



File System Design And Implementation For Virtual Operating Systems

Taurin Tan-atichat
Computer Science & Engineering
Advisor: Joe Pasquale

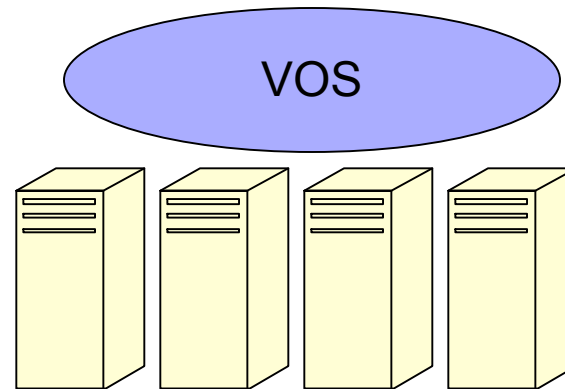
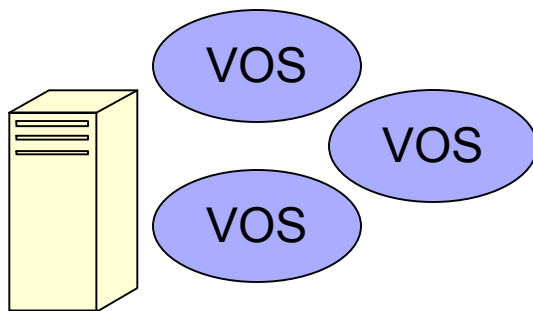


Outline

- Motivation
- Background
- Design
- Implementation
- Optimization

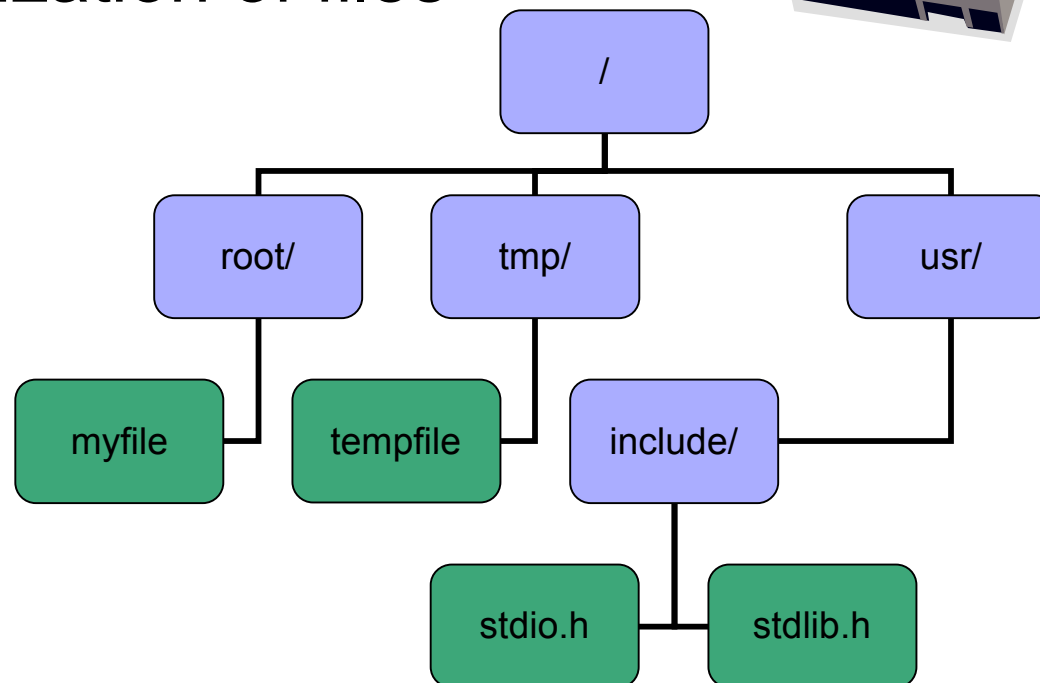
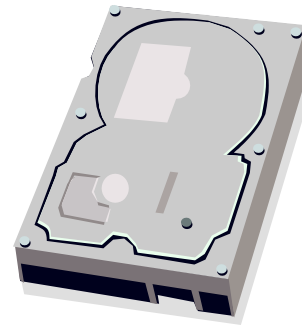
Motivation – Virtual Operating System

- Increasing number of computing devices with different architectures
- Ability to add, modify, and remove kernel features without administrative privileges
- Capable of running multiple instances on one physical machine or one instance over a cluster of machines



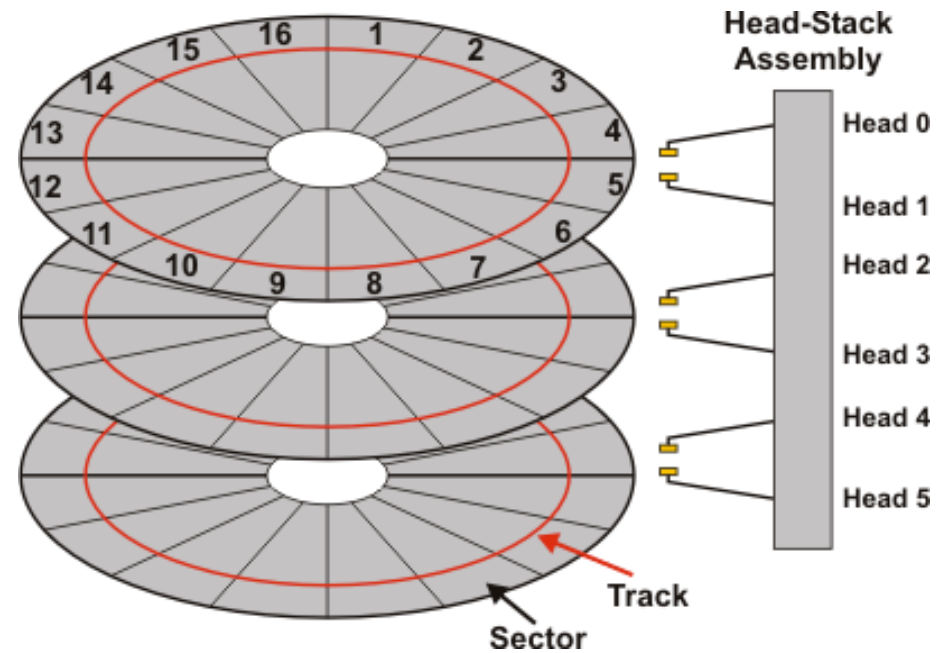
Motivation – File System

- Permanent storage
- Virtual memory
- Organization of files



Background – Physical Model

- Seek time
- Rotational Latency






Background – Logical Model

- Array of blocks
- Blocks are fixed sized
 - Typically 1024 bytes
- Operations
 - Read block
 - Write block

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15
16	17	18	19
20	21	22	23
24	25	26	27
28	29	30	31



Design - Goals

- Intuitive user interface
- Scalable
 - Hierarchical directory structure
 - Create, update, move, remove files
- Efficient
- Minimum restrictions (name, size of files)
- Secure



Design - Organization

- File System Metadata

- Information about the entire file system

- File Metadata

- Information about a specific file

- Datablock

- Actual data

FSM	FSM	FSM	FSM
FM	FM	FM	FM
FM	FM	FM	FM
DB	DB	DB	DB
DB	DB	DB	DB
DB	DB	DB	DB
DB	DB	DB	DB
DB	DB	DB	DB

Implementation – File System Metadata (superblock)

■ Sizes

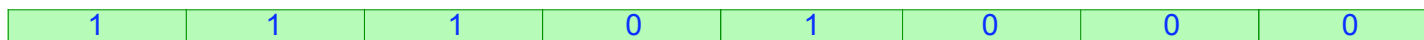
- Entire file system
- Block

■ Free blocks for File Metadata and Datablocks

- Bitmaps

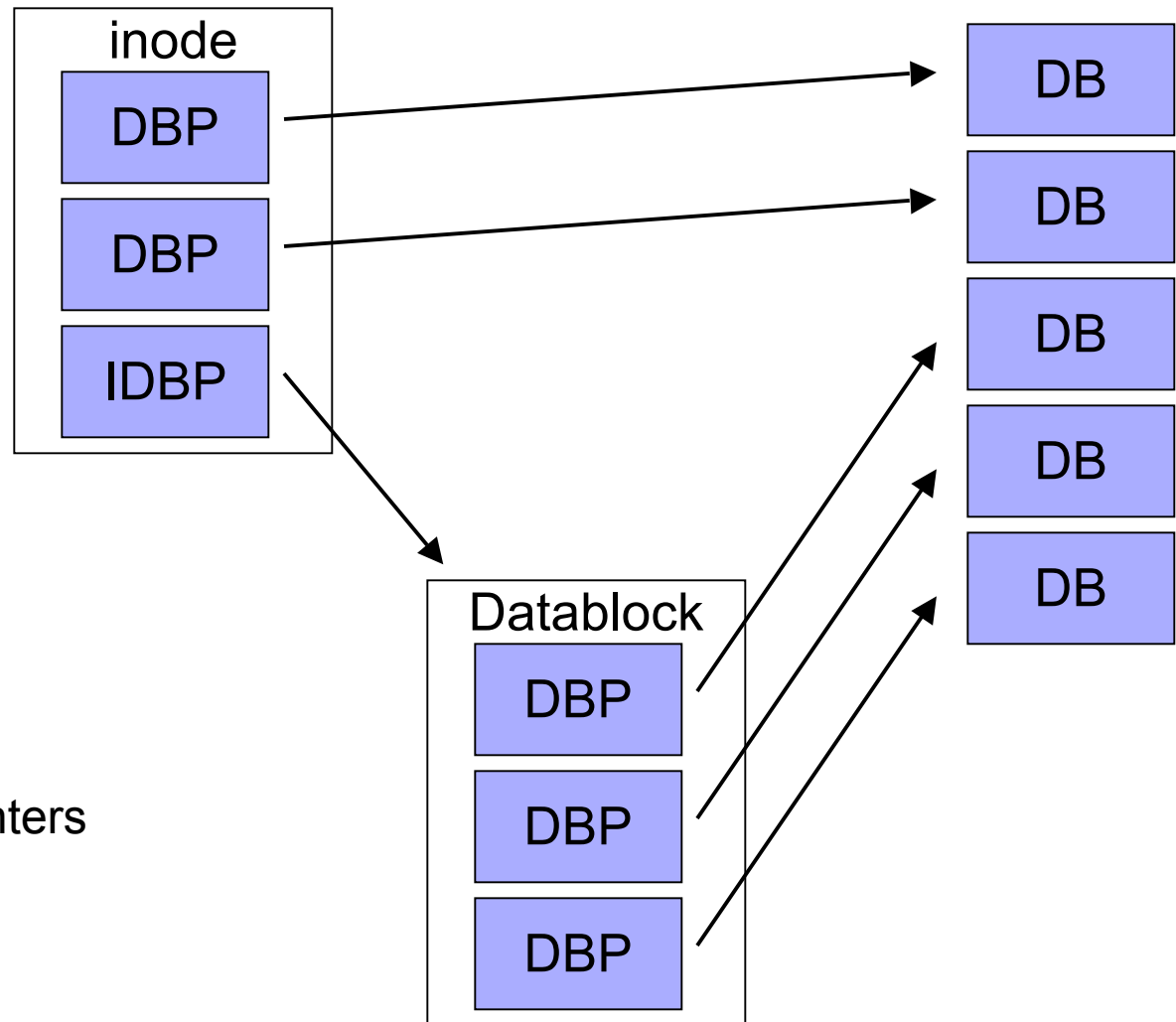
■ Locations

- File Metadata bitmap
- Datablock bitmap
- File Metadata root
- Datablock start



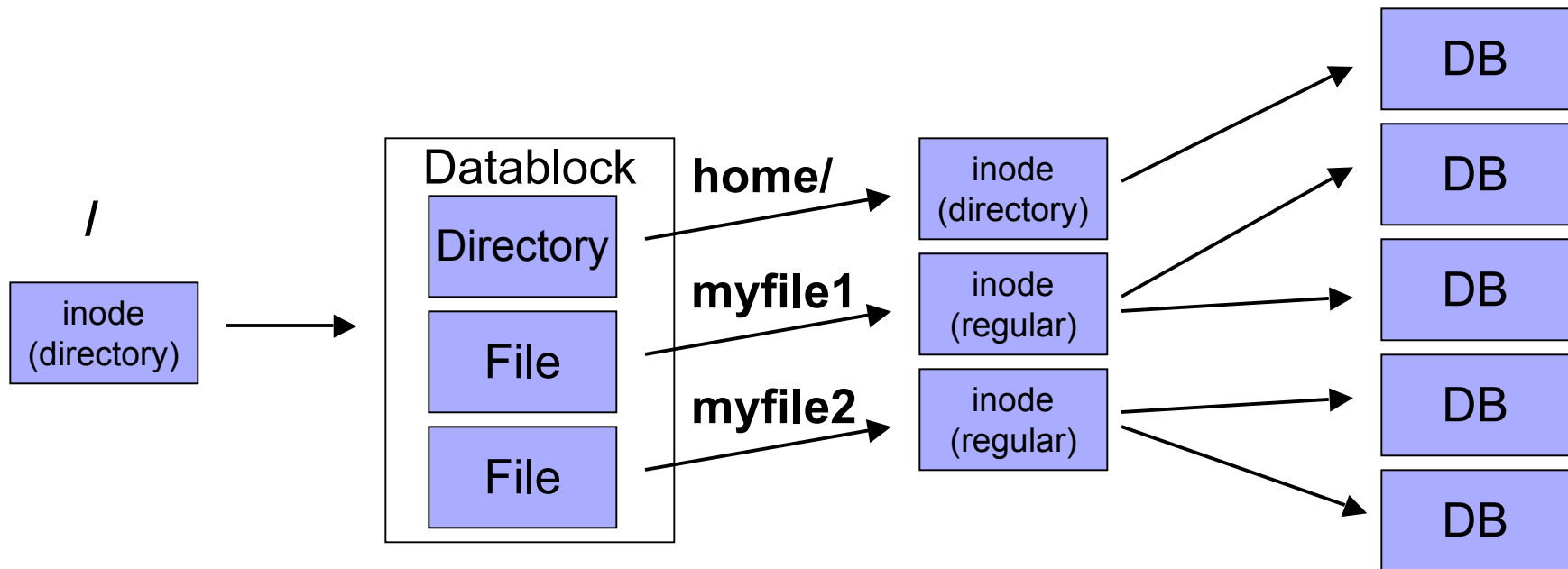
Implementation – File Metadata (inode)

- Type
 - Directory
 - Regular File
- Size
- Owner/Access Control
- Timestamps
- Datablock Pointers
- Indirect Datablock Pointers



Implementation - Directories

- Name
- File Metadata (inode) Pointer





Optimization

- Disk access is very slow
 - Disk is in milliseconds (10^{-3})
 - Memory is in nanoseconds (10^{-9})
- Cache frequently used metadata in memory
- Cache recently used datablocks in memory



Acknowledgements & References

- Joe Pasquale
- Calit2
- Stevens, W. Richard, and Stephen A. Rago. Advanced Programming in the UNIX® Environment. 2nd ed. Upper Saddle River: Addison-Wesley, 2005.
- Bovet, Daniel P., and Marco Cesati. Understanding the Linux Kernel. 2nd ed. Sebastopol: O'Reilly, 2003.
- Silberschatz, Abraham, Greg Gagne, and Peter Baer Galvin. Operating System Concepts. 7th ed. Hoboken: John Wiley & Sons, 2005.