



OS X Trash Freedom

2010

1 min read • 249 words

UX Philosophy: This hack represents a fundamental tension in user interface design: the balance between safety (trash as a safety net) and efficiency (direct deletion). The 120GiB accumulation demonstrates how "helpful" features can become counterproductive without active management.

I noticed today that i had 120GiB of data in my Mac's Trashcan. I had enough. so I tried to kill it, and discovered a nice hidden feature.

```
rm -fr ~/.Trashln -s /dev/null ~/.Trash
```

Technical Insight: This command removes the trash directory and replaces it with a symbolic link to `/dev/null`, effectively making the trash "black hole" any deleted files. The genius lies in leveraging macOS's fallback behavior when the trash is unavailable.

Now, when you delete a file in Finder, you get a nice pop-up warning that the file will be permanently deleted.

Bliss.

Minimalist Computing: This single word encapsulates the satisfaction of removing friction from daily computing. The hack represents the broader Unix philosophy of "do one thing well" applied to desktop interaction.

Update (September 25th, 2010): As it turns out, OS X defaults to this behavior whenever ~/.Trash is unwritable as a folder. If you were to place any file in ~/.Trash, you would see the same behaviour. Oh well, it's still rather useful in function.

System Behavior Discovery: This update reveals the accidental nature of the discovery, demonstrating how exploring edge cases in system behavior can uncover useful features. The humble acknowledgment ("Oh well") shows intellectual honesty in recognizing when a "hack" is actually just documented system behavior.