

Mucura: your personal file repository in the cloud

F. Hernandez¹, W. Wu², R. Du², S. Li², W. Kan²

¹IN2P3/CNRS Computing Center & IHEP Computing Center | ²IHEP Computing Center



1 RESULTS

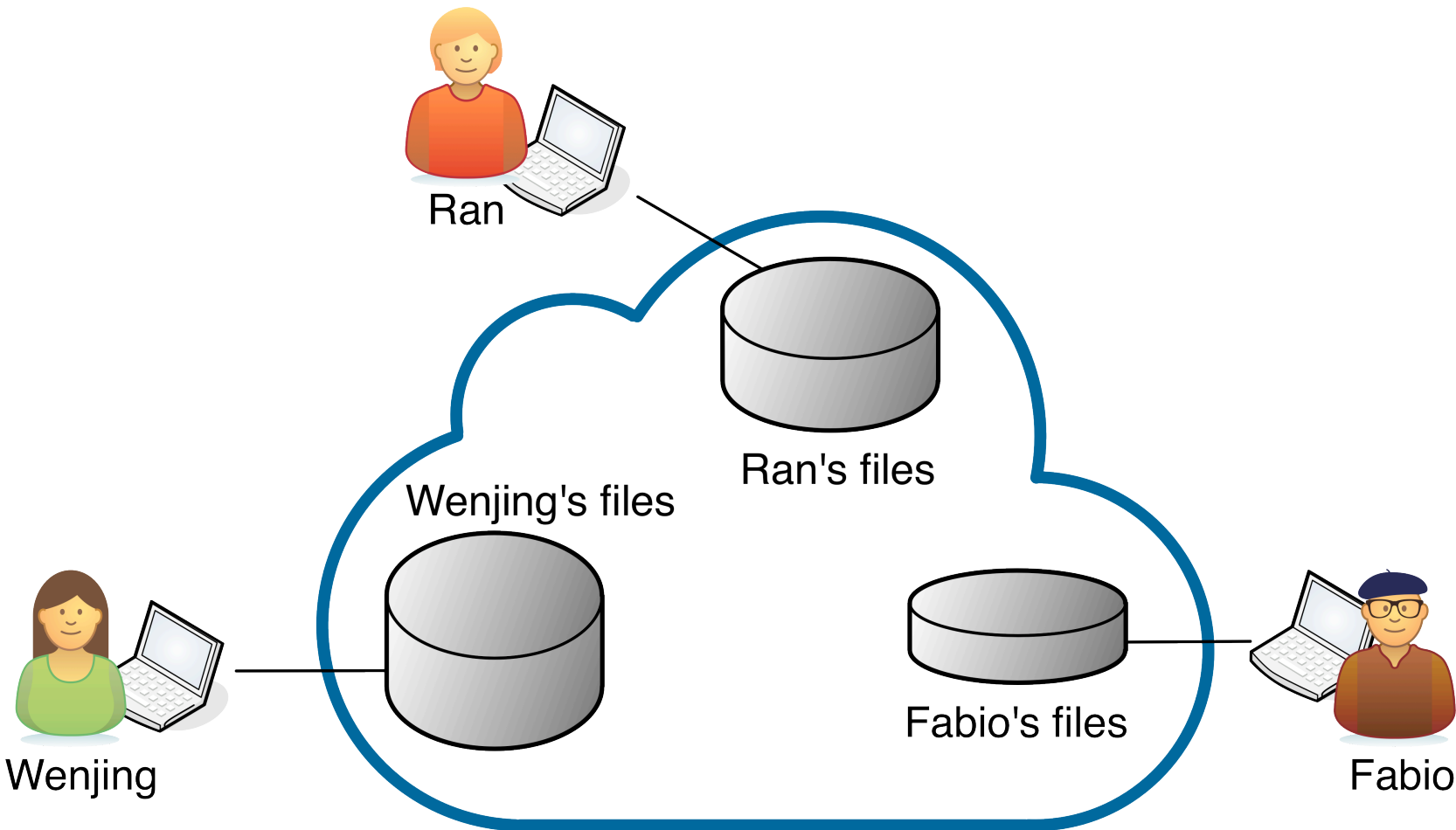
We implemented a prototype of a **remote file repository** backed by **key-value data stores**, accessible both **interactively** and from **grid jobs**.

2 GOAL

Our goal is to develop an **open-source software system** for operating **highly-available remote file repositories**.

Targets:

- end-users: **individuals** needing a personal, always-on, shareable storage space in the cloud.
- service providers: **data centers** supporting scientific research communities.



Users' view of their always-on personal file repository. Each user can remotely access her own files from her personal computers. The storage capacity of the repository can be adjusted to her individual needs.

3 DESIGN

Guiding principles:

- **API**: compatible with **Amazon S3**
- **storage backend**: on top of **key-value data stores**

Amazon S3 compatibility allows to use existing client-side tools, both GUI- and CLI-based.

We intend to exploit persistent, distributed data stores which:

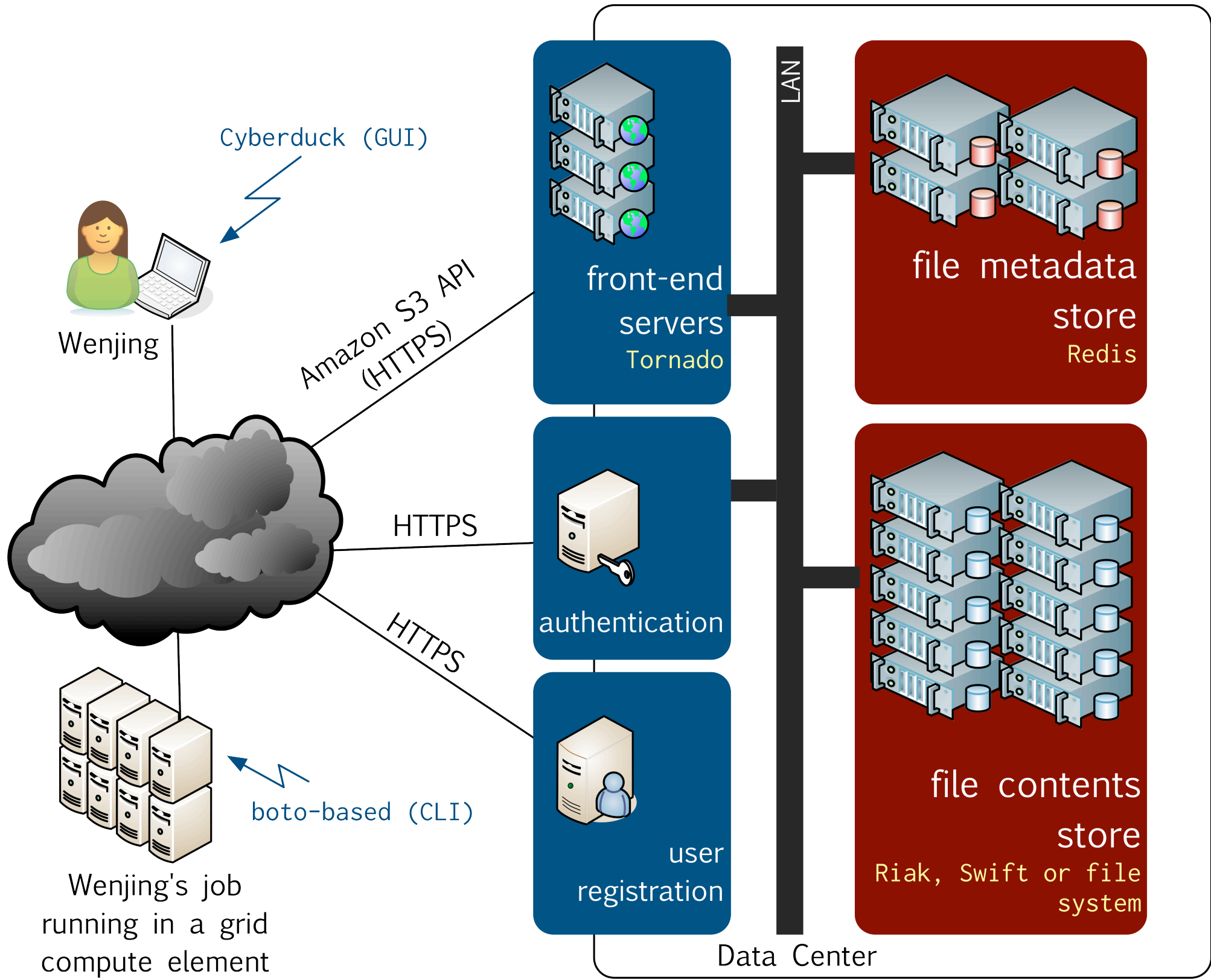
- can be built on top of **commodity storage and network hardware**
- **scale out** easily by adding storage servers
- provide configurable levels of redundancy to make **data available** even in presence of hardware failures or network partitions

4 BENEFITS

We aim to provide the following benefits:

for end-users	for service providers
redundancy makes your files highly available	redundant copies of files make backup copies unnecessary
organize your storage space to suit your individual needs	self-healing system help reduce operational costs
share your files with other individuals, registered users or otherwise	adjust storage capacity to accommodate demand
choose the more convenient GUI and CLI client-side software among the several available possibilities	use of commodity hardware help reduce equipment cost
your grid jobs can also use the repository on your behalf	exploit the existing X509 certificates-based identification infrastructure
use familiar tools and metaphors to interact with your repository (e.g. file browser, drag & drop)	exploit accounting records for billing purposes
	usage records allow for traceability

5 IMPLEMENTATION



High-level view of the components of the Mucura system. The metadata store and the file contents store are accessible to the users through the front-end S3-compatible servers. The software packages used for implementing each component of the system are shown.

Components:

- **client**: leverages existing Amazon S3-compatible GUI clients. Mucura-specific CLI for grid jobs — *proof-of-concept implemented*.
- **front-end server**: exposes a subset of Amazon S3 API — *prototyped*.
- **file metadata store**: stores file metadata in an in-memory key-value store — *prototyped*.
- **file contents store**: stores the contents of the files in a distributed key-value store or a networked file system — *prototyped on top of networked file system*.
- **authentication server**: delivers time-limited credentials for registered users from valid grid proxies — *proof-of-concept implemented*.
- **user registration server**: allows the end-users to get enrolled in the system — *proof-of-concept implemented*.

All components are developed in **Python**.

6 RELATED WORK

Several companies offer commercial cloud-based file storage services using proprietary software.

Although OpenStack's Swift and Mucura follow similar goals, for Mucura we use third-party data stores as the underlying platform for persisting both file contents and metadata.

7 PERSPECTIVES

We are prototyping alternative implementations of the file contents store on top of Riak and OpenStack Swift.

In addition to incrementally adding features to the existing core components, we aim at creating a **community** involving end-users, storage providers and software developers.

We plan to improve our web presence, use appropriate channels for software distribution, collect feedback from all interested parties and start an alpha-test period next summer.

8 CONCLUSIONS

Persistent distributed key-value data stores have properties that make them attractive to serve as backends of scalable file repositories.

The prototype system we implemented on top of them will help us understand their strengths and weaknesses in operating conditions.

9 CONTACT

mucura@ihep.ac.cn

Mucura: big clay container (sort of amphora) used for storing beverages, water, cereals and also for funeral rites by natives of pre-colombian ethnic groups in Colombia and other American countries

