

Very Very Simple File System (VVSFS)

Jack Kilrain (u6940136) Daniel Herald (u7480080) Angus Atkinson (u7117106)

22 October 2023

Contents

0.1	Overview	1
0.2	Tasks Completed	1
0.3	Testing	1
0.4	Baseline	2
0.4.1	Unlink Dentries and Removing Directories	2
0.4.2	Renaming	2
0.4.3	Inode Attributes	2
0.4.4	Supporting FS Stats	2
0.5	Advanced	2
0.5.1	Indirect Blocks	2
0.6	Extensions	2
0.6.1	Hardlinks and Symbolic Links	2
0.6.2	Special Devices	2

0.1 Overview

TODO

0.2 Tasks Completed

TODO

0.3 Testing

- We created our own test suit (vvsfs/vvsfs_tests)
 - This was used to drive test driven design as we could create tests for expected behaviour and build new features.
 - Additionally we utilised it as a regression test suit to ensure that new code didn't break anything. And whenever we fixed problems that were discovered we built a test to ensure we didn't break it again.
 - It is composed of a set of helper scripts that provide automatic generation of a test environment, and an assertion framework to provide nice error messages.
- We used the pjdftest to check our implentation for posix compliance and various other edge cases. By the end we passed all tests with the following exceptions:
 1. The tests for large files (2gb) files.
 2. The filesystem does not keep track of the . & .. files in directories as such we failed the test testing that folder link counts were incremented correctly. We chose to ignore this due to <https://edstem.org/au/courses/12685/discussion/1633469>.

3. The filesystem does not correctly update ctime on truncate. (TODO: Does anyone want to fix this?)
4. The filesystem does not store high precision time, only seconds like minix & ext2. (TODO: Does anyone want to fix this?)

0.4 Baseline

0.4.1 Unlink Dentries and Removing Directories

TODO

0.4.2 Renaming

TODO

0.4.3 Inode Attributes

TODO

0.4.4 Supporting FS Stats

TODO

0.5 Advanced

0.5.1 Indirect Blocks

TODO

0.6 Extensions

0.6.1 Hardlinks and Symbolic Links

TODO

0.6.2 Special Devices

TODO