

Scaling ownCloud beyond today's filecache

Dr. Jörn Dreyer
Solutions Architect
ownCloud GmbH
tw/gh: @butonic



Customer feedback:
“oc_filecache table is a
bottleneck”?

Working Theory:
→ tree implemented as a
naive data model
→ etag propagation triggers
many updates



Replace filecache
with direct
filesystem
access?



Sharding
ownCloud
instances?

Rawstorage - Challenges

C1: oc_filecache actually is an index

- shares, comments and tags are tied to file ids
- scanning is a poor sync from file system to db

C2: file systems do not propagate etags

- neither mtime
- WebDAV RFC requires atomic PUT

Rawstorage - Approach

Idea: use posix filesystem as metadata storage

- fileid = inode
- etag as extended attribute, propagated by oc

Limitations - OK for testing approach

- inodes are assumed to be unique
 - may collide with inodes on other mount points
 - beware: inodes are reused by the filesystem
- requires xattr PHP extension

Rawstorage - Current State

Works as designed

- requires caching lookup of path by fileid / inode

App being tested by IBM

- GPFS / Spectrum Scale can use policy engine or light weight events (LWE) to propagate etags
 - external changes become instantly visible to sync clients
 - if the propagation is atomic it can move from oc to the fs
 - changes the original use case
- preparing performance comparison at IBM lab

Federated cluster - Challenges

C1: db in general may become a bottleneck

- too many users, files, activities...
- horizontal vs. vertical scaling

C2: federation is not seamless

- URLs between instances are different
- Federation is exposed in the sharing UI

C3: migration path

- user by user migration
- test and compare new versions

Federated cluster - Approach

Use haproxy to hide several federated instances

- login sets a cookie that sends the next request to the right instance
- each node is an independent ownCloud instance
- an app allows interaction between the instances to make login and inter node sharing seamless

Limitations OK for PoC

- sharing to groups is not used

Federated cluster - Current State

Work in progress

- login and redirection to correct instance is done
- sharing requires patches to core
 - allow altering the share object in hooks
 - dedicated sharing implementation
 - link tokens must be unique for the cluster

Working with a customer on a PoC

Do the MathFuture Work

Rawstorage: move even more to file system?

- map shares to ACLs?
- tags and comments in extended attributes?

Federated Rawstorage cluster?

- Combination would allow seamless user by user migration without having to move any data
- inodes are even unique over all nodes

Send your use case and your ideas to

Questions?



Image Licenses

- [Giuseppe Milo - Sunrise in Madeira Beach](#) [CC-BY-NC 2.0](#)
- All others CC0

Dr. Jörn Dreyer
Solutions Architect
ownCloud GmbH
tw/gh: @butonic