ownCloud

Infinite scale is a design principle

Our journey from using a database cluster to the spaces concept

Agenda

1 Database less storage with decomposedFS

2 Spaces as technical management units

3 Federated Storage

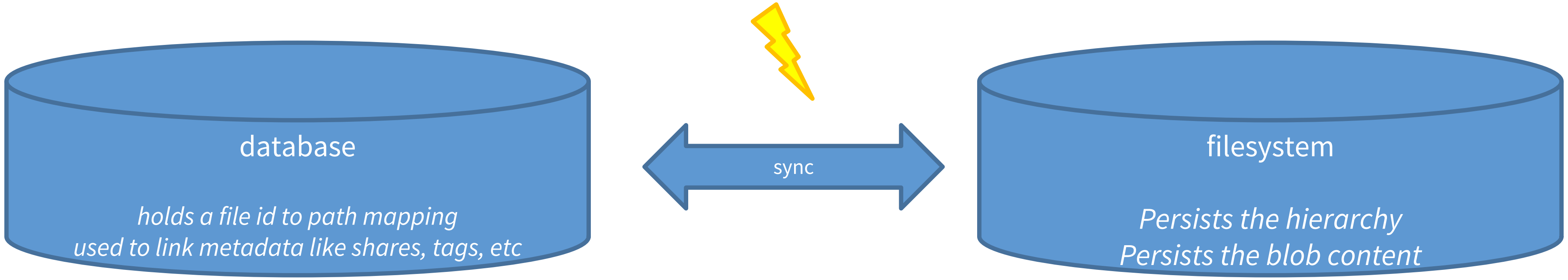
4 Next Steps

Jörn Dreyer
oCIS Lead Architect

Michael Barz
Product Owner &
Team Lead oCIS

Why does ownCloud 10 need a database?

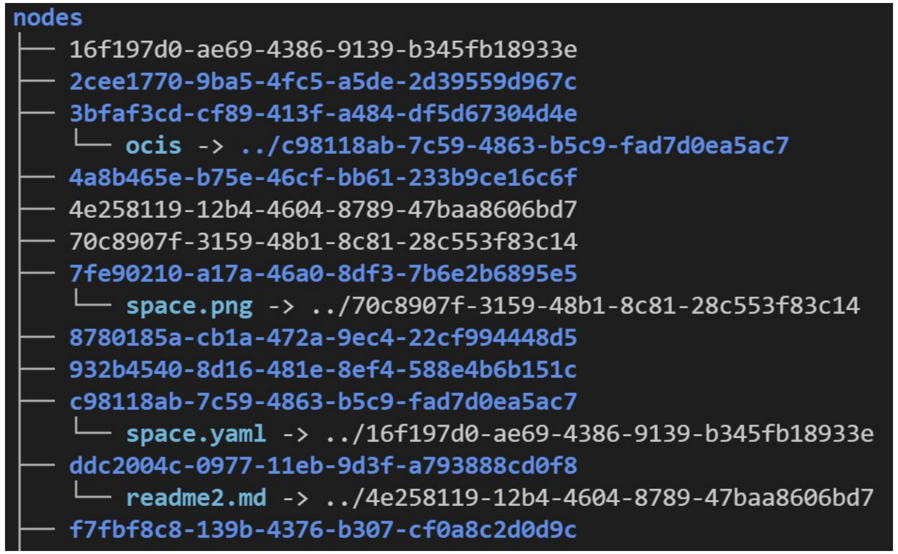
- ownCloud 10 needs to assign a stable ID to files to attach additional metadata for:
  - Share permissions, expiry, link tokens, etc..
  - Tags, Comments, and App data


database
holds a file id to path mapping
used to link metadata like shares, tags, etc
sync
filesystem
Persists the hierarchy
Persists the blob content

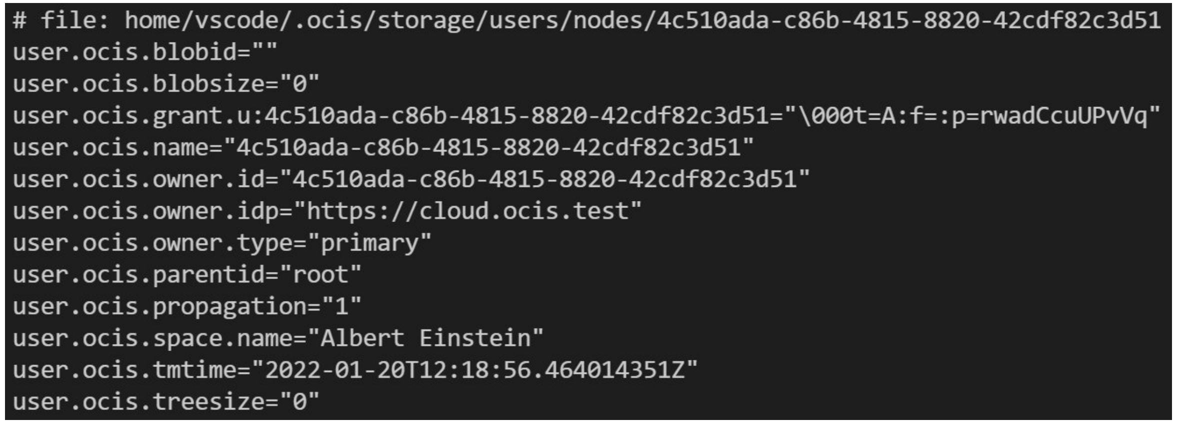
- Keeping database and filesystem in sync has been a cause of bugs and performance issues in the past

Can we put all metadata in the filesystem?

- Use the filesystem as a key/value store for id lookup and metadata storage



```
nodes
├── 16f197d0-ae69-4386-9139-b345fb18933e
├── 2cee1770-9ba5-4fc5-a5de-2d39559d967c
├── 3bfaf3cd-cf89-413f-a484-df5d67304d4e
│   └── ocis -> ../c98118ab-7c59-4863-b5c9-fad7d0ea5ac7
├── 4a8b465e-b75e-46cf-bb61-233b9ce16c6f
├── 4e258119-12b4-4604-8789-47baa8606bd7
├── 70c8907f-3159-48b1-8c81-28c553f83c14
├── 7fe90210-a17a-46a0-8df3-7b6e2b6895e5
│   └── space.png -> ../70c8907f-3159-48b1-8c81-28c553f83c14
├── 8780185a-cb1a-472a-9ec4-22cf994448d5
├── 932b4540-8d16-481e-8ef4-588e4b6b151c
├── c98118ab-7c59-4863-b5c9-fad7d0ea5ac7
│   └── space.yaml -> ../16f197d0-ae69-4386-9139-b345fb18933e
├── ddc2004c-0977-11eb-9d3f-a793888cd0f8
│   └── readme2.md -> ../4e258119-12b4-4604-8789-47baa8606bd7
├── f7fbf8c8-139b-4376-b307-cf0a8c2d0d9c
```

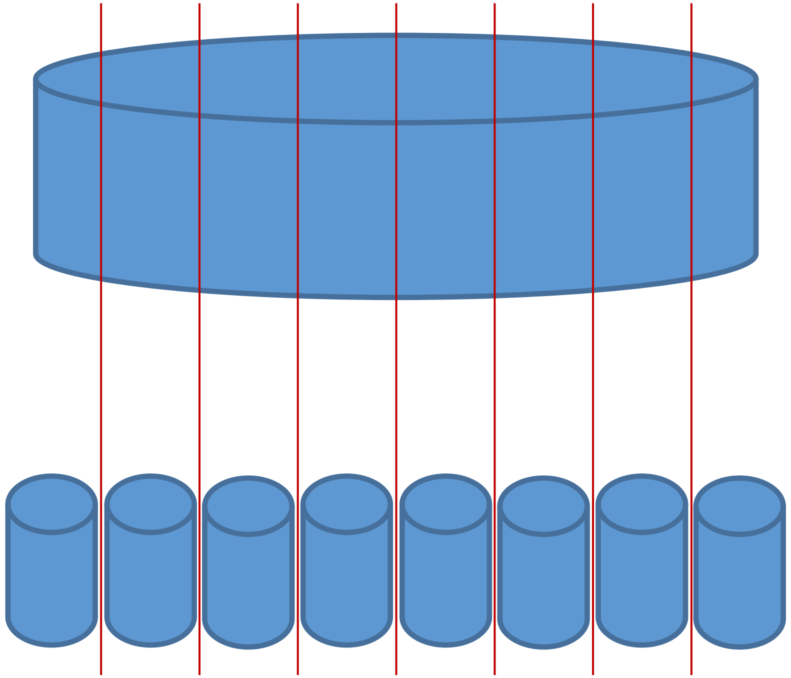


```
# file: home/vscode/.ocis/storage/users/nodes/4c510ada-c86b-4815-8820-42cdf82c3d51
user.ocis.blobid=""
user.ocis.blobsize="0"
user.ocis.grant.u:4c510ada-c86b-4815-8820-42cdf82c3d51="\000t=A:f=:p=rwadCcuUPvVq"
user.ocis.name="4c510ada-c86b-4815-8820-42cdf82c3d51"
user.ocis.owner.id="4c510ada-c86b-4815-8820-42cdf82c3d51"
user.ocis.owner.idp="https://cloud.ocis.test"
user.ocis.owner.type="primary"
user.ocis.parentid="root"
user.ocis.propagation="1"
user.ocis.space.name="Albert Einstein"
user.ocis.tmtime="2022-01-20T12:18:56.464014351Z"
user.ocis.treesize="0"
```

- Leverages the Linux Kernel vfs cache to look up a resource by id O(1)
- Tradeoff: full path lookup needs multiple stat calls (cached by kernel vfs)

Divide et Impera

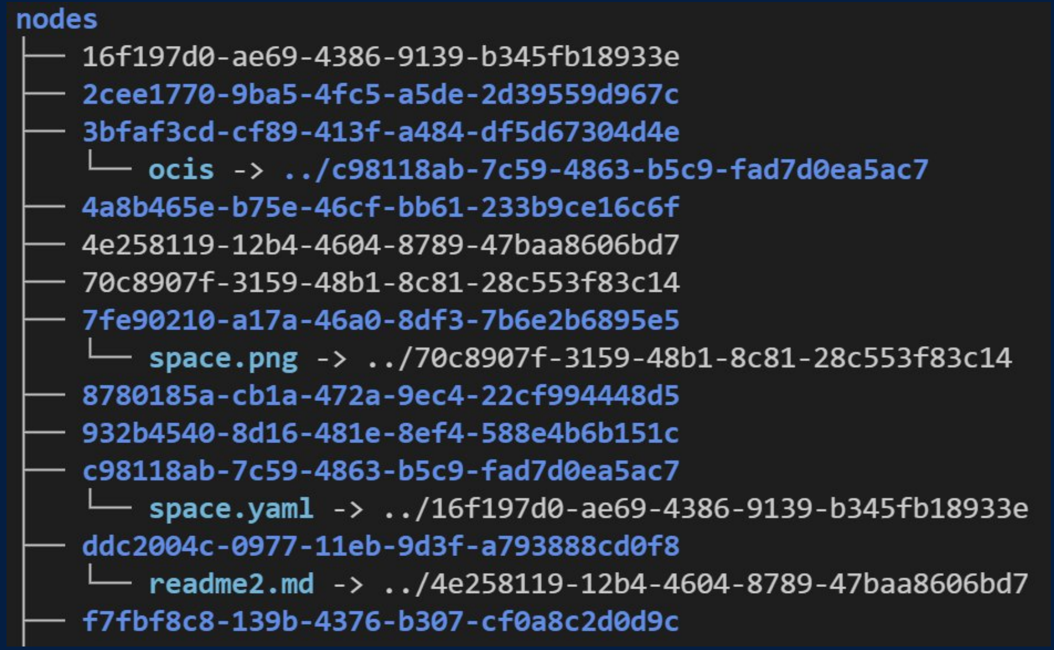
- A flat list of nodes creates overhead
  - Deleting a directory has to identify all children
  - Archiving a space has to identify all nodes that belong to a space, eg. when deprovisioning a user or project
- Shard nodes by space and make spaces the primary unit of management
- Storage providers identify resources by a root ResourceID and an optional relative path. Always!



every space has a dedicated nodes folder, a dedicated trash, persists metadata and can list CS3 grants and mountpoints

Database less storage with decomposed FS

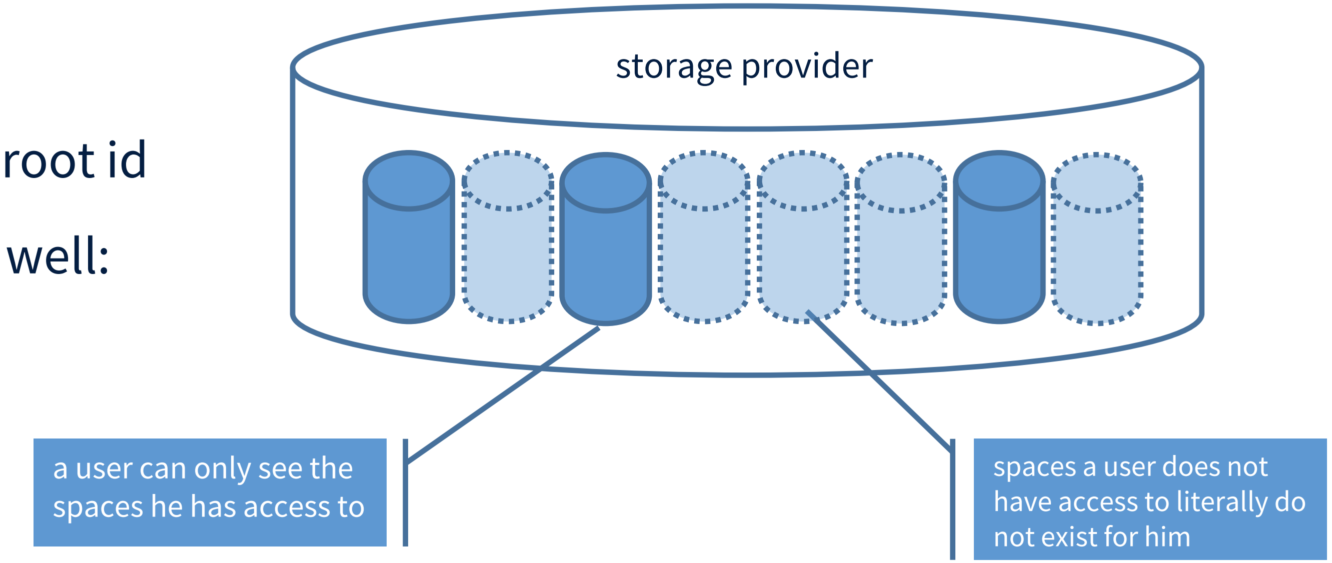
- Requires no RDBMS
- Uses a POSIX filesystem as a key/value store
- Atomic by mapping all write operations to either rename or symlink creation
- Prefer additional reads over expensive writes
- Metadata and blob storage can easily be put on true key/value stores like S3
  - Blobstore on S3 is implemented
  - Metadata in Redis, anyone?



```
nodes
├── 16f197d0-ae69-4386-9139-b345fb18933e
├── 2cee1770-9ba5-4fc5-a5de-2d39559d967c
├── 3bfaf3cd-cf89-413f-a484-df5d67304d4e
│   └── ocis -> ../c98118ab-7c59-4863-b5c9-fad7d0ea5ac7
├── 4a8b465e-b75e-46cf-bb61-233b9ce16c6f
├── 4e258119-12b4-4604-8789-47baa8606bd7
├── 70c8907f-3159-48b1-8c81-28c553f83c14
├── 7fe90210-a17a-46a0-8df3-7b6e2b6895e5
│   └── space.png -> ../70c8907f-3159-48b1-8c81-28c553f83c14
├── 8780185a-cb1a-472a-9ec4-22cf994448d5
├── 932b4540-8d16-481e-8ef4-588e4b6b151c
├── c98118ab-7c59-4863-b5c9-fad7d0ea5ac7
│   └── space.yaml -> ../16f197d0-ae69-4386-9139-b345fb18933e
├── ddc2004c-0977-11eb-9d3f-a793888cd0f8
│   └── readme2.md -> ../4e258119-12b4-4604-8789-47baa8606bd7
├── f7fbf8c8-139b-4376-b307-cf0a8c2d0d9c
```

Storage providers become storage space providers

- Storage providers learn to recognize spaces
- A mandatory rootID in every request allows the storage provider to identify the correct space
- A root id may be any node in a space
- An optional path is always relative to the root id
- All other space properties are sharded as well:
  - Arbitrary metadata and policies
  - Grants and mountpoints,
  - Trash, lifecycle and workflows
  - Search indexes



a user can only see the spaces he has access to

spaces a user does not have access to literally do not exist for him

The reva storage registry becomes a storage space registry

- Active spaces discovery
- Metadata aggregation and caching
- A “bookmarking” service for spaces

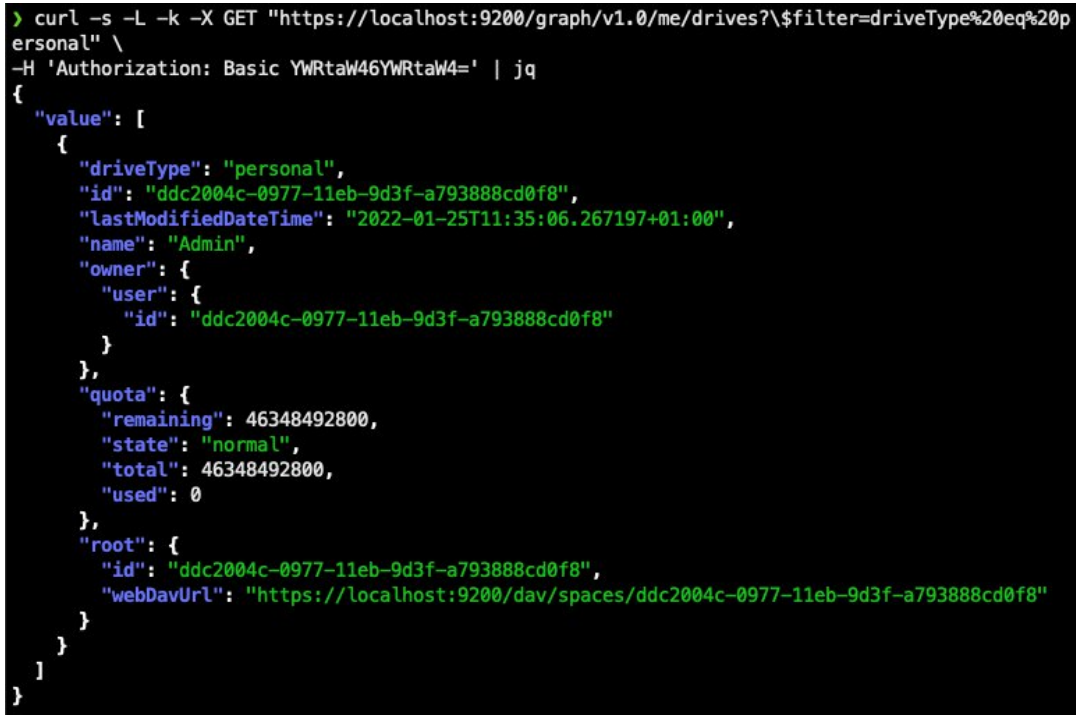
userID	spaceID	mount path	mtime	address
einstein	90891c44	/users/einstein	1643126310	127.0.0.1:9157
marie	7cadd5d6	/users/marie	1643143253	127.0.0.1:9157
marie	ec597ff5	/projects/moon	1643128601	cernbox.cern.ch:9161
richard	ec597ff5	/projects/moon	1643128601	cernbox.cern.ch:9161
richard	b5c2d969	/users/richard	1643114320	127.0.0.1:9157

a user can change the mount path to his liking, constrained by instance rules

Introduces indirection!

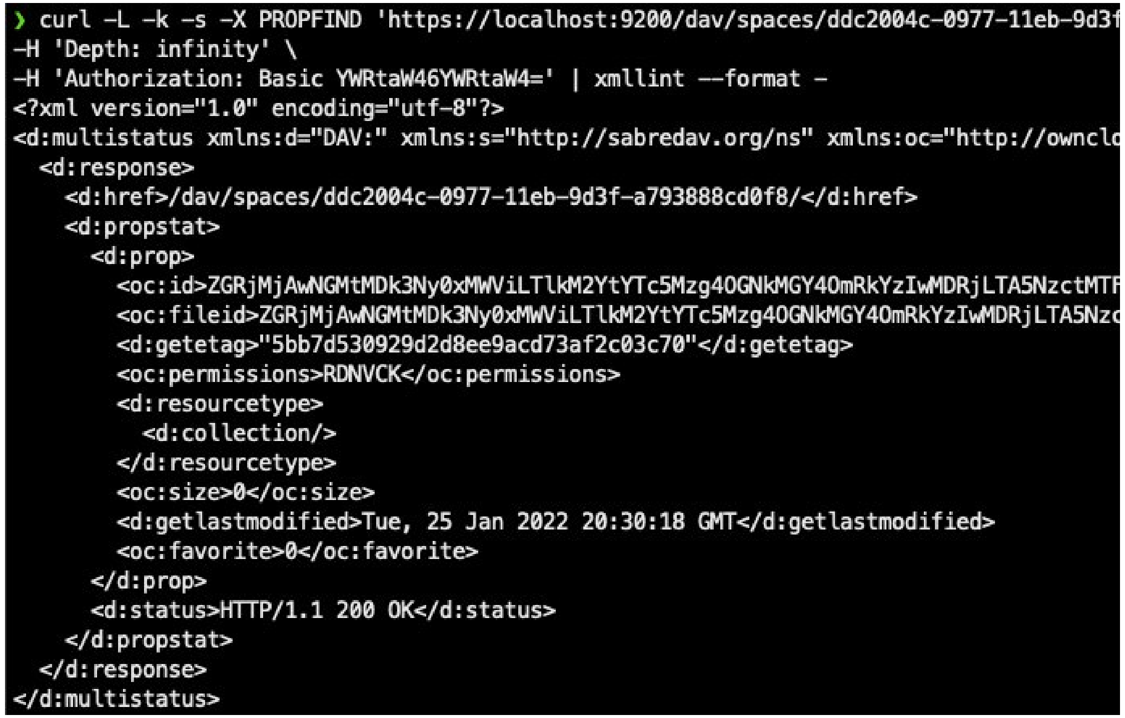
Example space discovery

- Discover spaces using LibreGraph /me/drives
- Translates into CS3 gateway.ListStorageSpaces



```
❯ curl -s -L -k -X GET "https://localhost:9200/graph/v1.0/me/drives?\$filter=driveType%20eq%20p
ersonal" \
-H 'Authorization: Basic YWRtaW46YWRtaW4=' | jq
{
  "value": [
    {
      "driveType": "personal",
      "id": "ddc2004c-0977-11eb-9d3f-a793888cd0f8",
      "lastModifiedDateTime": "2022-01-25T11:35:06.267197+01:00",
      "name": "Admin",
      "owner": {
        "user": {
          "id": "ddc2004c-0977-11eb-9d3f-a793888cd0f8"
        }
      },
      "quota": {
        "remaining": 46348492800,
        "state": "normal",
        "total": 46348492800,
        "used": 0
      },
      "root": {
        "id": "ddc2004c-0977-11eb-9d3f-a793888cd0f8",
        "webDavUrl": "https://localhost:9200/dav/spaces/ddc2004c-0977-11eb-9d3f-a793888cd0f8"
      }
    }
  ]
}
```

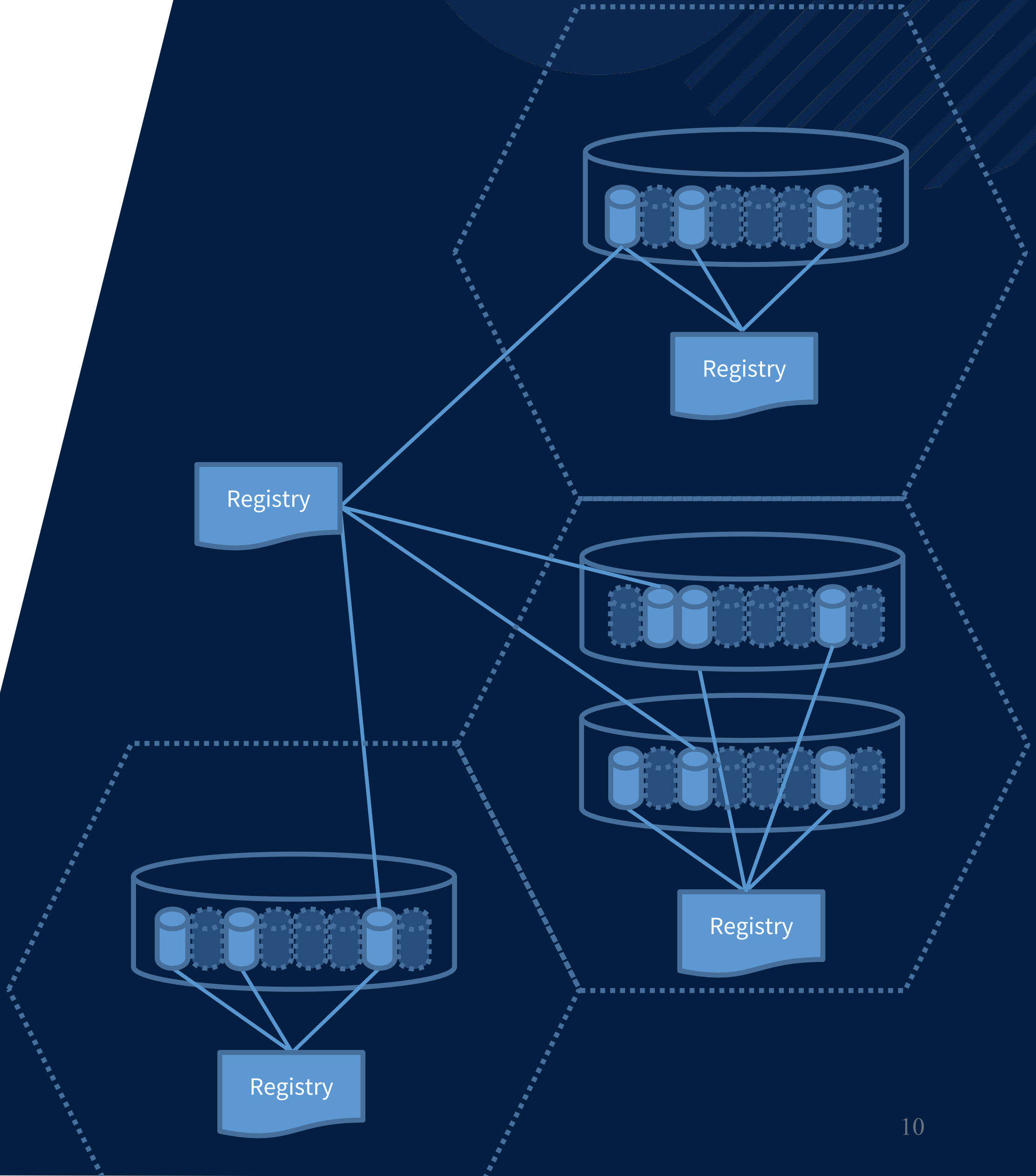
- Navigate space using WebDAV /dav/spaces
- Translates into CS3 gateway.ListContainer / Stat



```
❯ curl -L -k -s -X PROPFIND 'https://localhost:9200/dav/spaces/ddc2004c-0977-11eb-9d3f
-H 'Depth: infinity' \
-H 'Authorization: Basic YWRtaW46YWRtaW4=' | xmllint --format -
<?xml version="1.0" encoding="utf-8"?>
<d:multistatus xmlns:d="DAV:" xmlns:s="http://sabredav.org/ns" xmlns:oc="http://ownclo
  <d:response>
    <d:href>/dav/spaces/ddc2004c-0977-11eb-9d3f-a793888cd0f8/</d:href>
    <d:propstat>
      <d:prop>
        <oc:id>ZGRjMjAwNGMtMDk3Ny0xMWViLTlkM2YtYTc5Mzg4OGNkMGY4OmRkYzIwMDRjLTA5NzctMTF
        <oc:fileid>ZGRjMjAwNGMtMDk3Ny0xMWViLTlkM2YtYTc5Mzg4OGNkMGY4OmRkYzIwMDRjLTA5Nzc
        <d:getetag>"5bb7d530929d2d8ee9acd73af2c03c70"</d:getetag>
        <oc:permissions>RDNVCK</oc:permissions>
        <d:resourcetype>
          <d:collection/>
        </d:resourcetype>
        <oc:size>0</oc:size>
        <d:getlastmodified>Tue, 25 Jan 2022 20:30:18 GMT</d:getlastmodified>
        <oc:favorite>0</oc:favorite>
      </d:prop>
      <d:status>HTTP/1.1 200 OK</d:status>
    </d:propstat>
  </d:response>
</d:multistatus>
```

Federated Storage

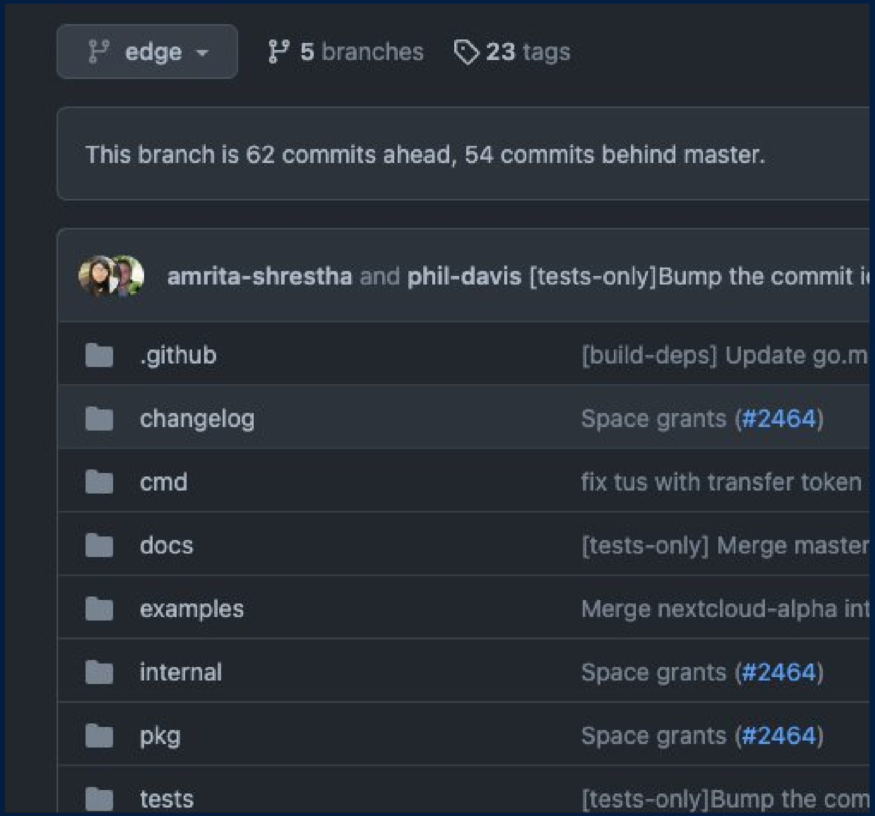
- Spaces as primary unit of management
  - Capture the storage lifecycle of users, groups, projects and eg. apps
  - Allow choosing different tradeoffs for workflows, geo-replication and more
- Registries and spaces do not have to be hosted on the same domain
- OpenID Connect allows spaces and registries to work as a federated storage



We started the edge branch

- Already implemented the spaces architecture
- oCIS master is running on “edge” already
- The Gateway has been simplified dramatically and could be replaced by a middleware

  thanks Hugo! :-)
- The WebDAV service has adapted
  - Former urls still work
  - New endpoint was added directly access the space by ID



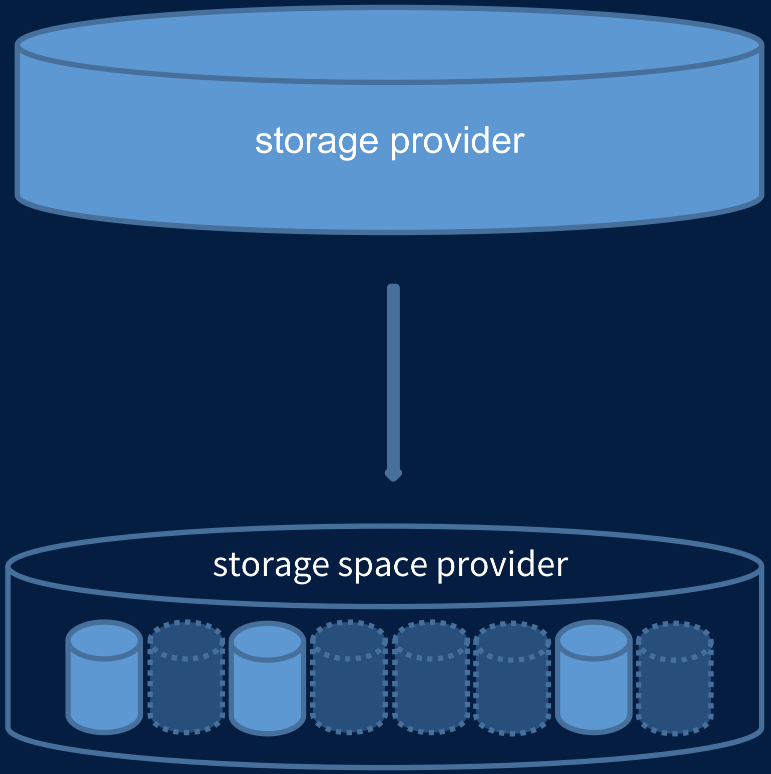
This branch is 62 commits ahead, 54 commits behind master.

amrita-shrestha and phil-davis [tests-only]Bump the commit i

.github	[build-deps] Update go.m
changelog	Space grants (#2464)
cmd	fix tus with transfer token
docs	[tests-only] Merge master
examples	Merge nextcloud-alpha int
internal	Space grants (#2464)
pkg	Space grants (#2464)
tests	[tests-only]Bump the com

Transform existing implementations

- Storage drivers need to become aware of spaces
- Actually simplified the implementation as it makes the home enabled flag unnecessary
- All other services which access resources via the CS3 APIs need to reference them by spaceID and relative path (e.g WOPIServer)
- Clients can discover the spaceID for a path using the storage space registry
- Clients are empowered to build an optimal namespace for the user



Proving our concept

- oCIS is running on it with full test coverage (E2E-, API-, and Integration tests) using the decomposedFS
- CERN and ownCloud are collaborating on bringing an EOS testsystem into the CI to prove this approach also for CERNBox *what about CephFS?*
- We created the cs3api-validator as a standalone litmus testing tool
- Our K6 based performance benchmarks are monitoring the changes nightly
- We are looking forward to get edge merged into master

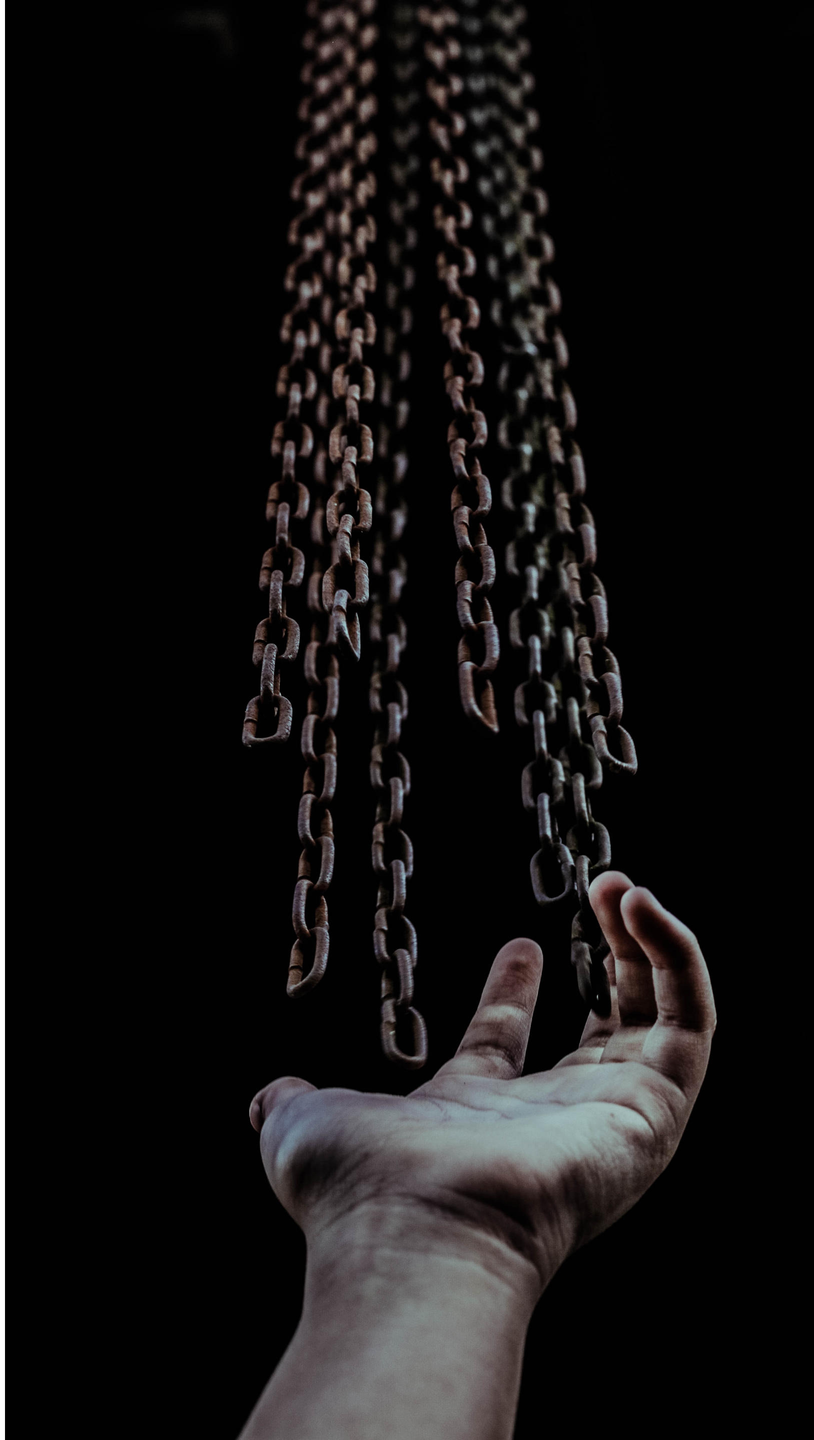
Thank you!

Links

- Libre Graph Api https://github.com/owncloud/libre-graph-api
- Edge branch https://github.com/cs3org/reva/tree/edge
- Spaces Registry https://owncloud.dev/extensions/storage/spacesregistry/
- K6 Benchmark Tests https://github.com/owncloud/cdperf
- cs3api-validator https://github.com/owncloud/cs3api-validator

Dr. Jörn Friedrich Dreyer
jfd@owncloud.com, GitHub: @butonic

Michael Barz
mbarz@owncloud.com, GitHub: @micbar



ownCloud