Evolution of CERNBox

Sync&Share platform

Jakub T. Mościcki, CERN IT-ST

CS3 Amsterdam 2017
Short history of CERNBox

• service in active development (since 2014)

• integration and development of open-source components

• collaboration with the community (CS3, GEANT,…) and software developers (EOS,…) and companies (ownCloud)
Evolution

- Driver1: user demand and evolution of functionality
- Driver2: current and expected service growth
- Architecture evolution highlights
  - Synchronization & data flow
  - Sharing
  - Integration with other services
User demand

- Engineers
- Scientists
- Administration & Management
- Media Team & Onsite Services
• Office workflows
• Human-generated files, low traffic
• Ease-of-use & functionality
• Windows/Mac/Linux compatibility
• Specific application requirements

• For example, very large video files, ~30-50GB per file
- Engineers and general purpose “Unix” computing
- Lots of automatically generated files
- 2nd top user: accelerator studies, 2.5M simulation files
• Datasets, code repositories & general Unix use

• Top user: a physicist from a big experiment:

• 7M files (quota exceeded!), 2.3M LHC physics data files
proof of concept
architecture prototype
with LAMP stack
basic production service
EOS-based architecture
further scaling & new functionality
CERNBox Architecture Evolution
Highlights

• Synchronization
• Sharing
• New functionality & service integration
Synchronization
Synchronization & Data Flow

HTTP/Webdav

CERNBox Gateway

browser traffic

data / metadata

HTTP

Webdav

Webserver

direct access

namespace nodes

EOS

storage nodes

sync client

direct access

filesystem

xrootd
Sync client & protocol

• Users expect functionality & reliability of Dropbox and efficiency of local filesystems & local networks

• Protocol evolution: anticipate optimizations and advanced features, e.g.:
  • new chunking protocol from ownCloud, async uploads,…
  • event-based, git-based or binary protocols in the future / special use-cases, Infinite/Drive (?)

• Offline work: current protocol implementation is latency sensitive (with HTTP1.1)
Synchronization & Data Flow 2.0

- **Sync client**
- **CERNBox Gateway**
- **Webserver**
- **Sync protocol**
- **Browser traffic**
- **Data / metadata**
- **Native protocol**
- **Direct access**
- **Filesystem**
- **Xrootd**
- **Namespace nodes**
- **EOS**
- **Storage nodes**

**Sync service**
Sharing
Sharing in on-premise storage clouds

• Sharing made very easy

• A successful model

Users sharing files in CERBox (subset)
22K shares, 105 CERN groups, 320 External institutes
Sharing architecture

- REST API
- sync client
- CERNBox Gateway
- REST API
- Webserver
  - sharing app
    - ACLs
      - owner, target, ...
    - oc_share
  - MySQL
- direct access
  - filesystem
    - xrootd

- namespace nodes
- EOS
- storage nodes
Requires multiple sync folders
Sharing architecture 2.0

sync client → CERNBox Gateway → Webserver

REST API

namespace nodes
storage nodes

direct access
filesystem
xrootd

sync client

CERNBox Gateway

REST API

Webserver

Sharing architecture 2.0

REST API

Webserver

sharing μ service

ACLs
owner, target, ...

namespace nodes
storage nodes

EOS

direct access
filesystem
xrootd
Direct access to shared folders on storage
Challenge: scaling and infinite recursion

User *john* sharing a folder with user *kuba*

point of sharing/unsharing
Challenge: update propagation scaling for group sharing

User *john* sharing a folder with egroup *staff* of which user *kuba* is member
ACL management: basic sharing

point of sharing/unsharing
ACL management: challenges

Shared folder moved to private area
Competing strategies how to handle such cases

- local storage view via sync client
- filesystem users and expectations for `mv` semantics
ACL management: more corner cases for move
New functionality & service integration
Web Apps, Office and Collaborative Editing

CERNBox Gateway

browser traffic

Webserver

apps
picture gallery
file viewers

WOPI Client App

WOPI Client μ service

data / metadata

EOS

namespace nodes

storage nodes

Web Application Open Platform Interface Protocol
Web Apps and Scientific Computing

CERNBox Gateway

Data / metadata

Webserver

Browser traffic

Apps

Notebooks

Analysis files

Redirect

Direct access

Filesystem

SWAN Service
Federated Sharing, OCM

REST API
sync protocol

CERNBox Gateway

sync protocol

data / metadata

sync μ service

Webserver

native protocol

namespace nodes

remote access

ACLs

sharing μ service

direct access

OCM protocol
CERNBox

Open Source Storage

/home
~1PB
/eos/cms
/eos/atlas
/eos/project
~100 PB
Summary

- CERNBox is in full expansion and in very active development
- Challenged by the growing demands in terms of scaling and functionality
- Excellent collaboration with CS3 and GEANT community and software providers over last years!
  - JRC, AARNet, OwnCloud, GEANT and many others…

Looking towards further collaboration with the community on service integration, implementations & optimization, new ideas and protocol improvements