

Data transfer with UFTP

Bernd Schuller (b.schuller@fz-juelich.de)
Federated Systems and Data
Jülich Supercomputing Centre

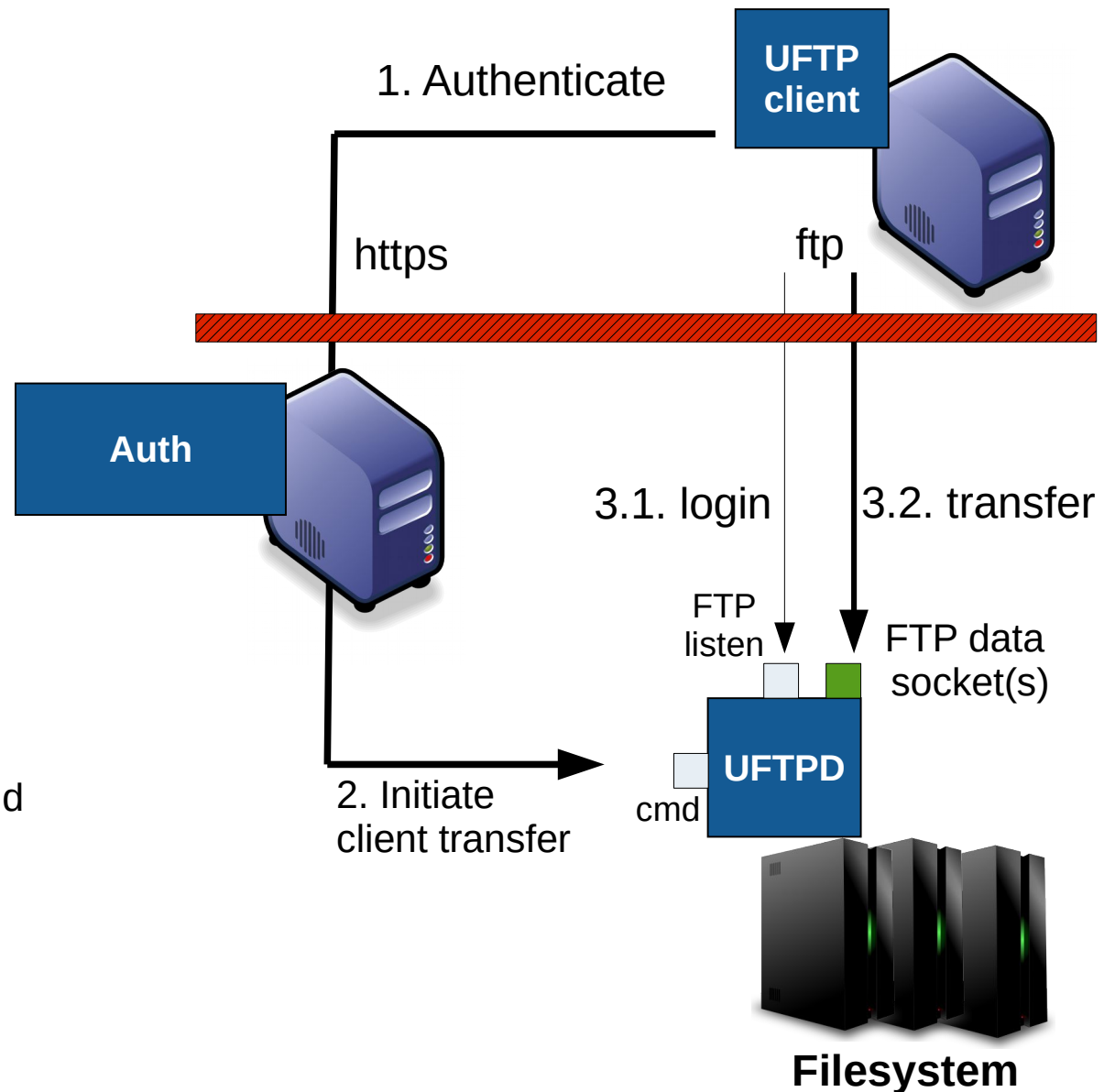
March 2022

- UFTP overview – functionalities and features
- Deployment at Jülich Supercomputing Centre (JSC)
- Use cases and usage examples
- Summary & Outlook

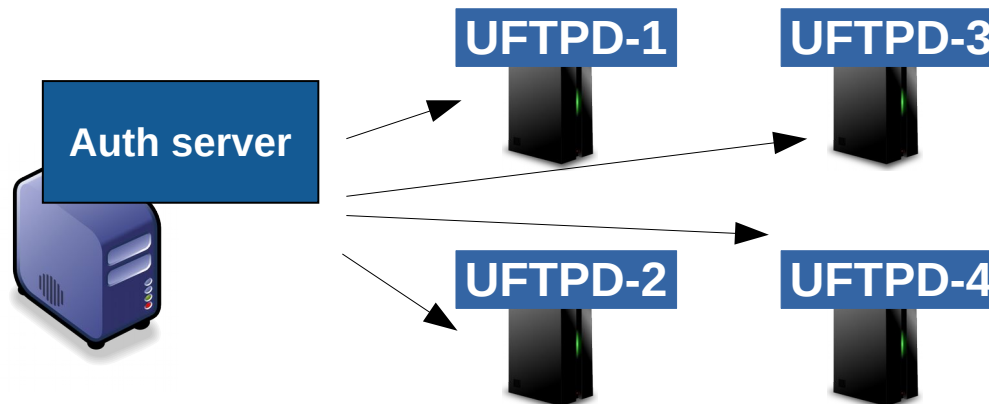
- „UNICORE FTP“ --- it grew out of the need for a high-performance data transfer for the UNICORE federated middleware (aka ‚Grid‘)
- Client/server file transfer system
- Based on the FTP protocol
(... but without FTP's security issues)
- Flexible integration options (authentication, user mapping), firewall-friendly
- Developed and maintained as Open Source
by Jülich Supercomputing Centre

UFTP architecture

- UFTP client
- UFTPD file server
- Authentication server
 - REST APIs
 - Client authentication, user ID mapping
 - Transfer setup, access control
 - Returns connection info and ONE TIME PASSWORD for logging into UFTPD



- FTP compliant
 - you can use standard FTP tools and libraries
- Additional options
 - compression and/or encryption of data stream(s)
 - multiple TCP streams per data connection
 - bandwidth management (both server and client)
- UFTPD Server supports partial file reads/writes
 - Create multiple sessions in parallel for scalability
- Multiple UFTPD backends are possible (round-robin)



- Recommended: ssh-style keys
 - For the ,uftp' client
 - At JSC: uftp use is separated from SSH use!!
- OAuth tokens
 - For (web) integration
 - Unity
 - Other OAuth IdPs
- Username/password
- ... are you missing anything? Contact us!

SSH-style keys

read from uftpd server!
\$HOME/.uftp/authorized_keys

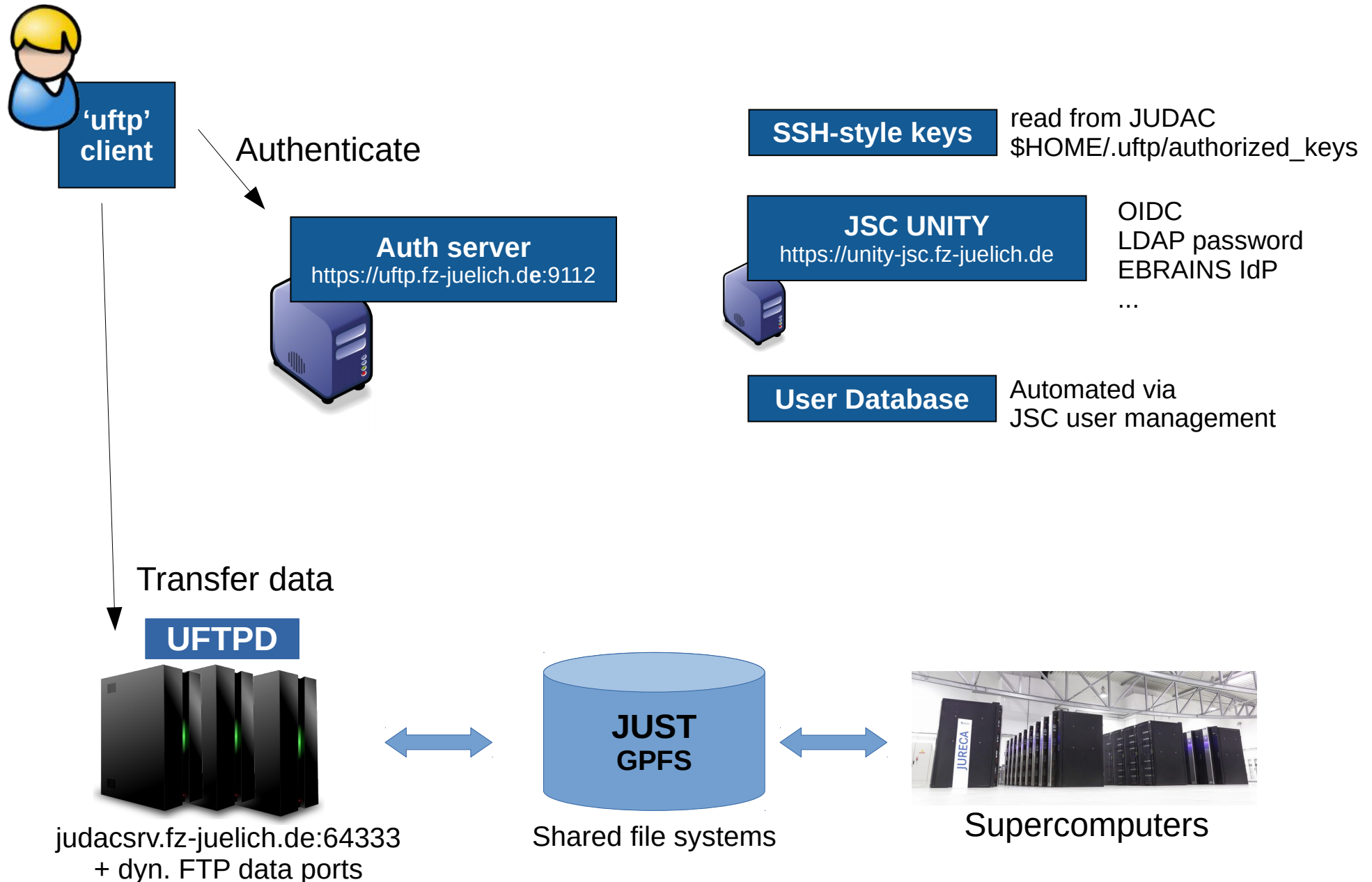


JSC UNITY

<https://unity-jsc.fz-juelich.de>

LDAP
HBP OIDC
...

JSC's production deployment



- Secure, high-performance data access/transfer
 - ,uftp' commandline client

■ Commands

- `cp, ls, mkdir, rm` – data access
- `checksum` – get MD5 / SHA* checksums
- `share` – create and manage shares
- `info` – get server and authentication information
- `auth` – authenticate only (for use with other ftp tools)

■ Pre-installed on JSC's HPC machines

- `module load uftp`

■ Download and use on your local workstation

- <https://sourceforge.net/projects/unicore/files/Clients/UFTP-Client/>
- Available as zip, deb, rpm
- Linux, Mac, Windows. Requires Java 8+

Setting up ssh-style keys for UFTP

- On the CLIENT: create dedicated key pair for UFTP use

UFTP
client

- `mkdir -p ~/.uftp`
- `ssh-keygen -f ~/.uftp/id_uftp`

- Copy the public key(s) to the correct location on the file SERVER (uftp):

UFTPD



`$HOME/.uftp/authorized_keys`

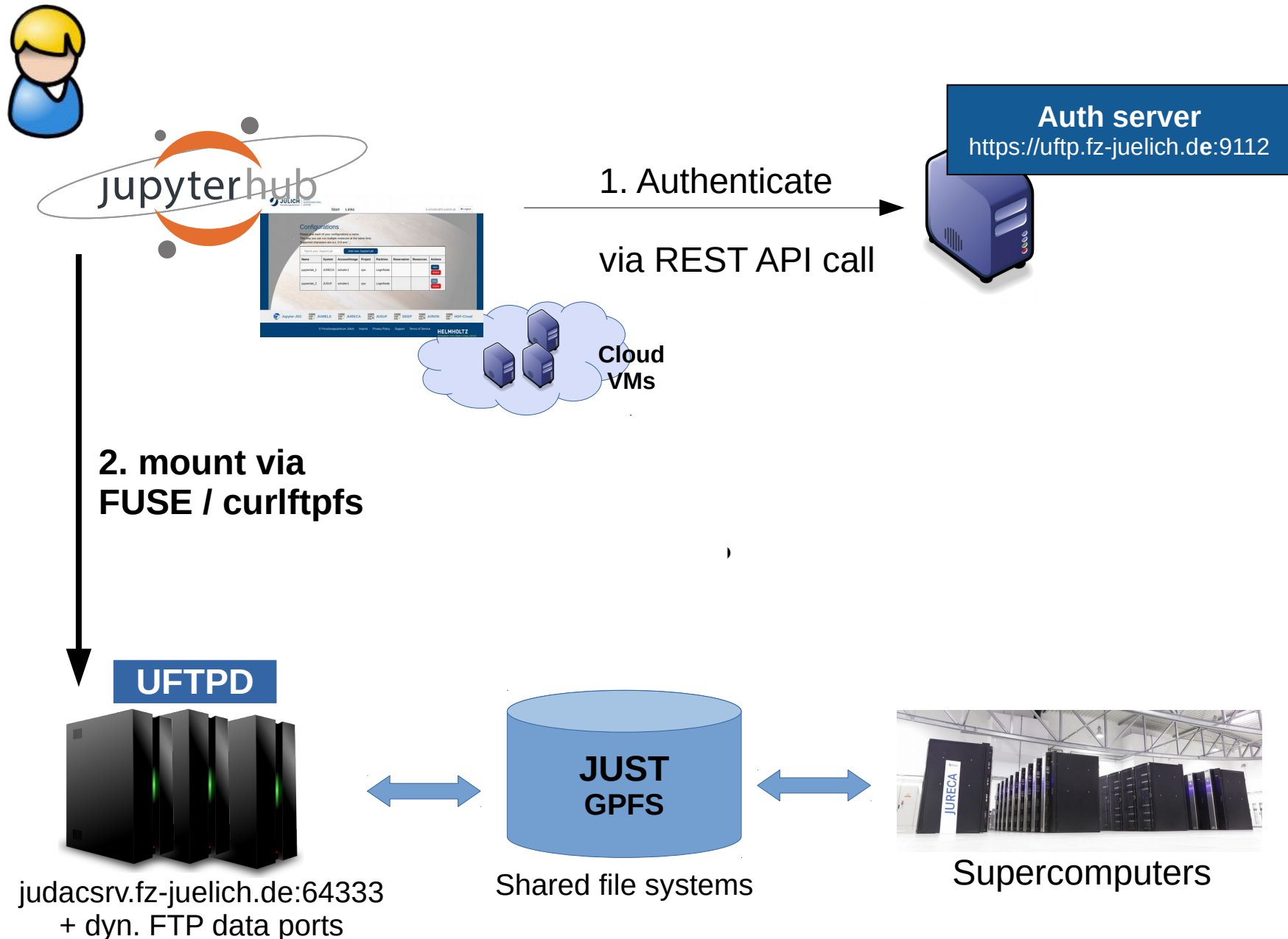
- `ssh userid@judac.fz-juelich.de mkdir -p .uftp/`
- `scp id_uftp.pub userid@judac.fz-juelich.de:`
- `ssh userid@judac.fz-juelich.de`
`cat id_uftp.pub >> .uftp/authorized_keys`

Demo: data management & transfer

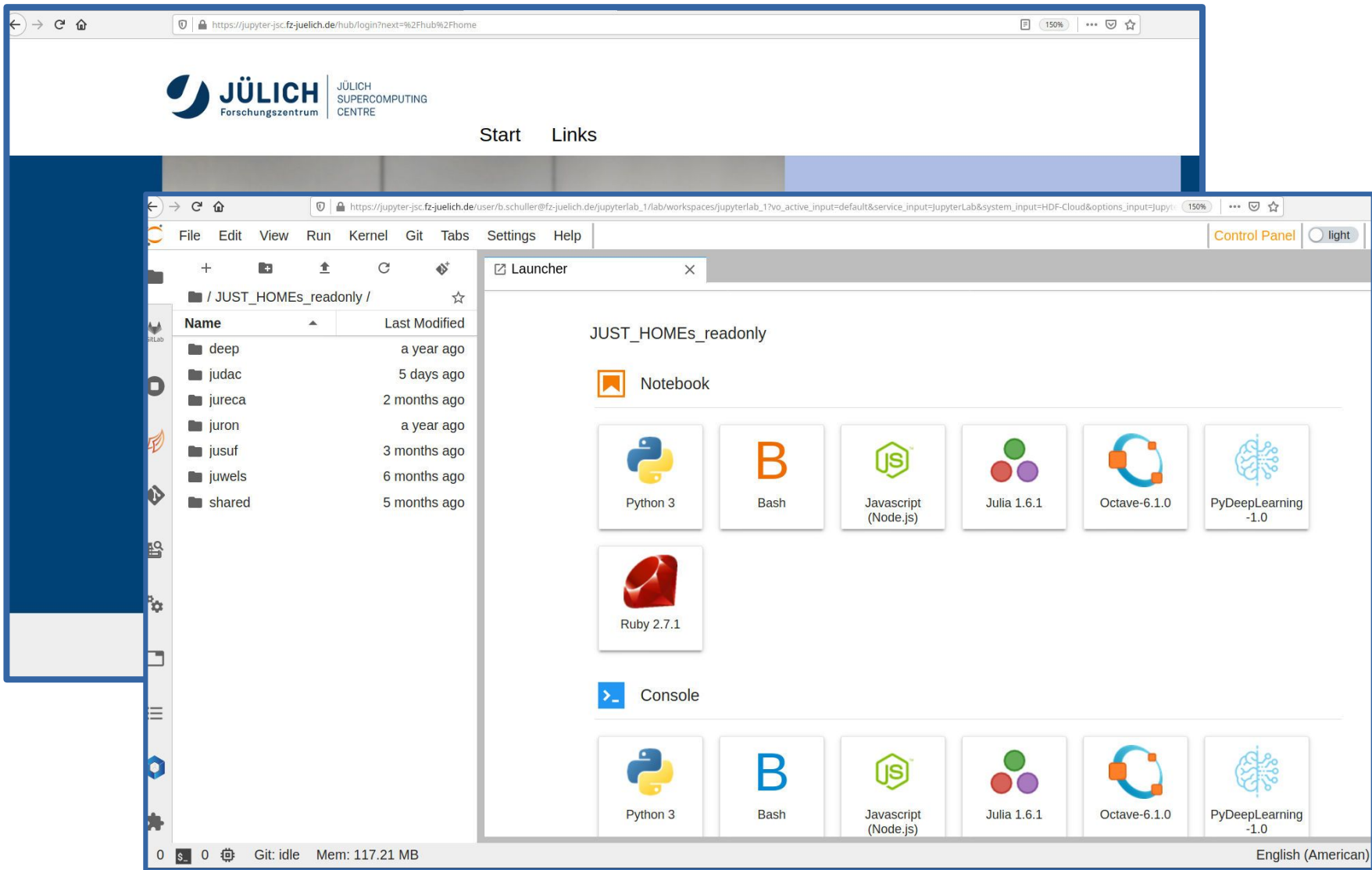
```
$ export JUDAC=https://uftp.fz-juelich.de:9112/UFTP_Auth/rest/auth/JUDAC:  
$ export USER=schuller1  
$ export ID=$HOME/.uftp/id_uftp  
  
$ uftp ls -v -u $USER -i $ID $JUDAC/p/projects/cjsc  
  
$ uftp cp -v -u $USER -i $ID $JUDAC/p/... <local_path>
```

- Integrate data access/transfer functionality into applications
 - REST APIs for authentication and FTP compliance for data access/transfer
- For example: mount HPC file systems
 - UFTP + FUSE / curlftpfs

Mounting HPC filesystems via UFTP and FUSE



User's GPFS \$HOME available on JupyterLab running on HDFCloud VM

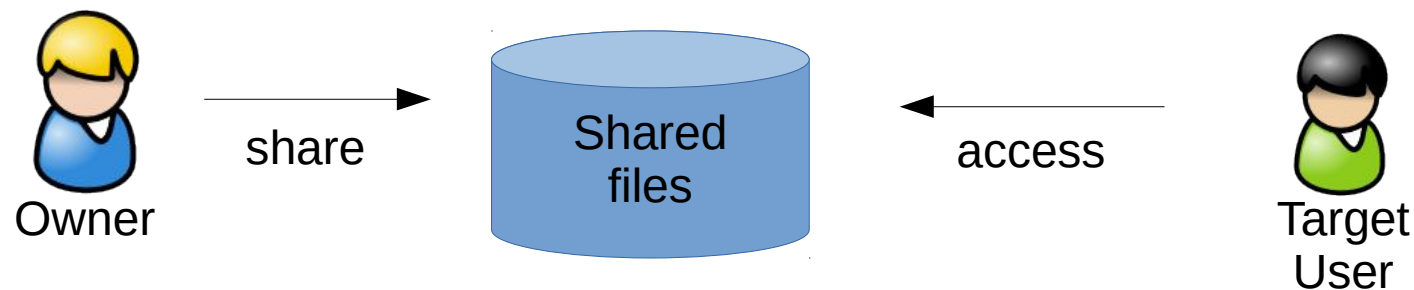


The screenshot displays the JupyterLab web interface. The top header shows the Jülich Forschungszentrum logo and navigation links for 'Start' and 'Links'. The main area is divided into two panes. The left pane, titled 'Launcher', shows a file explorer view of the directory '/ JUST_HOMEs_readonly /'. It lists several subdirectories: 'deep', 'judac', 'jureca', 'juron', 'jusuf', 'juwels', and 'shared', each with its last modified date. The right pane, titled 'Launcher', shows a grid of application launchers for various languages: Python 3, Bash, Javascript (Node.js), Julia 1.6.1, Octave-6.1.0, PyDeepLearning -1.0, and Ruby 2.7.1. The bottom status bar indicates 'Git: idle' and 'Mem: 117.21 MB'.

<https://jupyter-jsc.fz-juelich.de>

- Sharing of data sets

- Data sharing
 - The owner of a dataset wants to grant access to other users without direct access to the file system
- HPC environment
 - Potentially large files – want to avoid moving/copying
 - Transparent – don't want to change file ownership
 - Authenticated vs anonymous data access via HTTP



Data sharing - 2

■ Client

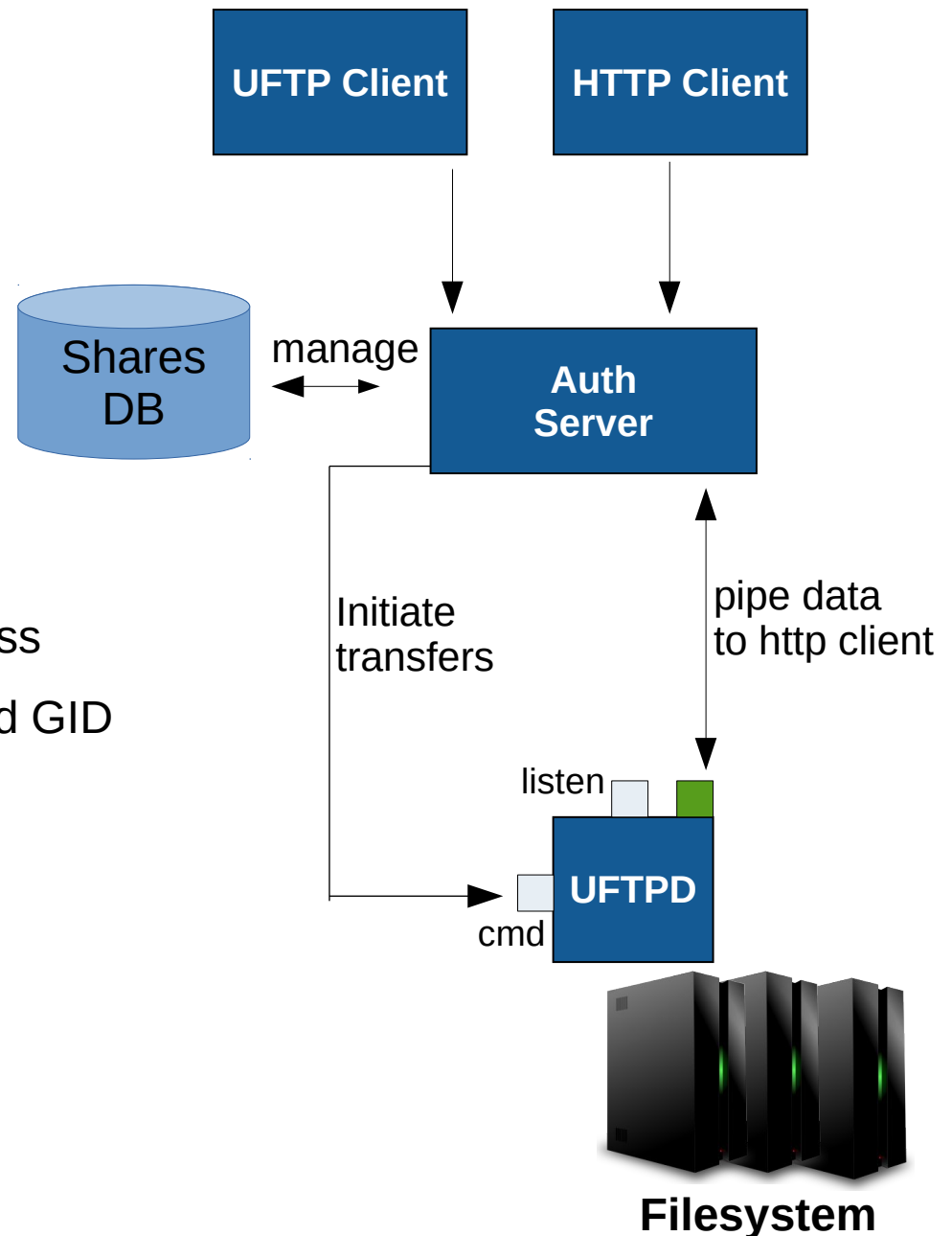
- Sharing: ,uftp' or REST API
- Access: https

■ Auth server

- Provides REST services for share management and file access
- Uses sharing user's Unix UID and GID to access data
- Pipes data from UFTPD to HTTP client

■ UFTPD

- File server



■ Owner shares a file

- Path (e.g. /data/hpc1/results/dataset.tgz)
- Anonymous access will be possible
 - *or to a specific target user*
(e.g. „CN=schuller1, OU=ssh-local-users“)
- Access type: ‚read‘ or ‚write‘

■ Target user accesses a shared file

- Using HTTP tool like wget/curl
- Using ‚uftp‘ client

■ Data provided via UFTPD (and via Auth server for https)

■ Owner's Unix account and group are used to access shared data

Demo: anonymous share and 'wget' download

```
$ export JUDAC=https://uftp.fz-juelich.de:9112/UFTP_Auth/rest/share/JUDAC  
$ export USER=schuller1  
$ export ID=$HOME/.uftp/id_uftp
```

```
$ uftp share -u $USER -i $ID --server $JUDAC /path/to/my/file.dat
```

Shared to: <URL>

```
$ wget --content-disposition <URL>
```

SUPERCOMPUTING AT THE LEADING EDGE



- UFTP in Gauss Centre for Supercomputing
 - HLRS, JSC, LRZ



- Secure, high-performance data access/transfer solution
 - Powerful ,uftp' commandline client
 - Scalable serverside architecture
 - **MANY integration options:** REST API / FTP compliance
- Data sharing in HPC environments
 - Make data sets accessible to „outside“ users (without HPC account)
- Production service at JSC
<https://apps.fz-juelich.de/jsc/hps/judac/uftp.html>
- Other HPC centers are offering UFTP access
 - HLRS, LRZ (InHPC.de project)

Software manuals

- <https://uftp-docs.readthedocs.io/en/latest/>

Source code (BSD-2 licensed)

- <https://github.com/UNICORE-EU/uftp>