

# Making a Case for Out-of-Band File System Middleware

## For Digital Asset Management

Presented by:

Jacob Farmer, CTO  
Cambridge Computer

- **There is a lot of desirable storage management functionality that neither belongs in the file system nor in digital asset management software.**
  - Many get by with 3<sup>rd</sup> party storage management applications, scripting, home grown applications
  - Some go ahead and integrate storage management functionality into their applications
- **The logical solution is to insert a middleware layer to bridge the gap between application and file system.**
  - IRODS and SRB attempt to do this
  - We believe that there is a need for something lighter weight that sits out of the data-path.
  - We are not sure exactly what it does, what the user interface looks like, and how best to implement an API

## • Data Migration

- Life cycle management and prioritization of resources
- Data protection (backup and replication)
- Hardware refresh
- Workflow automation
  - Process workflows
  - Curation workflows
  - The hand-off from research to curation

## • Geographic placement/distribution of data

- For collaboration or for redundancy

## • Data integrity verification

## • Reporting and chargeback

## • Access control – beyond traditional file system permissions

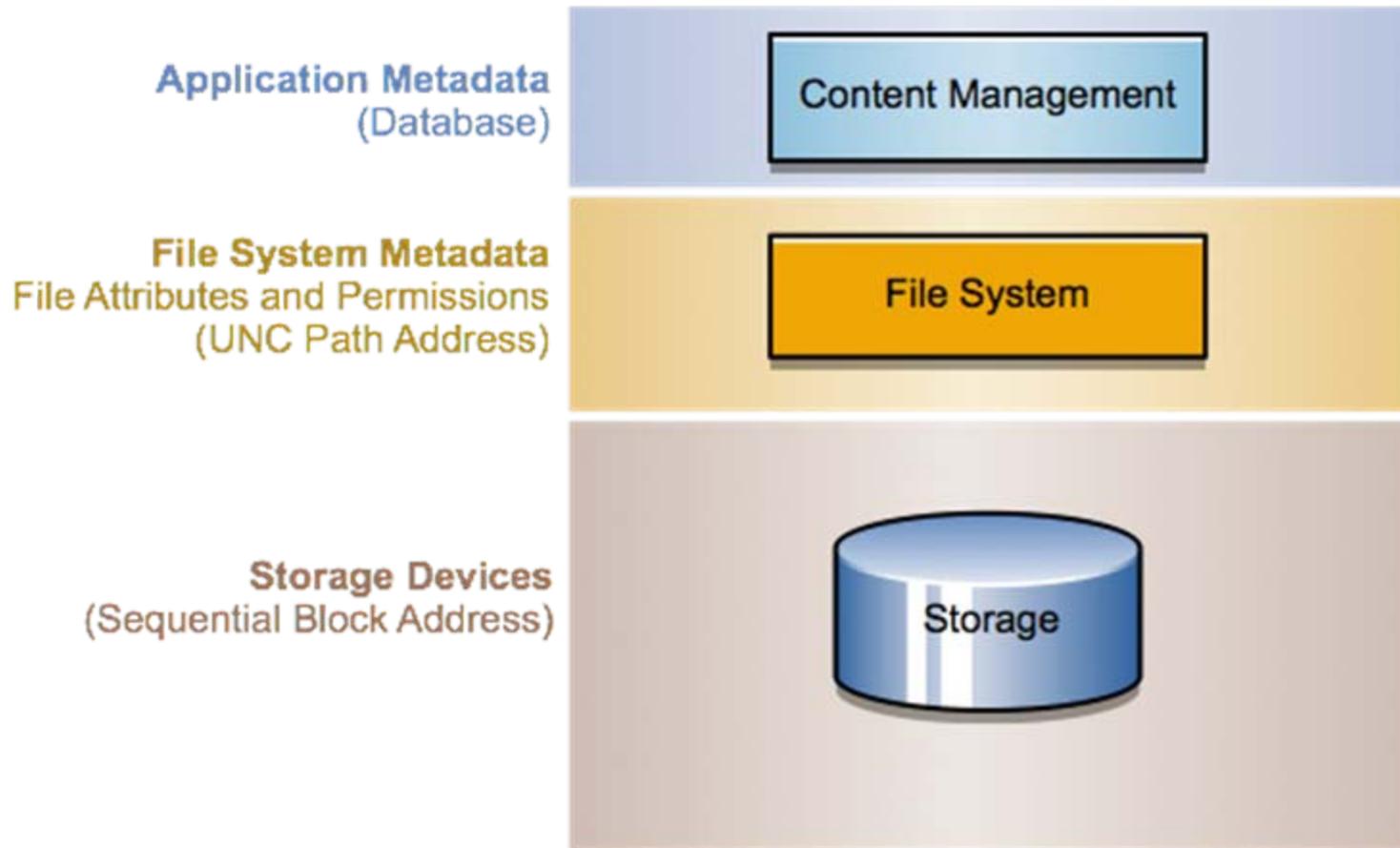
# Bridging the Gap Between Object Storage and File Systems



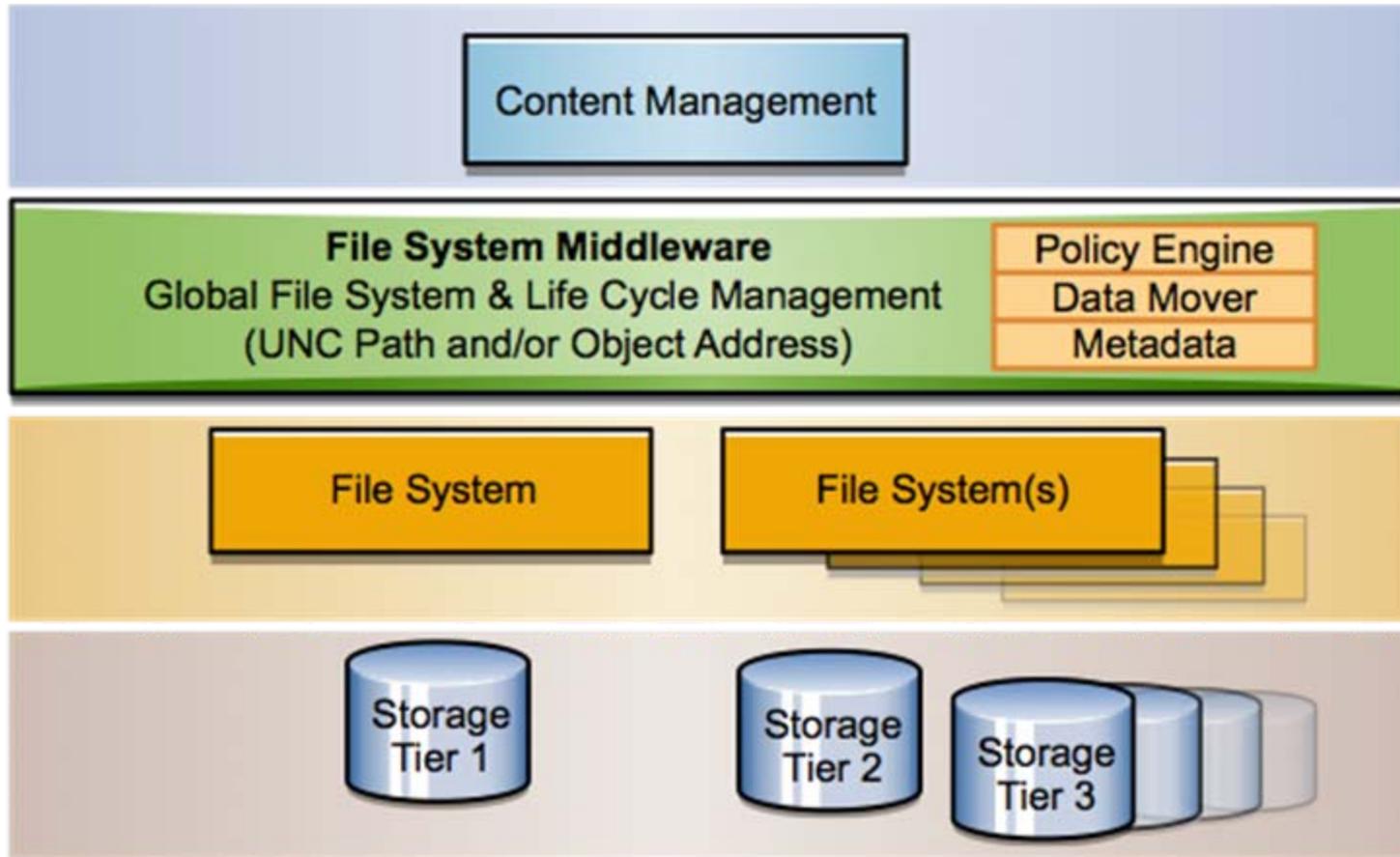
- Several vendors are marketing flat object storage systems as alternatives to conventional tree-based file systems.
- Features include:
  - Immutability
  - Ease of expansion
  - Unique persistent identifier (UUID)
  - Expandability
  - Redundancy
- Object stores often perform other functions
  - Limited metadata management
  - Replication
  - Data integrity verification
- How do you leverage these technologies?
  - Reconcile the world of file systems to that of object storage
  - Support object stores from multiple vendors
  - Migrate between object stores



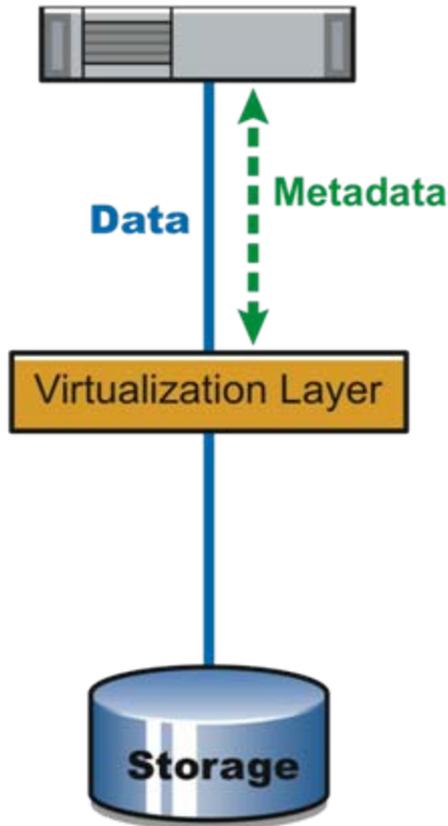
# Typical Content Management “Stack”



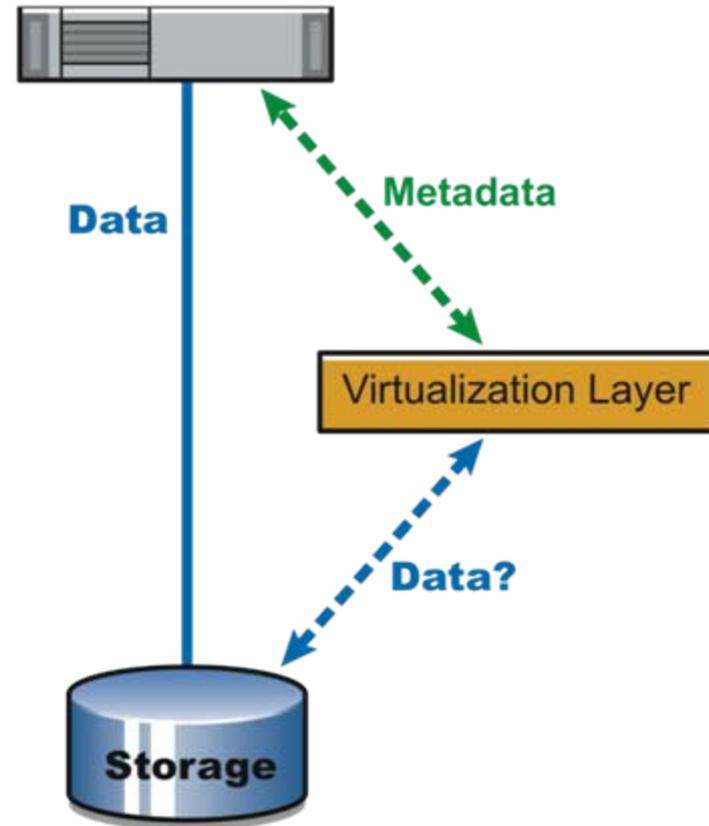
# Inserting File System Middleware



# Storage Abstraction Logic Can Sit In-band or Out-of-band



In-band Virtualization



Out-of-Band Virtualization

# Where Does it Sit: In-Band or Out-of-Band?



- **In-Band: This is a tough place to sit**
  - Must be highly available
  - Must take responsibility for writing data with high integrity
  - Will introduce latency, which is a problem for many applications
- **Out-of-Band: No impact on performance, but**
  - Some lag time for the system to synchronize
    - Lots of file system crawling
  - Need really slick user interfaces to entice users to embrace the system.
  - Need some kind of carrot/stick mechanisms to get users to your bidding

