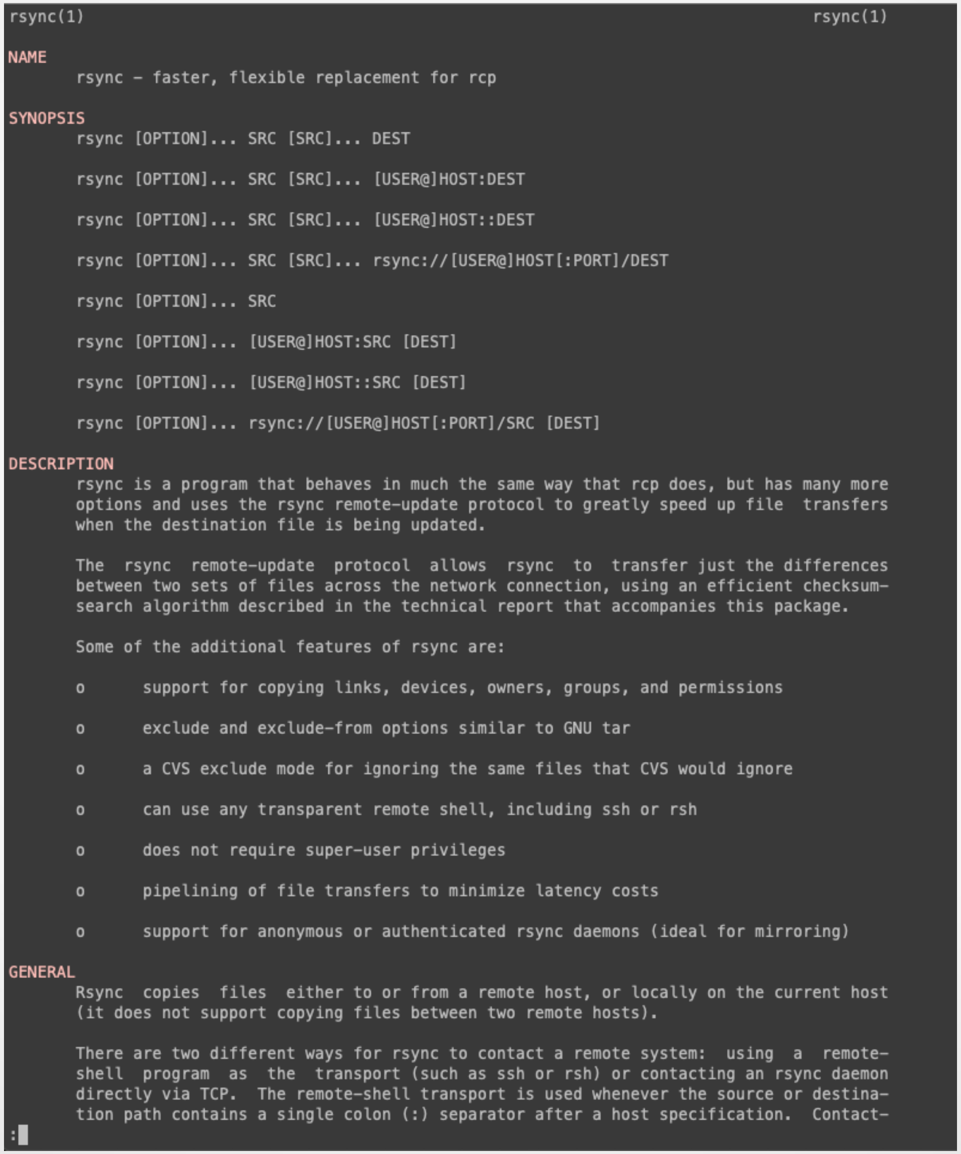
https://gravitational.com/blog/scp-familiar-simple-insecure-slow/

# SFTP



```
dfr@ncois:~ $ sftp lm3
Connected to lm3.
sftp> ?
Available commands:
bye                                Quit sftp
cd path                            Change remote directory to 'path'
chgrp grp path                     Change group of file 'path' to 'grp'
chmod mode path                    Change permissions of file 'path' to 'mode'
chown own path                     Change owner of file 'path' to 'own'
df [-hi] [path]                    Display statistics for current directory or
                                   filesystem containing 'path'
exit                               Quit sftp
get [-afPpRr] remote [local]       Download file
reget [-fPpRr] remote [local]      Resume download file
reput [-fPpRr] [local] remote      Resume upload file
help                               Display this help text
lcd path                           Change local directory to 'path'
lls [ls-options [path]]            Display local directory listing
lmkdir path                        Create local directory
ln [-s] oldpath newpath            Link remote file (-s for symlink)
lpwd                               Print local working directory
ls [-1afhlnrSt] [path]             Display remote directory listing
lumask umask                       Set local umask to 'umask'
mkdir path                         Create remote directory
progress                           Toggle display of progress meter
put [-afPpRr] local [remote]       Upload file
pwd                                Display remote working directory
quit                               Quit sftp
rename oldpath newpath             Rename remote file
rm path                            Delete remote file
rmdir path                         Remove remote directory
symlink oldpath newpath            Symlink remote file
version                            Show SFTP version
!command                           Execute 'command' in local shell
!                                  Escape to local shell
?                                  Synonym for help
sftp> 
```

# RSYNC



```
rsync(1)                                                                    rsync(1)

NAME
       rsync - faster, flexible replacement for rcp

SYNOPSIS
       rsync [OPTION]... SRC [SRC]... DEST

       rsync [OPTION]... SRC [SRC]... [USER@]HOST:DEST

       rsync [OPTION]... SRC [SRC]... [USER@]HOST::DEST

       rsync [OPTION]... SRC [SRC]... rsync://[USER@]HOST[:PORT]/DEST

       rsync [OPTION]... SRC

       rsync [OPTION]... [USER@]HOST:SRC [DEST]

       rsync [OPTION]... [USER@]HOST::SRC [DEST]

       rsync [OPTION]... rsync://[USER@]HOST[:PORT]/SRC [DEST]

DESCRIPTION
       rsync is a program that behaves in much the same way that rcp does, but has many more
       options and uses the rsync remote-update protocol to greatly speed up file  transfers
       when the destination file is being updated.

       The  rsync  remote-update  protocol  allows  rsync  to  transfer just the differences
       between two sets of files across the network connection, using an efficient checksum-
       search algorithm described in the technical report that accompanies this package.

       Some of the additional features of rsync are:

       o      support for copying links, devices, owners, groups, and permissions

       o      exclude and exclude-from options similar to GNU tar

       o      a CVS exclude mode for ignoring the same files that CVS would ignore

       o      can use any transparent remote shell, including ssh or rsh

       o      does not require super-user privileges

       o      pipelining of file transfers to minimize latency costs

       o      support for anonymous or authenticated rsync daemons (ideal for mirroring)

GENERAL
       Rsync  copies  files  either to or from a remote host, or locally on the current host
       (it does not support copying files between two remote hosts).

       There are two different ways for rsync to contact a remote system:  using  a  remote-
       shell  program  as  the  transport (such as ssh or rsh) or contacting an rsync daemon
       directly via TCP.  The remote-shell transport is used whenever the source or destina-
       tion path contains a single colon (:) separator after a host specification.  Contact-
:
```

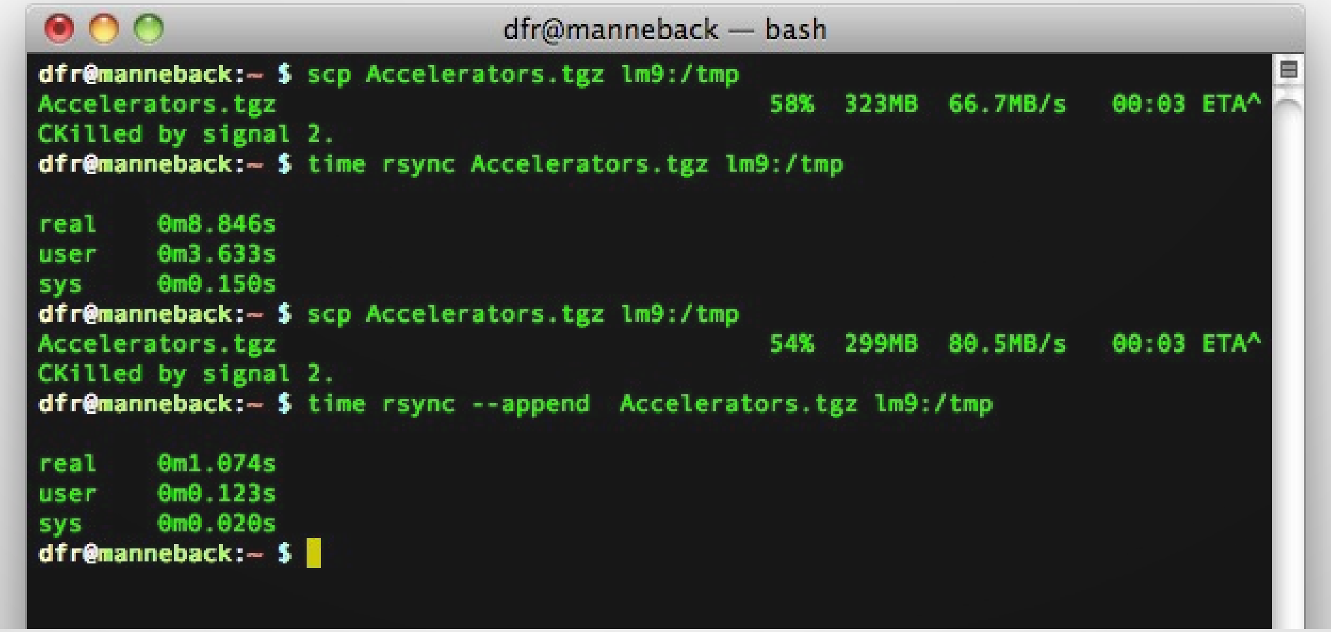
# Update only what changed: rsync

```
rsync [OPTIONS]... SRC [SRC]... [USER@]HOST:DEST
```

- Always use:  -az
- Other interesting arguments:
  - -v and --progress
  - --include or --exclude
  - --delete and/or --remove-source-file
  - --dry-run
  - --size-only or –checksum
- Works well with GNU parallel

# Resuming transfers

- When nothing changed but the transfer was interrupted
  - append: do not re-check partially transmitted files and resume the transfer where it was abandoned assuming first transfer attempt was with scp or with rsync --inplace



```
dfr@manneback:~ $ scp Accelerators.tgz lm9:/tmp
Accelerators.tgz                                58%  323MB  66.7MB/s   00:03 ETA^
CKilled by signal 2.
dfr@manneback:~ $ time rsync Accelerators.tgz lm9:/tmp

real    0m8.846s
user    0m3.633s
sys     0m0.150s
dfr@manneback:~ $ scp Accelerators.tgz lm9:/tmp
Accelerators.tgz                                54%  299MB  80.5MB/s   00:03 ETA^
CKilled by signal 2.
dfr@manneback:~ $ time rsync --append  Accelerators.tgz lm9:/tmp

real    0m1.074s
user    0m0.123s
sys     0m0.020s
dfr@manneback:~ $ 
```

# TAR | SSH

- From local to remote

```
$ tar zvzf - /path/to/data | ssh server "cat > /srv/data_server1.tar.gz"
```

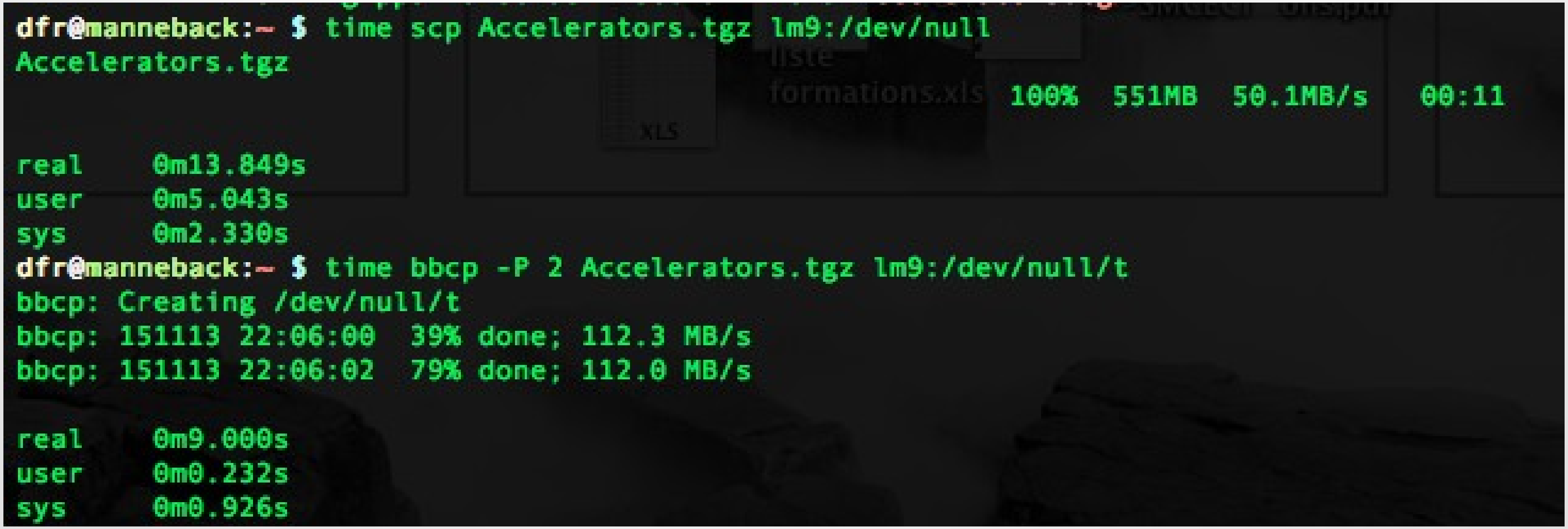
- From remote to local

```
$ ssh server tar czf - /path/to/data/ > ./data_server.tar.gz
```

- Avoid a lot of communication overhead linked to inodes
- Use the pv command to get a progress bar

# Parallel data transfer: bbcp

- Better use of the bandwidth than SCP
- Needs to be installed on both sides (easy to install)
- Needs friendly firewalls

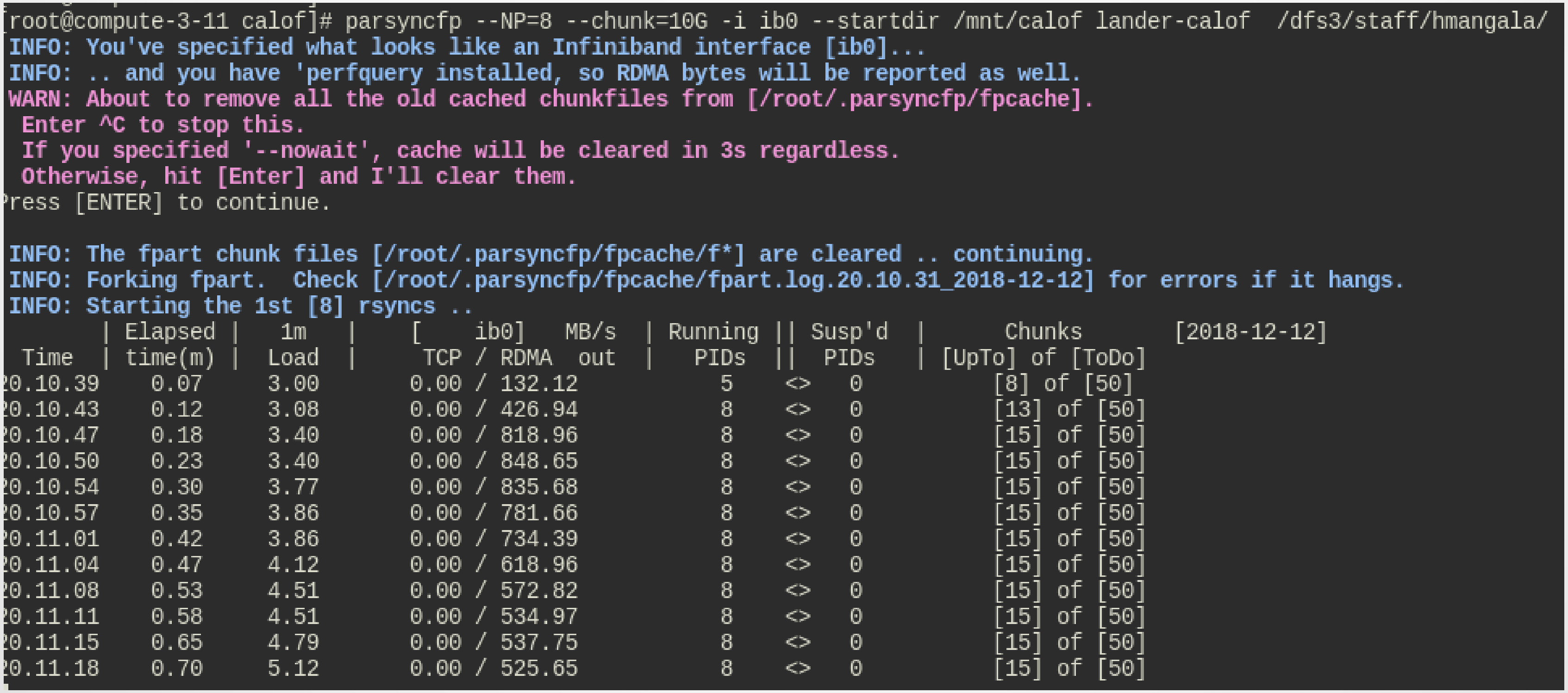


```
dfr@manneback:~ $ time scp Accelerators.tgz lm9:/dev/null
Accelerators.tgz
                                                      100%  551MB  50.1MB/s   00:11

real    0m13.849s
user    0m5.043s
sys     0m2.330s
dfr@manneback:~ $ time bbcp -P 2 Accelerators.tgz lm9:/dev/null/t
bbcp: Creating /dev/null/t
bbcp: 151113 22:06:00  39% done; 112.3 MB/s
bbcp: 151113 22:06:02  79% done; 112.0 MB/s

real    0m9.000s
user    0m0.232s
sys     0m0.926s
```

https://github.com/Stormwind99/bbcp

# Parallel rsync

- Parsyncfp

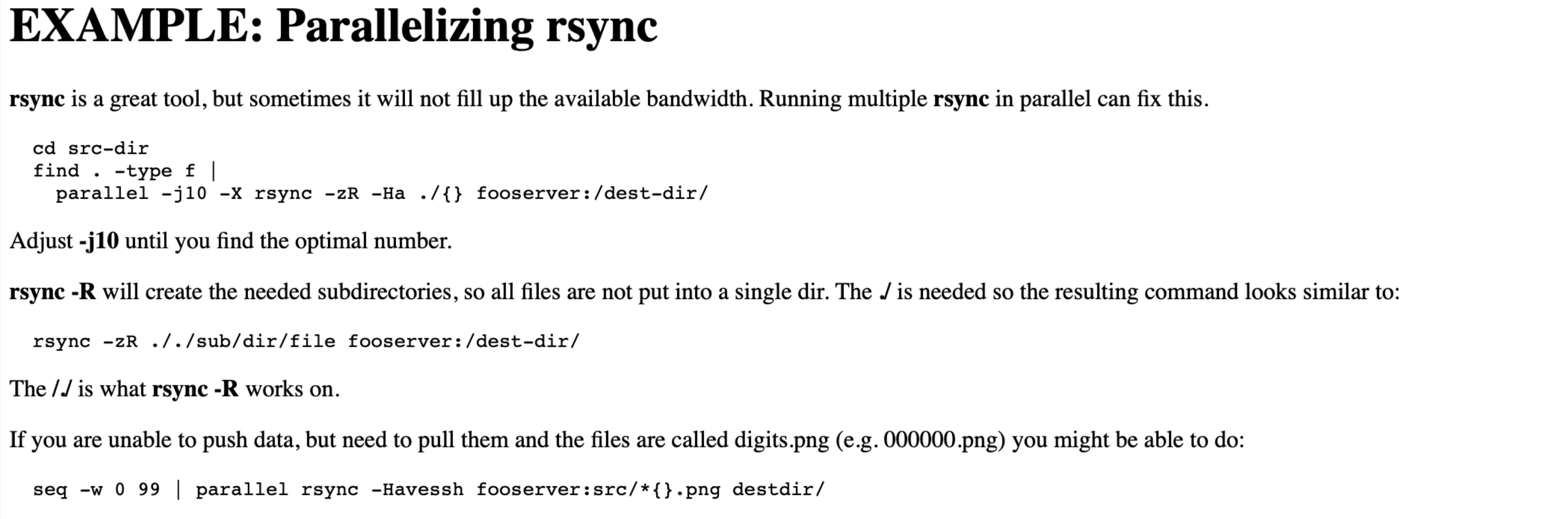


```
[root@compute-3-11 calof]# parsyncfp --NP=8 --chunk=10G -i ib0 --startdir /mnt/calof lander-calof  /dfs3/staff/hmangala/
INFO: You've specified what looks like an Infiniband interface [ib0]...
INFO: .. and you have 'perfquery installed, so RDMA bytes will be reported as well.
WARN: About to remove all the old cached chunkfiles from [/root/.parsyncfp/fpcache].
 Enter ^C to stop this.
 If you specified '--nowait', cache will be cleared in 3s regardless.
 Otherwise, hit [Enter] and I'll clear them.
Press [ENTER] to continue.

INFO: The fpart chunk files [/root/.parsyncfp/fpcache/f*] are cleared .. continuing.
INFO: Forking fpart.  Check [/root/.parsyncfp/fpcache/fpart.log.20.10.31_2018-12-12] for errors if it hangs.
INFO: Starting the 1st [8] rsyncs ..
        | Elapsed |   1m   |    [    ib0]   MB/s  | Running || Susp'd  |      Chunks       [2018-12-12]
  Time  | time(m) |  Load  |     TCP / RDMA  out  |   PIDs  ||  PIDs   | [UpTo] of [ToDo]
20.10.39    0.07     3.00       0.00 / 132.12          5    <>   0          [8] of [50]
20.10.43    0.12     3.08       0.00 / 426.94          8    <>   0          [13] of [50]
20.10.47    0.18     3.40       0.00 / 818.96          8    <>   0          [15] of [50]
20.10.50    0.23     3.40       0.00 / 848.65          8    <>   0          [15] of [50]
20.10.54    0.30     3.77       0.00 / 835.68          8    <>   0          [15] of [50]
20.10.57    0.35     3.86       0.00 / 781.66          8    <>   0          [15] of [50]
20.11.01    0.42     3.86       0.00 / 734.39          8    <>   0          [15] of [50]
20.11.04    0.47     4.12       0.00 / 618.96          8    <>   0          [15] of [50]
20.11.08    0.53     4.51       0.00 / 572.82          8    <>   0          [15] of [50]
20.11.11    0.58     4.51       0.00 / 534.97          8    <>   0          [15] of [50]
20.11.15    0.65     4.79       0.00 / 537.75          8    <>   0          [15] of [50]
20.11.18    0.70     5.12       0.00 / 525.65          8    <>   0          [15] of [50]
```

https://github.com/hjmangalam/parsyncfp

# Parallel rsync

- GNU Parallel + rsync



## EXAMPLE: Parallelizing rsync

**rsync** is a great tool, but sometimes it will not fill up the available bandwidth. Running multiple **rsync** in parallel can fix this.

```
  cd src-dir
  find . -type f |
    parallel -j10 -X rsync -zR -Ha ./{} fooserver:/dest-dir/
```

Adjust **-j10** until you find the optimal number.

**rsync -R** will create the needed subdirectories, so all files are not put into a single dir. The **./** is needed so the resulting command looks similar to:

```
  rsync -zR ././sub/dir/file fooserver:/dest-dir/
```

The **/./** is what **rsync -R** works on.

If you are unable to push data, but need to pull them and the files are called digits.png (e.g. 000000.png) you might be able to do:

```
  seq -w 0 99 | parallel rsync -Havessh fooserver:src/*{}.png destdir/
```

Limit the depth of `find` can help speed things up

https://www.gnu.org/software/parallel/man.html#EXAMPLE:-Parallelizing-rsync

# 3. Data sharing

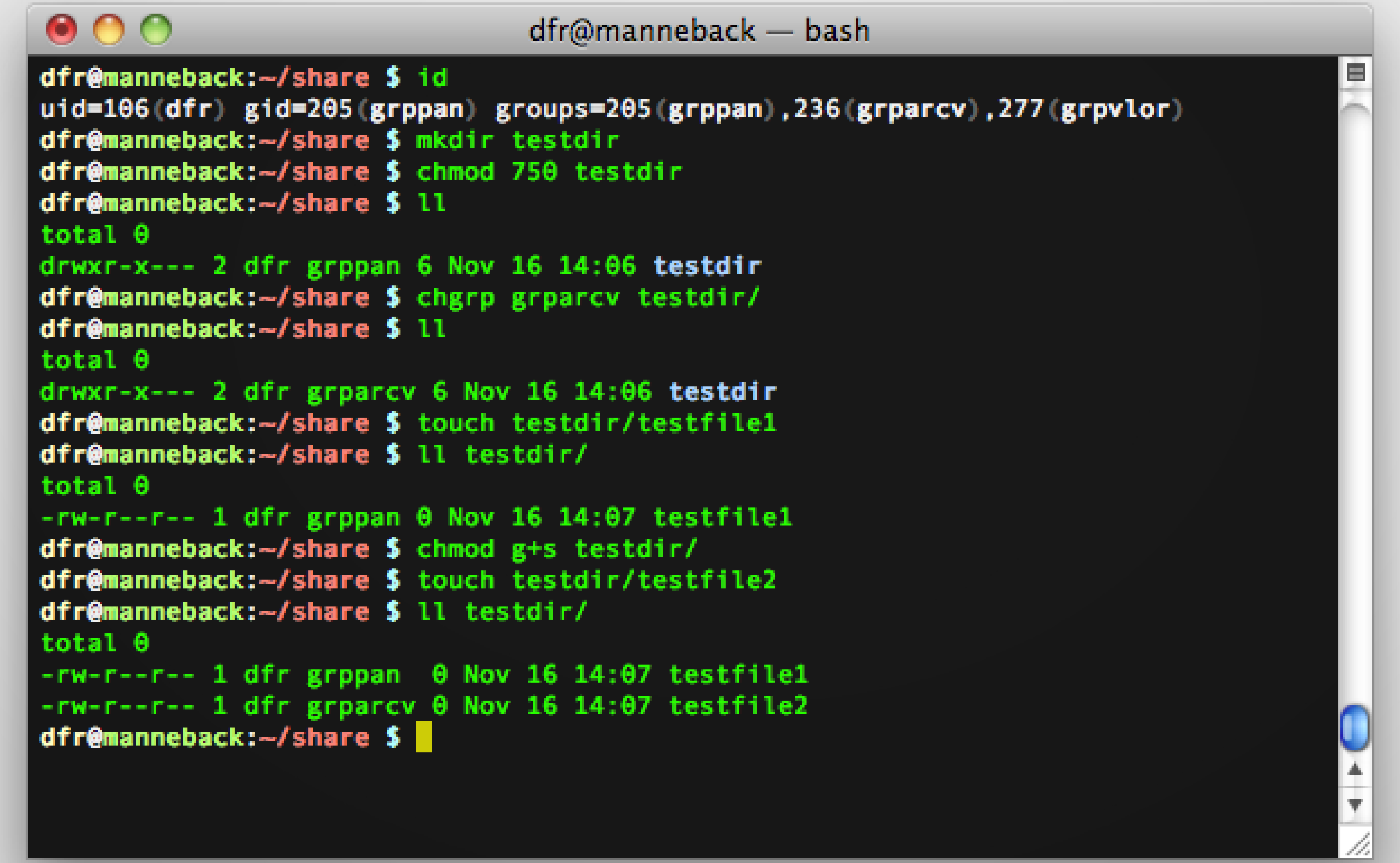
with other users (Unix permissions, Encryption)

with external users (Owncloud, Dataverse)

# Data sharing

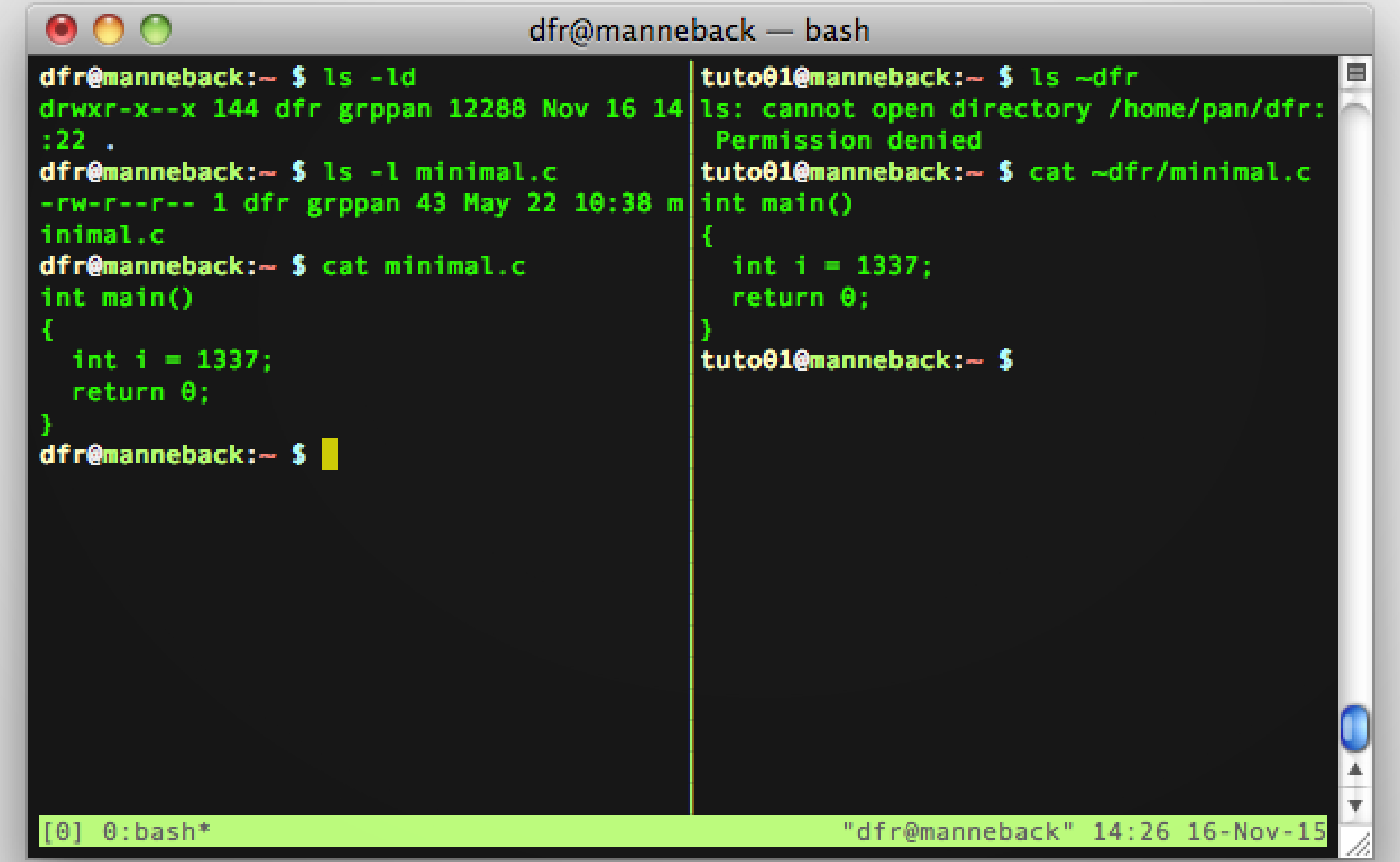
Data sharing with other users

# Sharing with the group



```
dfr@manneback:~/share $ id
uid=106(dfr) gid=205(grppan) groups=205(grppan),236(grparcv),277(grpvlor)
dfr@manneback:~/share $ mkdir testdir
dfr@manneback:~/share $ chmod 750 testdir
dfr@manneback:~/share $ ll
total 0
drwxr-x--- 2 dfr grppan 6 Nov 16 14:06 testdir
dfr@manneback:~/share $ chgrp grparcv testdir/
dfr@manneback:~/share $ ll
total 0
drwxr-x--- 2 dfr grparcv 6 Nov 16 14:06 testdir
dfr@manneback:~/share $ touch testdir/testfile1
dfr@manneback:~/share $ ll testdir/
total 0
-rw-r--r-- 1 dfr grppan 0 Nov 16 14:07 testfile1
dfr@manneback:~/share $ chmod g+s testdir/
dfr@manneback:~/share $ touch testdir/testfile2
dfr@manneback:~/share $ ll testdir/
total 0
-rw-r--r-- 1 dfr grppan  0 Nov 16 14:07 testfile1
-rw-r--r-- 1 dfr grparcv 0 Nov 16 14:07 testfile2
dfr@manneback:~/share $ 
```

# Sharing and hiding

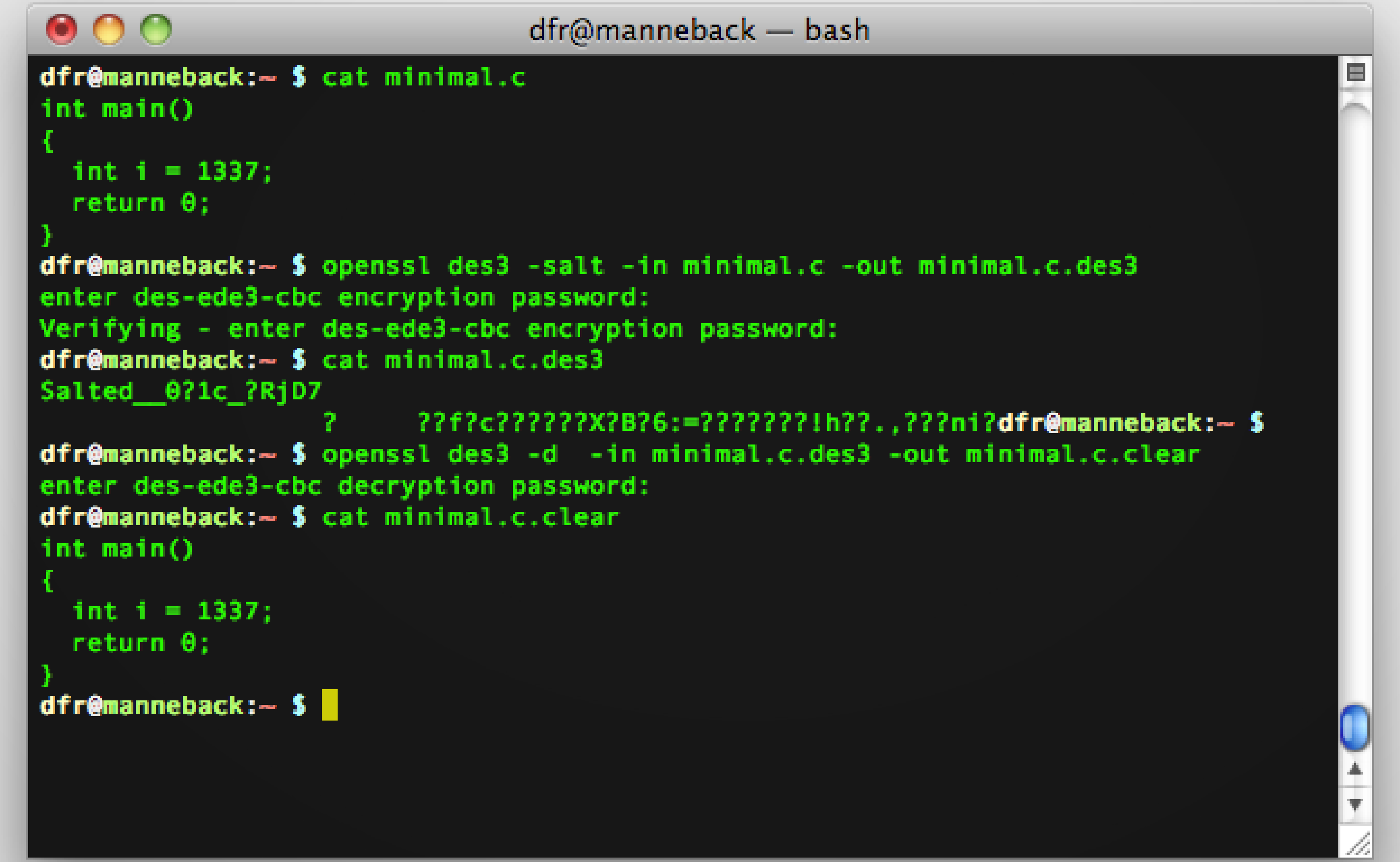


```
dfr@manneback:~ $ ls -ld
drwxr-x--x 144 dfr grppan 12288 Nov 16 14
:22 .
dfr@manneback:~ $ ls -l minimal.c
-rw-r--r-- 1 dfr grppan 43 May 22 10:38 m
inimal.c
dfr@manneback:~ $ cat minimal.c
int main()
{
  int i = 1337;
  return 0;
}
dfr@manneback:~ $
```

```
tuto01@manneback:~ $ ls ~dfr
ls: cannot open directory /home/pan/dfr:
 Permission denied
tuto01@manneback:~ $ cat ~dfr/minimal.c
int main()
{
  int i = 1337;
  return 0;
}
tuto01@manneback:~ $
```

```
[0] 0:bash*                                    "dfr@manneback" 14:26 16-Nov-15
```

# Sharing and encrypting



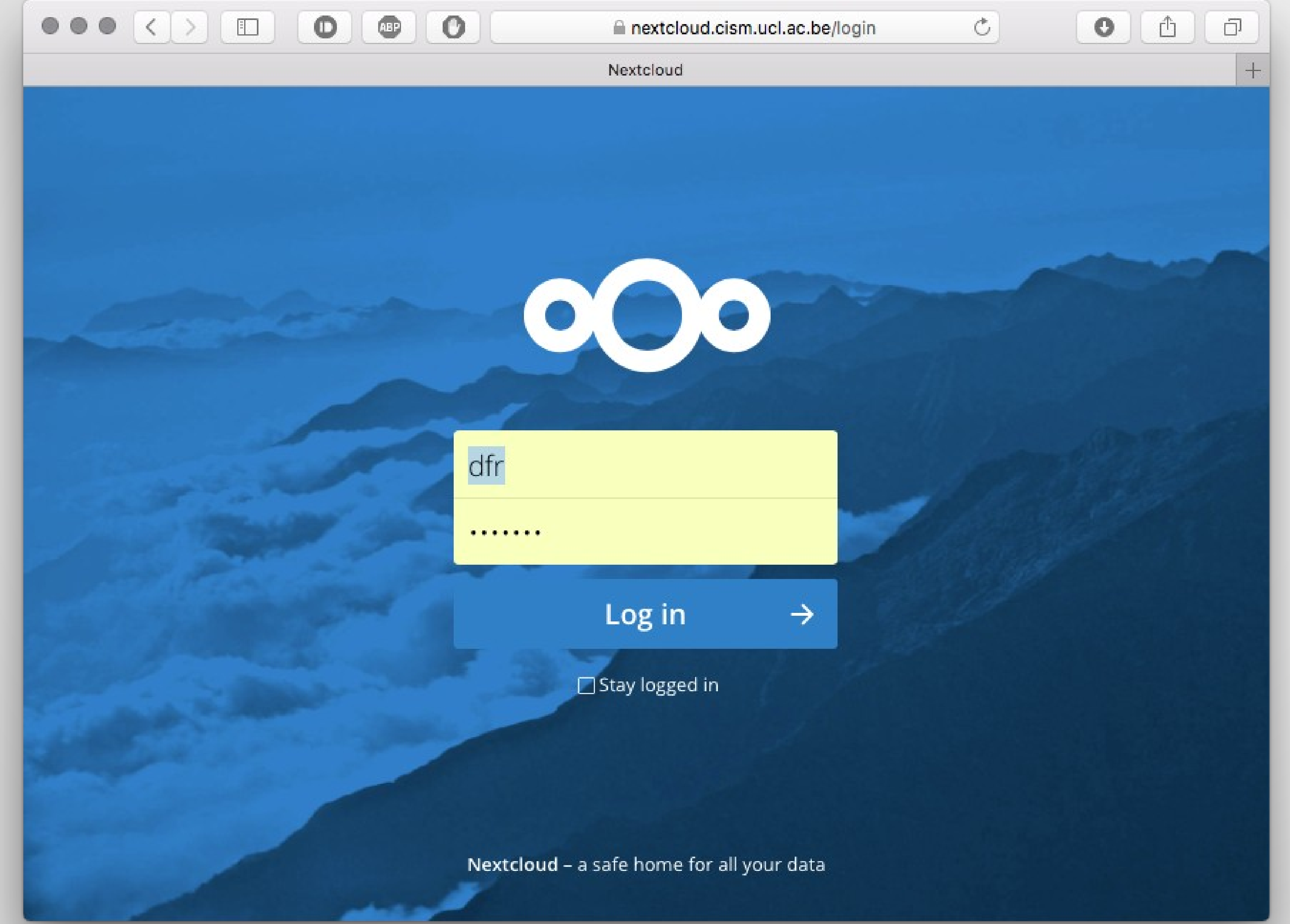
dfr@manneback — bash

```
dfr@manneback:~ $ cat minimal.c
int main()
{
  int i = 1337;
  return 0;
}
dfr@manneback:~ $ openssl des3 -salt -in minimal.c -out minimal.c.des3
enter des-ede3-cbc encryption password:
Verifying - enter des-ede3-cbc encryption password:
dfr@manneback:~ $ cat minimal.c.des3
Salted__0?1c_?RjD7
                  ?     ??f?c??????X?B?6:=???????!h??.,???ni?dfr@manneback:~ $
dfr@manneback:~ $ openssl des3 -d  -in minimal.c.des3 -out minimal.c.clear
enter des-ede3-cbc decryption password:
dfr@manneback:~ $ cat minimal.c.clear
int main()
{
  int i = 1337;
  return 0;
}
dfr@manneback:~ $ 
```

# Data sharing

Data sharing with external users

# Data sharing with external users



dfr

•••••••

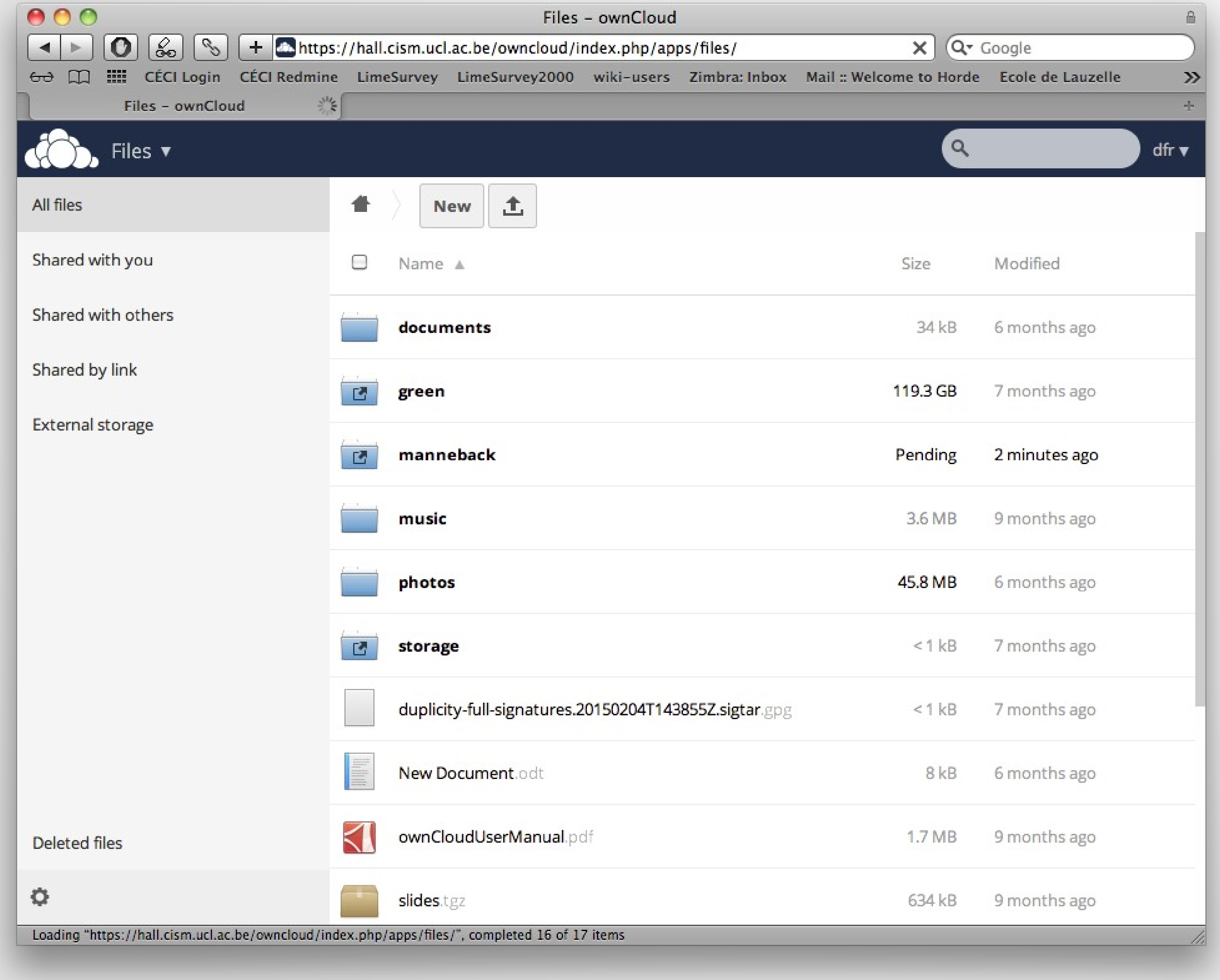
Log in →

□ Stay logged in

**Nextcloud** – a safe home for all your data

CISM login

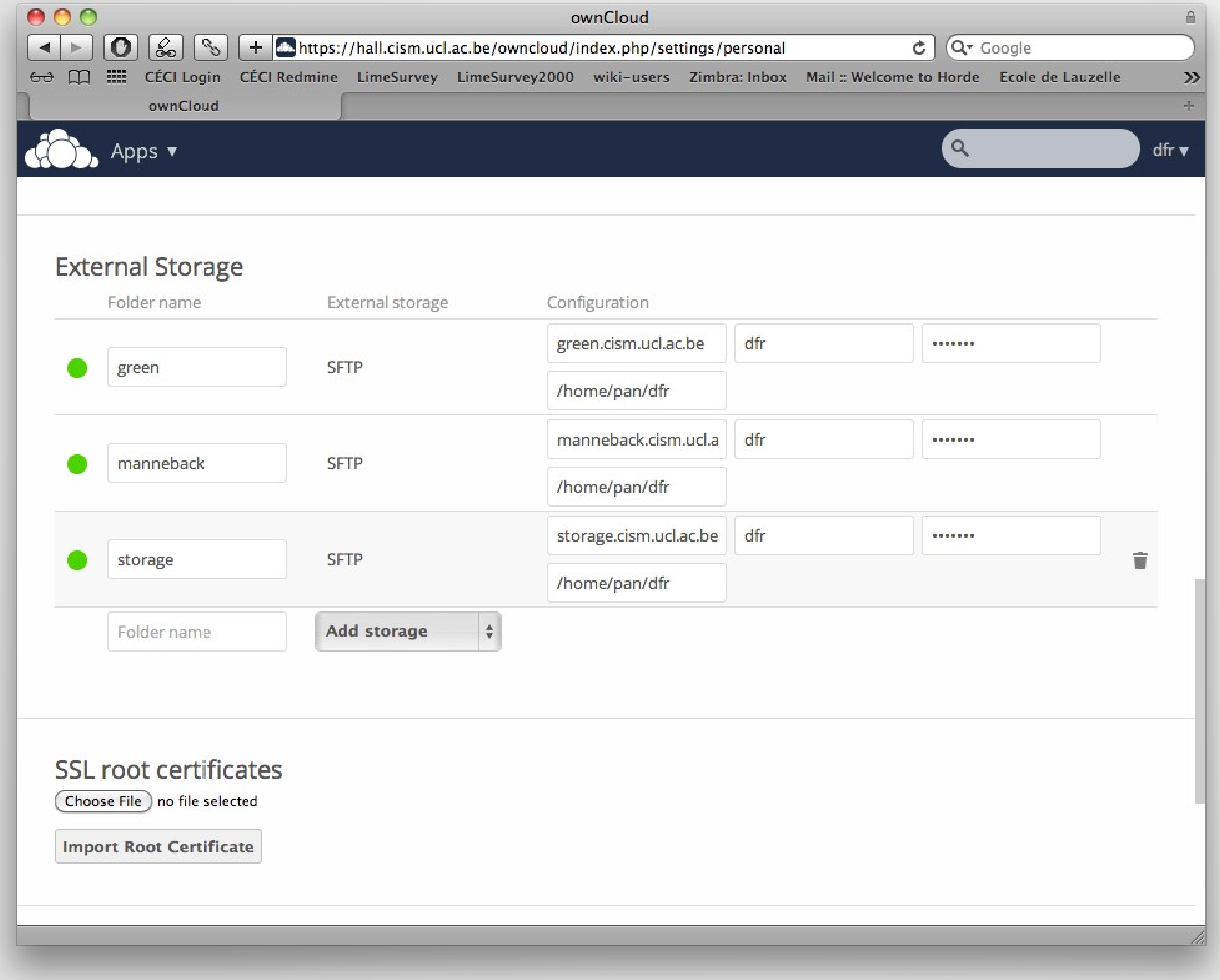
# Dropbox-like



| □ Name ▲ | Size | Modified |
|---|---|---|
| **documents** | 34 kB | 6 months ago |
| **green** | 119.3 GB | 7 months ago |
| **manneback** | Pending | 2 minutes ago |
| **music** | 3.6 MB | 9 months ago |
| **photos** | 45.8 MB | 6 months ago |
| **storage** | < 1 kB | 7 months ago |
| duplicity-full-signatures.20150204T143855Z.sigtar.gpg | < 1 kB | 7 months ago |
| New Document.odt | 8 kB | 6 months ago |
| ownCloudUserManual.pdf | 1.7 MB | 9 months ago |
| slides.tgz | 634 kB | 9 months ago |

Loading "https://hall.cism.ucl.ac.be/owncloud/index.php/apps/files/", completed 16 of 17 items

# External SFTP connectors



## External Storage

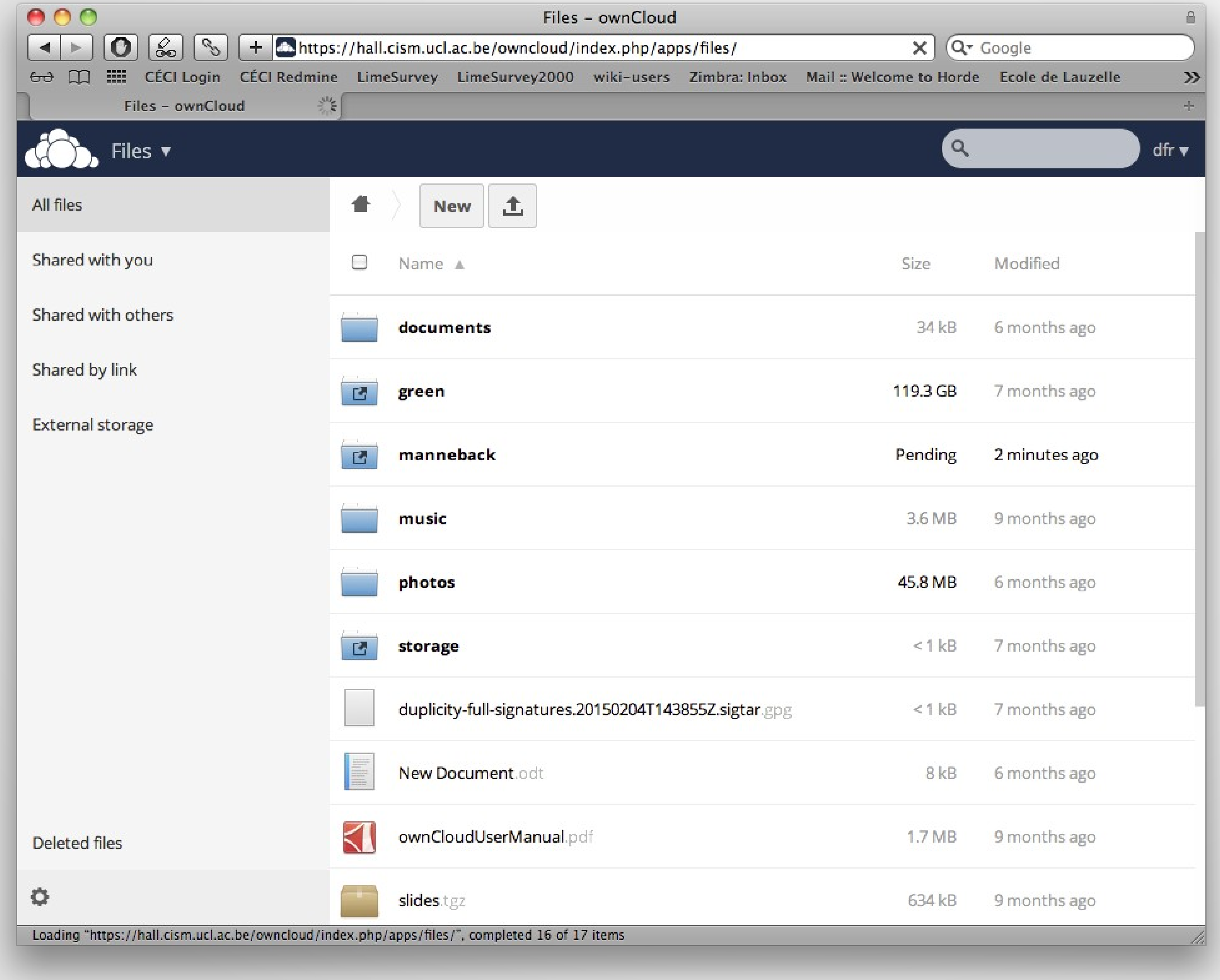
| | Folder name | External storage | Configuration | | |
|---|---|---|---|---|---|
| ● | green | SFTP | green.cism.ucl.ac.be<br>/home/pan/dfr | dfr | ••••••• |
| ● | manneback | SFTP | manneback.cism.ucl.a<br>/home/pan/dfr | dfr | ••••••• |
| ● | storage | SFTP | storage.cism.ucl.ac.be<br>/home/pan/dfr | dfr | ••••••• |
| | Folder name | Add storage | | | |

## SSL root certificates

Choose File no file selected

Import Root Certificate

# Dropbox-like

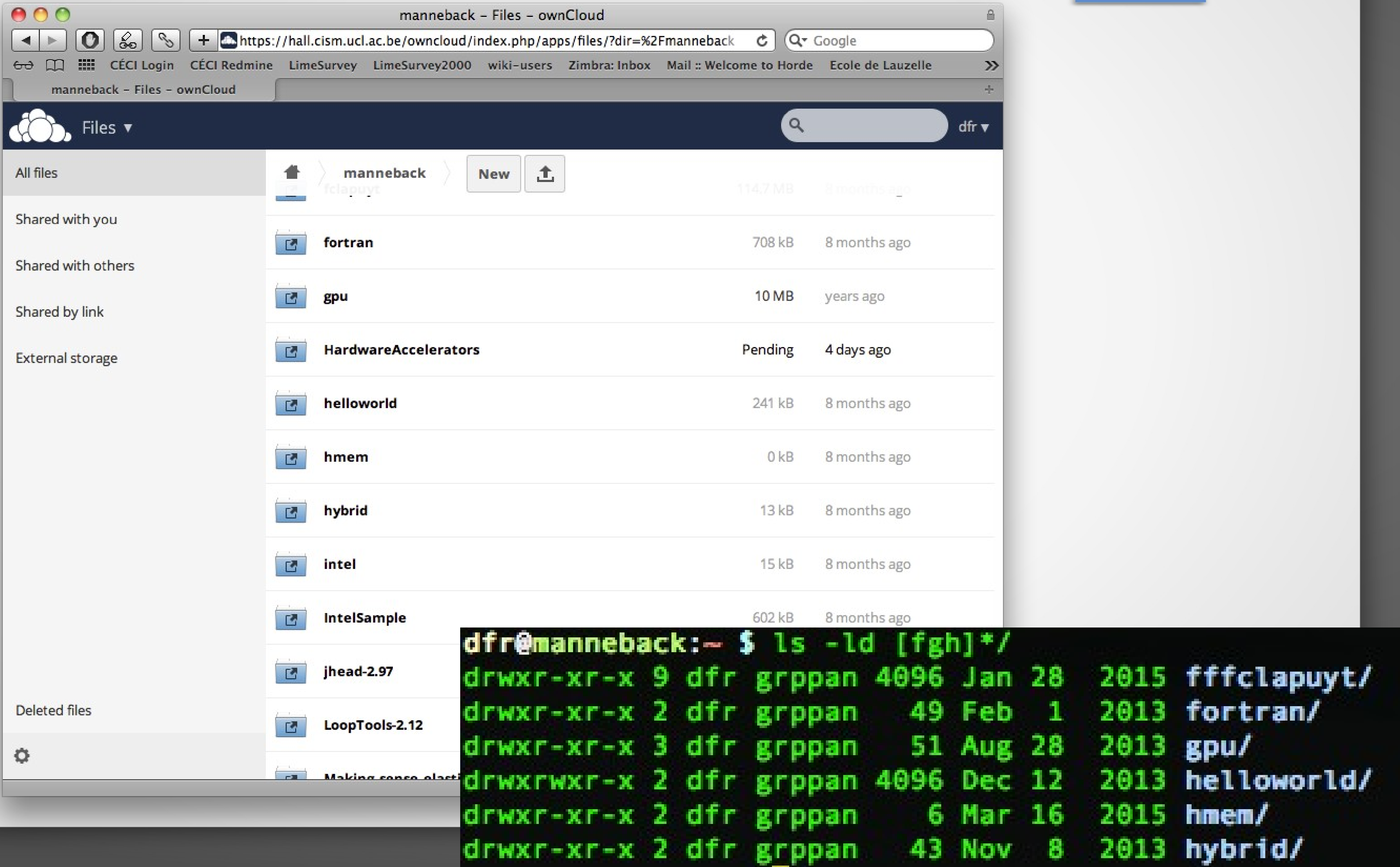


https://hall.cism.ucl.ac.be/owncloud/index.php/apps/files/

| Name ▲ | Size | Modified |
|---|---|---|
| **documents** | 34 kB | 6 months ago |
| **green** | 119.3 GB | 7 months ago |
| **manneback** | Pending | 2 minutes ago |
| **music** | 3.6 MB | 9 months ago |
| **photos** | 45.8 MB | 6 months ago |
| **storage** | < 1 kB | 7 months ago |
| duplicity-full-signatures.20150204T143855Z.sigtar.gpg | < 1 kB | 7 months ago |
| New Document.odt | 8 kB | 6 months ago |
| ownCloudUserManual.pdf | 1.7 MB | 9 months ago |
| slides.tgz | 634 kB | 9 months ago |

Loading "https://hall.cism.ucl.ac.be/owncloud/index.php/apps/files/", completed 16 of 17 items

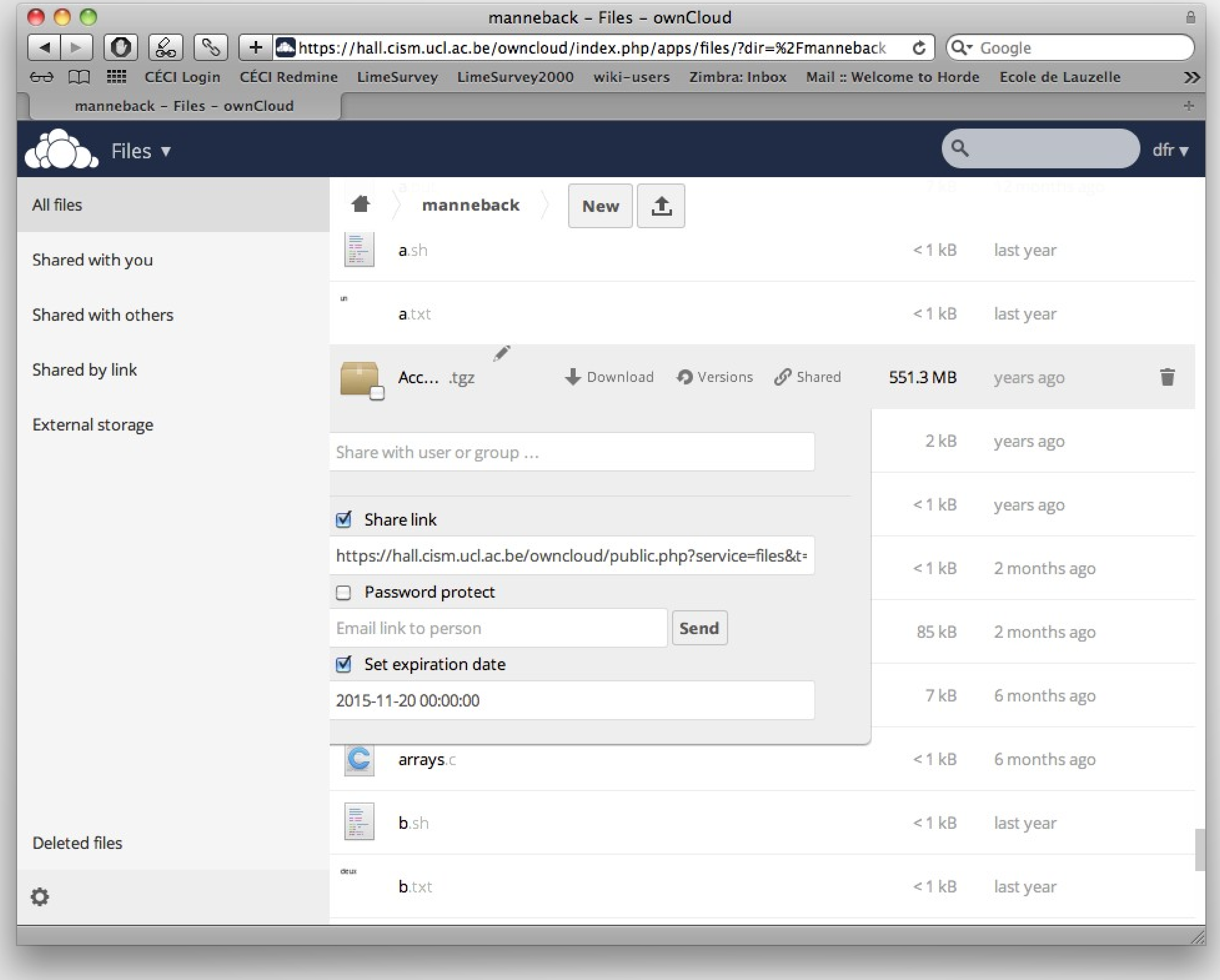
# My home on Manneback



| | | |
|---|---|---|
| fclapuyt | 114.7 MB | 8 months ago |
| fortran | 708 kB | 8 months ago |
| gpu | 10 MB | years ago |
| HardwareAccelerators | Pending | 4 days ago |
| helloworld | 241 kB | 8 months ago |
| hmem | 0 kB | 8 months ago |
| hybrid | 13 kB | 8 months ago |
| intel | 15 kB | 8 months ago |
| IntelSample | 602 kB | 8 months ago |
| jhead-2.97 | | |
| LoopTools-2.12 | | |
| Making sense elasti | | |

```
dfr@manneback:~ $ ls -ld [fgh]*/
drwxr-xr-x 9 dfr grppan 4096 Jan 28  2015 fffclapuyt/
drwxr-xr-x 2 dfr grppan   49 Feb  1  2013 fortran/
drwxr-xr-x 3 dfr grppan   51 Aug 28  2013 gpu/
drwxrwxr-x 2 dfr grppan 4096 Dec 12  2013 helloworld/
drwxr-xr-x 2 dfr grppan    6 Mar 16  2015 hmem/
drwxr-xr-x 2 dfr grppan   43 Nov  8  2013 hybrid/
```

# Can create a share URL



| | | |
|---|---|---|
| a.sh | < 1 kB | last year |
| a.txt | < 1 kB | last year |
| Acc… .tgz | 551.3 MB | years ago |
| | 2 kB | years ago |
| | < 1 kB | years ago |
| | < 1 kB | 2 months ago |
| | 85 kB | 2 months ago |
| | 7 kB | 6 months ago |
| arrays.c | < 1 kB | 6 months ago |
| b.sh | < 1 kB | last year |
| b.txt | < 1 kB | last year |

Share with user or group …

☑ Share link

https://hall.cism.ucl.ac.be/owncloud/public.php?service=files&t=

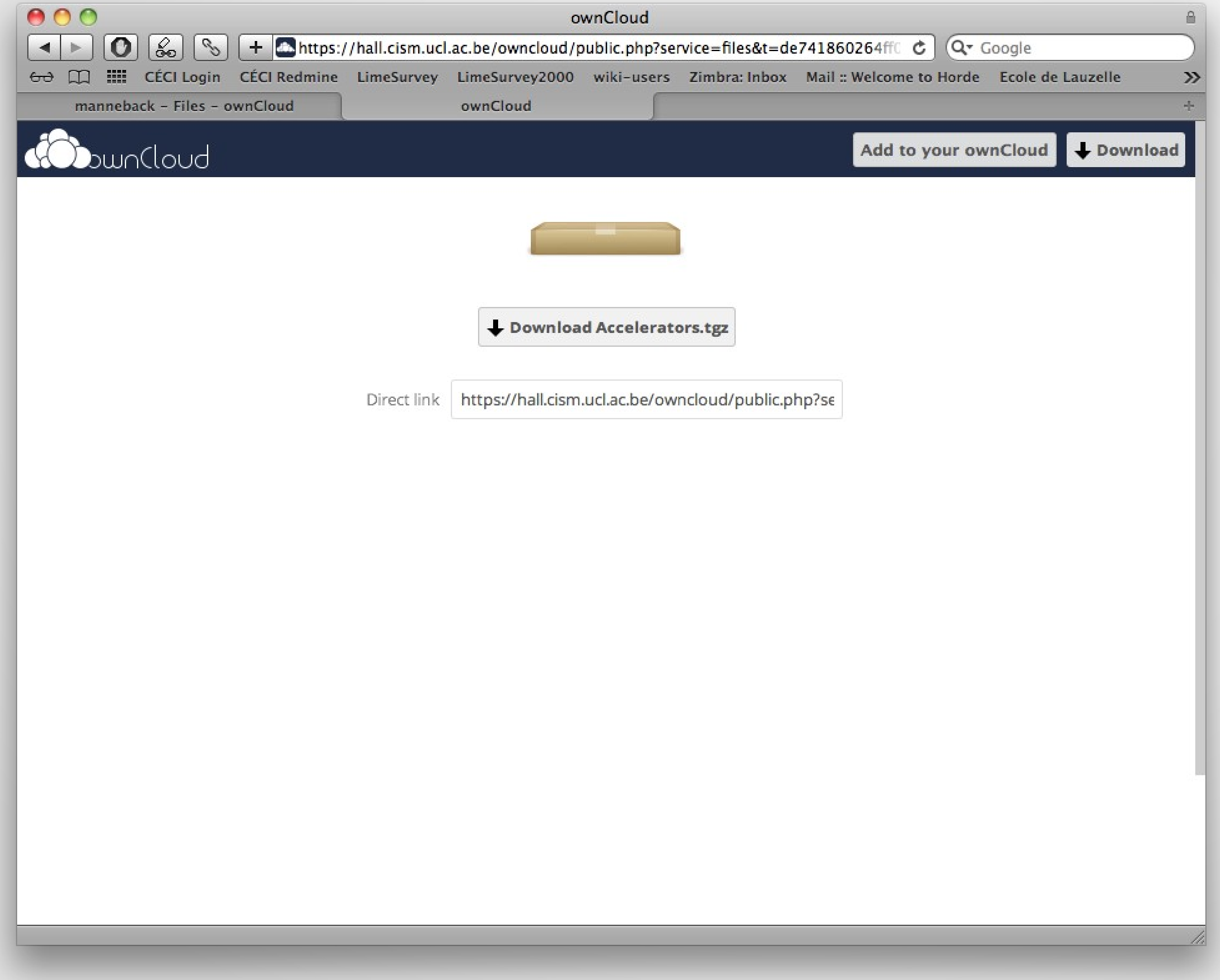
☐ Password protect

Email link to person [Send]

☑ Set expiration date

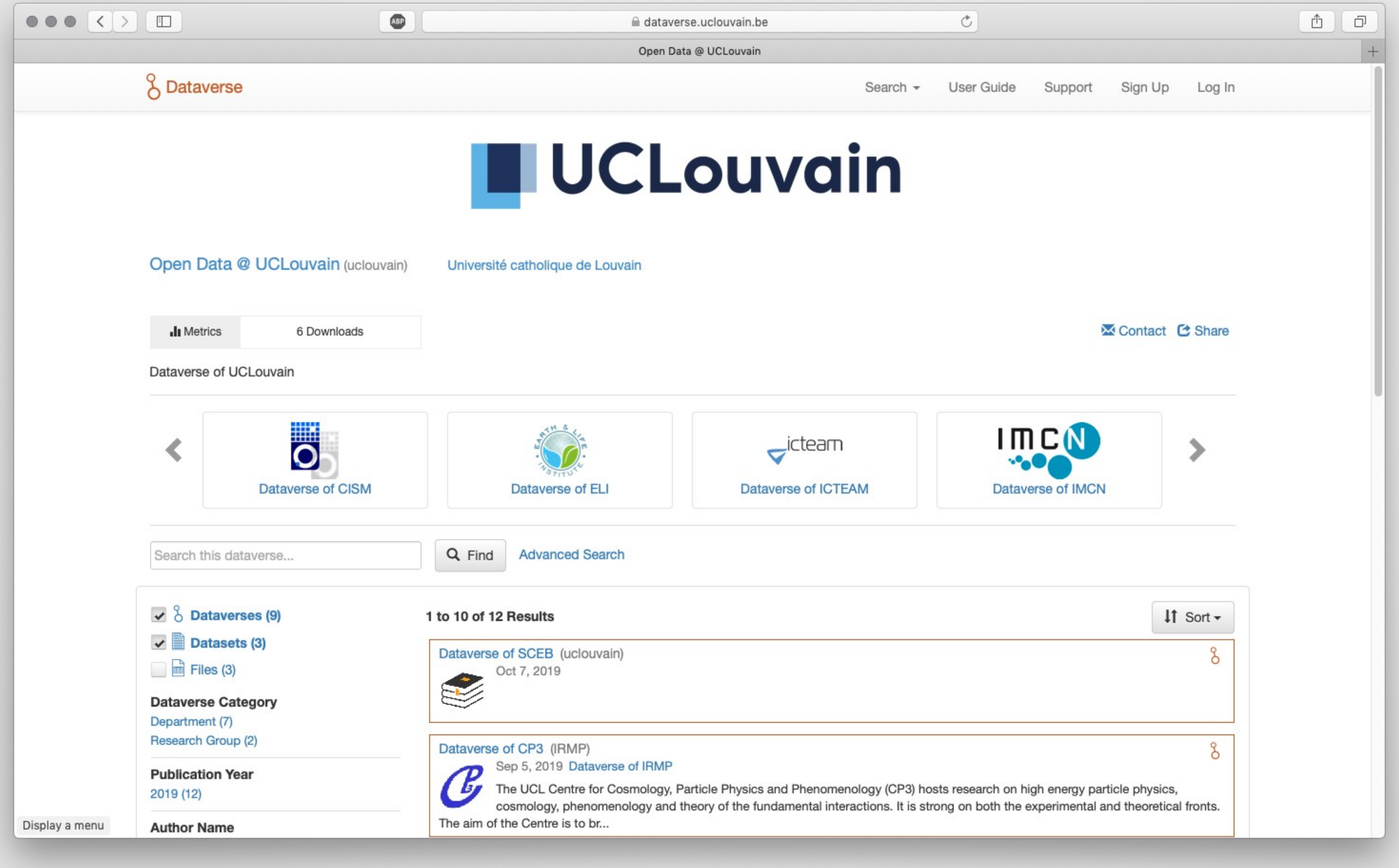
2015-11-20 00:00:00

# And distribute it



Direct link https://hall.cism.ucl.ac.be/owncloud/public.php?se

# Open data – FAIR data



dataverse.uclouvain.be

Open Data @ UCLouvain

Open Data @ UCLouvain (uclouvain)    Université catholique de Louvain

Metrics    6 Downloads

Contact    Share

Dataverse of UCLouvain

Dataverse of CISM    Dataverse of ELI    Dataverse of ICTEAM    Dataverse of IMCN

☑ Dataverses (9)
☑ Datasets (3)
□ Files (3)

**Dataverse Category**
Department (7)
Research Group (2)

**Publication Year**
2019 (12)

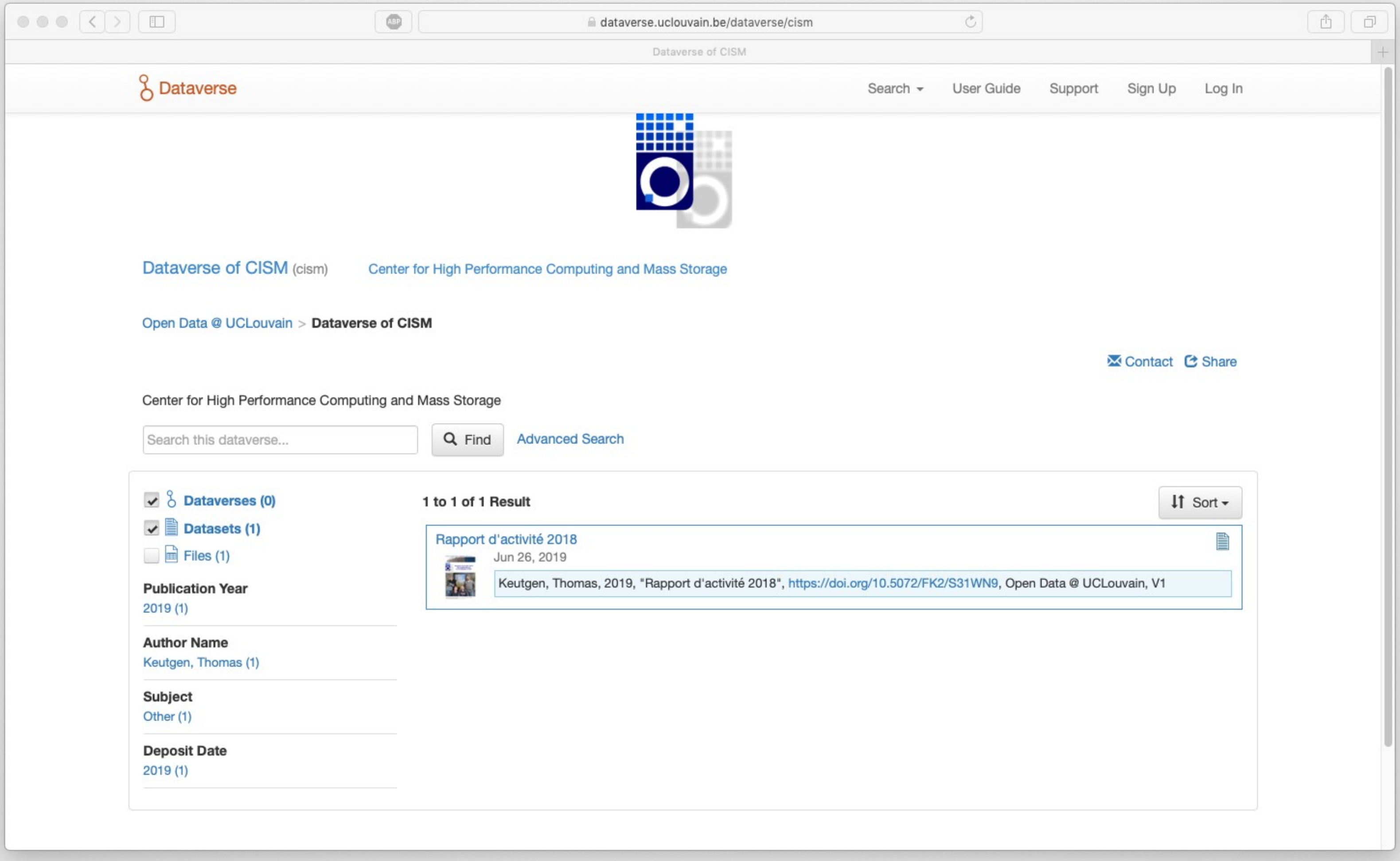
**Author Name**

**1 to 10 of 12 Results**

Dataverse of SCEB (uclouvain)
Oct 7, 2019

Dataverse of CP3 (IRMP)
Sep 5, 2019 Dataverse of IRMP
The UCL Centre for Cosmology, Particle Physics and Phenomenology (CP3) hosts research on high energy particle physics, cosmology, phenomenology and theory of the fundamental interactions. It is strong on both the experimental and theoretical fronts. The aim of the Centre is to br...

# Open data – FAIR data



Dataverse of CISM (cism) Center for High Performance Computing and Mass Storage

Open Data @ UCLouvain > **Dataverse of CISM**

Center for High Performance Computing and Mass Storage

Search this dataverse...

☑ **Dataverses (0)**

☑ **Datasets (1)**

□ Files (1)

**Publication Year**
2019 (1)

**Author Name**
Keutgen, Thomas (1)

**Subject**
Other (1)

**Deposit Date**
2019 (1)

**1 to 1 of 1 Result**

Rapport d'activité 2018
Jun 26, 2019
Keutgen, Thomas, 2019, "Rapport d'activité 2018", https://doi.org/10.5072/FK2/S31WN9, Open Data @ UCLouvain, V1

# Summary:

Storage: choose the right filesystem and the right file format

Transfer: use the parallel tools when possible

Sharing: use all the potential of the UNIX permissions and try Nextcloud and Dataverse