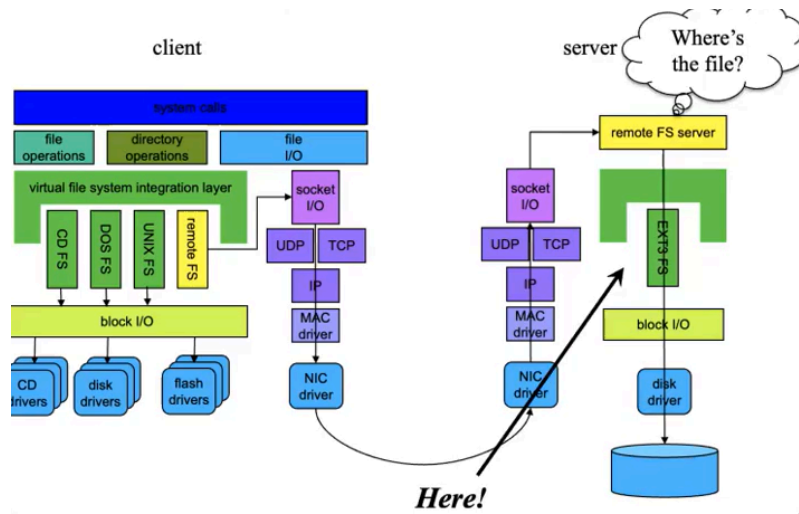


# Remote Access

## Remote Data Access Solutions

- APIs and transparency
  - How do users and processes access remote data
- Performance, robustness, and synchronization
- Architecture and its integration
- Client server protocol and work partitioning

## Remote File Access Architecture



## Remote File Advantages

- Very good application level transparency
- Good functional encapsulation
- Able to support multi client file sharing
- Potential good performance and robustness

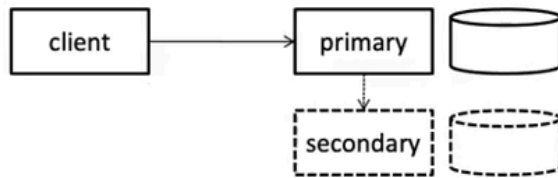
## Remote File Disadvantages

- Part of the implementation is in the OS
- Client and server tends to be more complex
- Model for client server storage

## Remote vs Distributed FS

- Remote file access
  - Client talks to primary server
  - Secondary server may take over if primary fails
  - Advantages: simplicity
- Distributed file system
  - Data is spread across many servers
  - Client may talk directly to many of them
  - Advantages: performance, scalability
  - Disadvantages: complexity

## Remote Disk/File Access



## Distributed File System

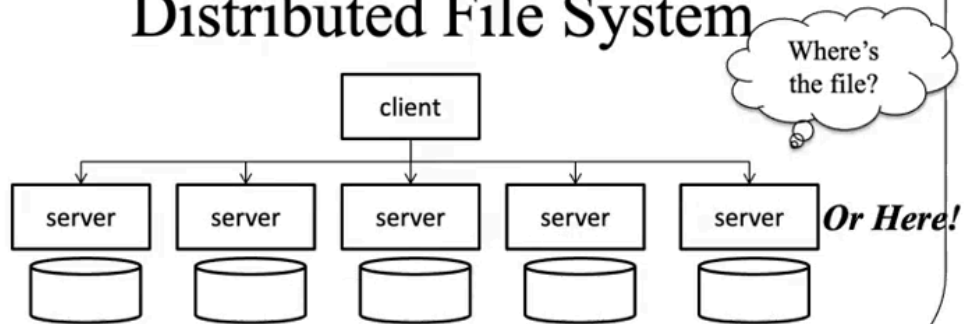


Figure 10

### Cloud Storage

- Encapsulation of servers and resources
  - Resources are abstract/logical
- Advantage: simple, scalable, available