Verifying the FAT32 filesystem in ACL2

Mihir Mehta

Department of Computer Science
University of Texas at Austin

mihir@cs.utexas.edu

06 April, 2018

Outline

Motivation and related work

Our approach

Outline

Why we need a verified filesystem

- Filesystems are everywhere, even as operating systems move towards making them invisible.
- In the absence of a clear specification of filesystems, users (and sysadmins in particular) are underserved.
- Modern filesystems have become increasingly complex, and so have the tools to analyse and recover data from them.
- It would be worthwhile to specify and formally verify, in the ACL2 theorem prover, the guarantees claimed by filesystems and tools.

Related work

- In Haogang Chen's 2016 dissertation, the author uses Coq to build a filesystem (named FSCQ) which is proven safe against crashes in a new logical framework named Crash Hoare Logic. His (exported) Haskell implementation performs comparably to ext4.
- Hyperkernel (Nelson et al., SOSP '17) is a "push-button" verification effort, but approximates by changing POSIX system calls for ease of verification.
- In our work, we instead aim to model an existing filesystem (FAT32) faithfully and match the resulting disk image byte-to-byte.

Outline