

中期答辩 - 基于数据库的文件系统设计与实现

陈林峰

北京理工大学

2023.4.6

时间安排

2022.12-2023.1 : 参与选题并完成开题报告

2023.1-2023.2 : 分析数据库实现，将数据库移植到 riscv 裸机平台，设计文件系统实现方式，完成文件系统的基本功能。

- 修改数据库接口，在不影响其原有功能的基础上可以自定义底层文件依赖
- 在qemu模拟的riscv平台上实现定义的接口，使得jammdb可以运行与裸机上
- 根据数据库的数据结构或者自定义数据结构完成层级文件系统的抽象和实现
- 实现POSIX定义的大部分文件系统调用接口

时间安排

2023.2-203.3: 完成文件系统的高级功能，优化文件系统的实现，提高文件系统性能

- 针对文件内容存储方式，优化文件读写性能
- 导出数据库接口，使得用户在同一块存储设备上既可以使用数据库也可以使用文件系统
- 针对数据库的key-value带来的便利，实现用户对文件元数据的扩展

时间安排

2023.3-2023.4: 将基于数据库的文件系统实现与其它文件系统进行对比分析，总结使用数据库系统构建文件系统带来的优势和不足之处。

- 完成jammdb的正确性测试
- 编写测例完成对文件系统实现的正确性测试
- 编写测例测试不同文件系统的性能，分析原因
- 编写测例使用DBFS的扩展功能，对比在传统文件中实现相关功能带来的代价

2023.4-2023.5: 完成论文撰写，完成课题报告

2023.5-2023.6: 参与论文答辩

项目组成

我们选择使rust进行项目开发，因此需要有一个rust编写的os(Alien) 来支持，同时，为了与其它文件系统比较，我们需要开发一个VFS框架来组织不同文件系统，因此整个dbfs项目由几个独立的子项目构成。

1. Alien: 一个使用rust实现的模块化的操作系统内核
2. rvfs: rust实现的vfs框架
3. fat32-vfs: fat32的vfs接口层
4. jammdb: 数据库的no_std移植
5. dbfs: 数据库文件系统的vfs接口层

项目完成度

- Alien OS可以运行用户态程序，并支持vfs和两个底层文件系统
- jammdb移植完成，可以做到不依赖操作系统
- dbfs按照设计文档完成实现
- fat32-vfs使用rust社区中下载量最多的库进行实现更有说服力
- vfs参照linux实现，实现大部分功能

上述项目作为可重用模块存在。

测试

- 简单程序功能测试
- 读写性能测试: 需要找到一些比较复杂说服力较强的测试程序
- 扩展功能测试

TODO

1. 参与比赛，做更完善的系统来测试，并且在真板子上测试
2. 使用rust for linux