Likes System on Disk

From Implementation to Optimization: A Journey in Persistence

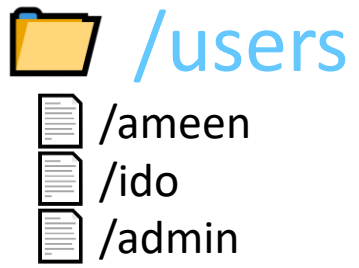
1. The Foundation: User Implementation

The Problem

- The system had no concept of 'Users'.
- Cannot implement 'Likes' without an actor.

The Solution

- Created a 'users' directory.
- File-based approach: A file named 'Ameen' means user Ameen exists.
- Simple existence check (O(1)).



2. Methodology & Concurrency

Strict Development Workflow

- 1. Implement Core Logic (in Module).
- 2. Write Tests (Ensure logic is sound).
- 3. Expose to CLI (User Interface).
- Advice: This order prevents debugging UI and Logic simultaneously.

Concurrency Management

- Learned: Race conditions occur when multiple users write at once.
- Mechanism: File Locking (fcntl.flock).
- Strategy: Lock the specific resource (file/journal) -> Write -> Unlock.

3. Architecture: Dual-Side Storage

❌ Rejected: Branch per User

Idea: Likes are a linked chain of commit-like objects per user.
Pros: Scales infinitely (no directory limits).
Cons: Slow Read (O(N)). To count a user's likes, we must traverse their entire chain.

✅ Chosen: User-Side + Commit-Side

Idea: One file per like in a directory.
Pros: Fast Read (approx O(1)). We can ask the FS 'does this file exist?' instantly.
Cons: File System limits (max files per directory).

4. Scaling: The Filesystem Bottleneck

The Trade-off

- Filesystems (like ext4) limit the number of files in a single directory.
- A popular commit with 1M likes = 1M files = Directory Bloat.

The Discussion

- Considered switching to XFS (Extended File System) which uses B-trees for unlimited files.
- Verdict: Directory still becomes heavy/slow to manage.

The Solution: Sharding (Bucketing)

- Distributed users into buckets based on the first 2 Hex characters of their ID.
- Result: 256 sub-directories.
- Prevents any single directory from hitting filesystem limits.

5. Integrity: Source of Truth & Recovery

👑 Source Of Truth (SOT)

Since we duplicate data (User Side & Commit Side), misalignment is possible.
Rule: If sides conflict, the User Side is trusted.

📓 Journal (Write-Ahead Log)

Protects against crashes during the dual-write.
Flow: Log Op → Execute → Delete Log.

Crash Recovery Flow:

1. System Boot.

2. Read Journal.

3. Found entry?
   -> It means crash happened before finish.

4. Re-run operation.

5. Clean Journal.

Conclusion

- Design Choices: Prioritized read-speed (O(1)) and data safety over storage space.
- Scalability: Solved filesystem limits via Sharding/Bucketing.
- Reliability: Journaling ensures the system survives crashes.
- Growth: Learned the value of modular code and disciplined testing workflows.

Outputs

```
(venv) root@8a68b8bc8135:/workspace# caf users
Users:
somaya
hmode
(venv) root@8a68b8bc8135:/workspace# caf user_likes hmode
1 Likes found:
53ee03a9c508909548db8e5454b0b113631b65f7
(venv) root@8a68b8bc8135:/workspace# caf commit_likes 53ee03a9c508909548db8e5454b0b113631b65f7
2 Likes found:
hmode
somaya
(venv) root@8a68b8bc8135:/workspace# caf user_likes somaya
1 Likes found:
53ee03a9c508909548db8e5454b0b113631b65f7
(venv) root@8a68b8bc8135:/workspace# caf commit ameen HelloWorld
Commit created successfully:
Hash: 0840f27979390cc4f934a813f19e8572cb1113ea
Author: ameen
Message: HelloWorld

(venv) root@8a68b8bc8135:/workspace# caf log
Commit history:

Commit: 0840f27979390cc4f934a813f19e8572cb1113ea
Author: ameen
Date: 2026-01-31 00:16:39

    HelloWorld

--------------------------------------------------

Commit: 53ee03a9c508909548db8e5454b0b113631b65f7
Author: ameen
Date: 2026-01-28 23:27:36

    goku

--------------------------------------------------

(venv) root@8a68b8bc8135:/workspace# caf like somaya  0840f27979390cc4f934a813f19e8572cb1113ea
User "somaya" liked commit "0840f27979390cc4f934a813f19e8572cb1113ea".
(venv) root@8a68b8bc8135:/workspace# caf user_likes somaya
2 Likes found:
0840f27979390cc4f934a813f19e8572cb1113ea
53ee03a9c508909548db8e5454b0b113631b65f7
(venv) root@8a68b8bc8135:/workspace# caf unlike somaya  0840f27979390cc4f934a813f19e8572cb1113ea
User "somaya" unliked commit "0840f27979390cc4f934a813f19e8572cb1113ea".
(venv) root@8a68b8bc8135:/workspace# caf commit_likes  0840f27979390cc4f934a813f19e8572cb1113ea
0 Likes found.
(venv) root@8a68b8bc8135:/workspace#
```

Questions