

# OpenDedup

Cloud Storage Gateway for Netbackup

# Integration Points

# What Gaps Does OpenDedupe Fill

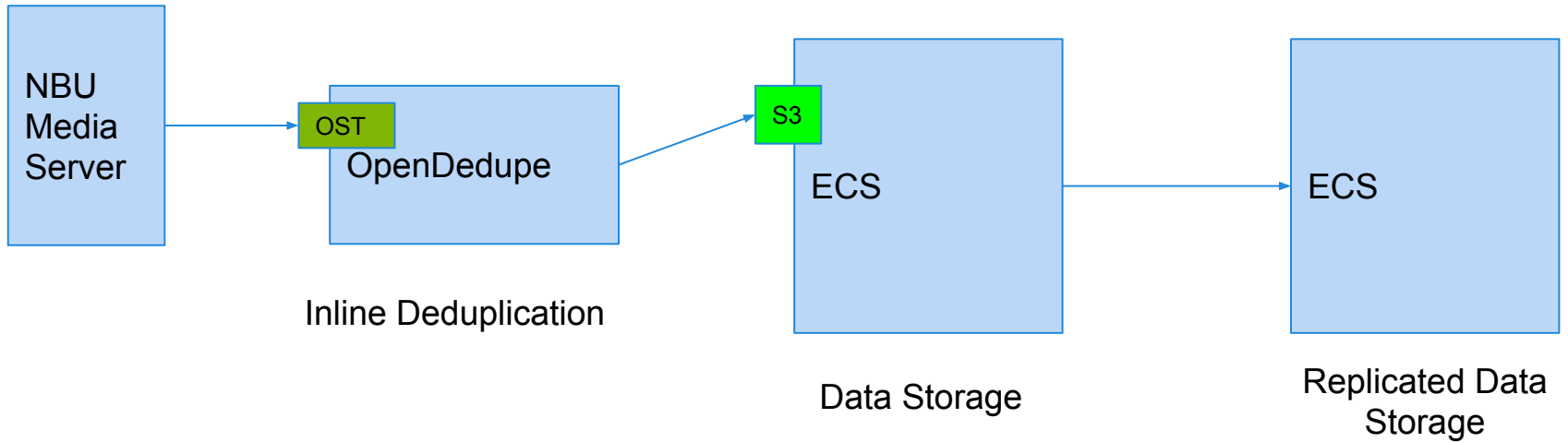
- Dedupe To the Cloud from
  - Enterprise Vault
  - NetBackup
  - BackupExec
- Dedupe To Our Solutions
  - NetBackup Appliance as Cloud Storage Gateway
  - InfoScale Access as Storage Target
- Solve Specific Challenges
  - Global Deduplication across domains
  - Workloads that do not dedupe well with MSDP
    - DB Dumps
    - Vaulted NDMP Dumps
    - Any workloads that dedupe well with Data Domain and Don't with MSDP

# OST Integration

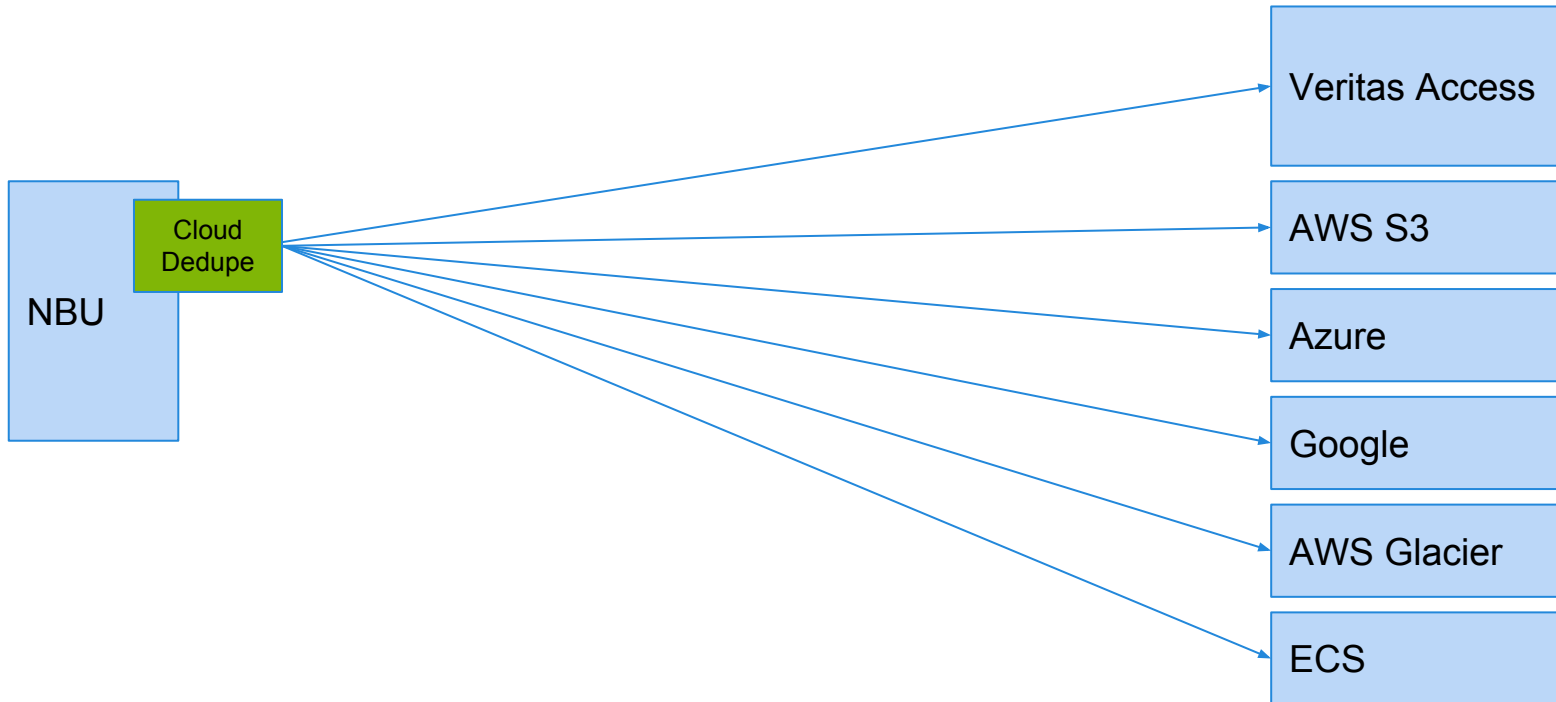
Today:

- Accelerator - All forms
- Global Dedupe across media servers/domains
- Optimized Duplication (In Qualification)
- Replication (In Qualification)
- AIR (In Qualification)

# ECS Architecture



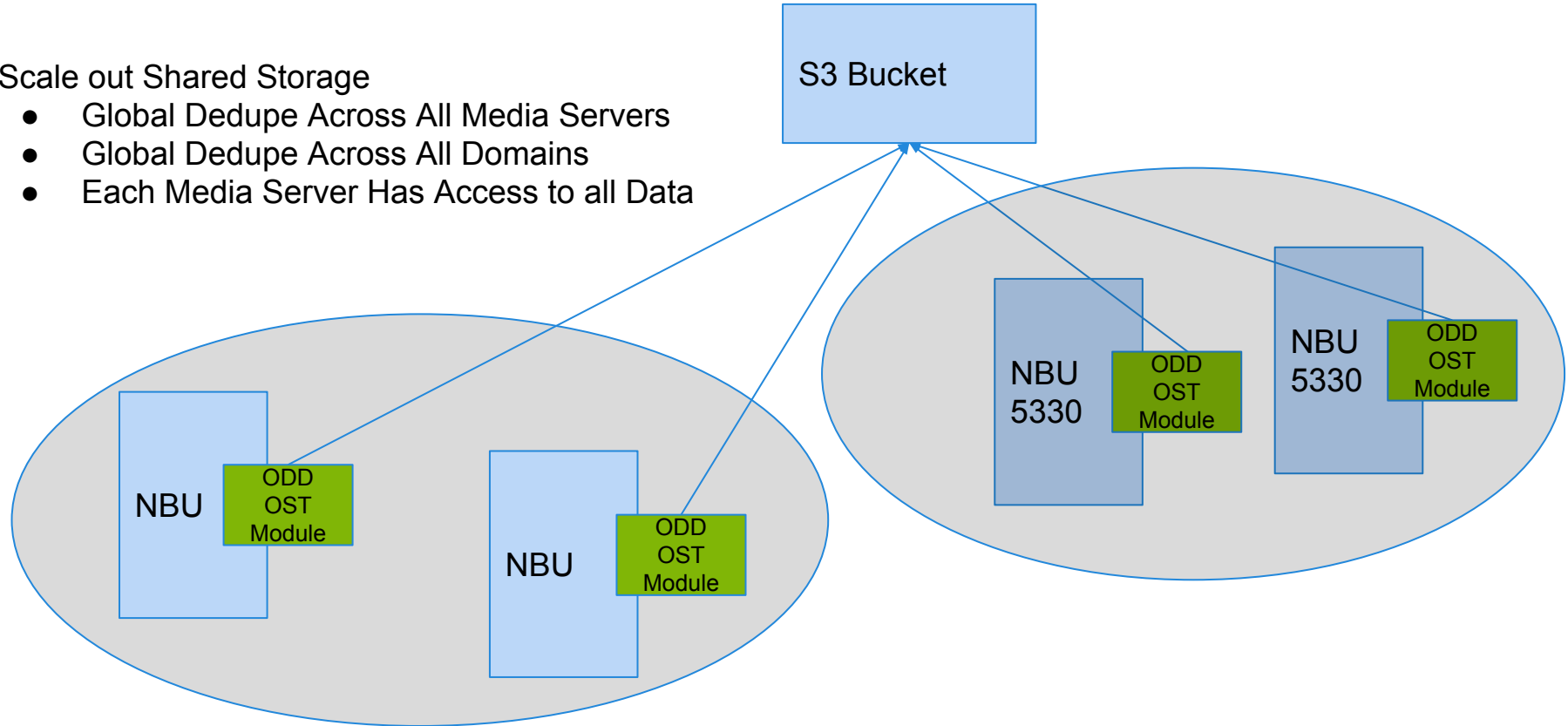
# NetBackup Appliance Architecture



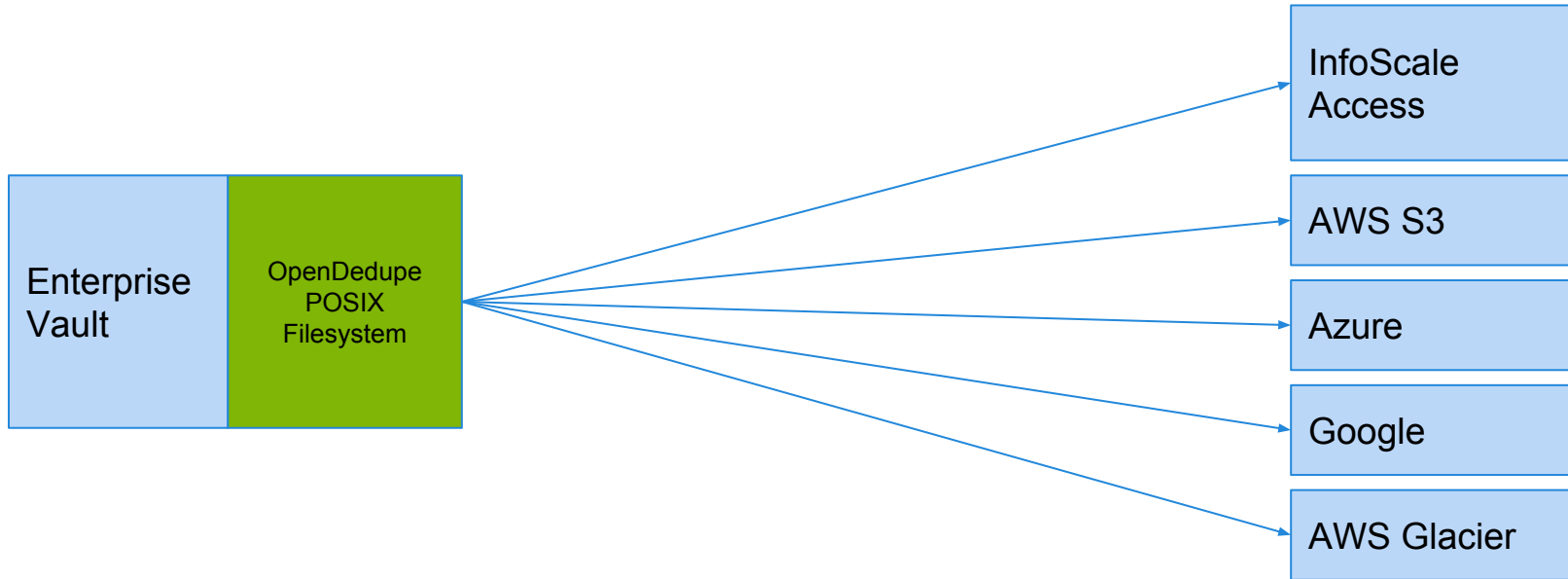
# NetBackup Appliance Architecture

## Scale out Shared Storage

- Global Dedupe Across All Media Servers
- Global Dedupe Across All Domains
- Each Media Server Has Access to all Data



# EV Architecture





# What is OpenDedupe

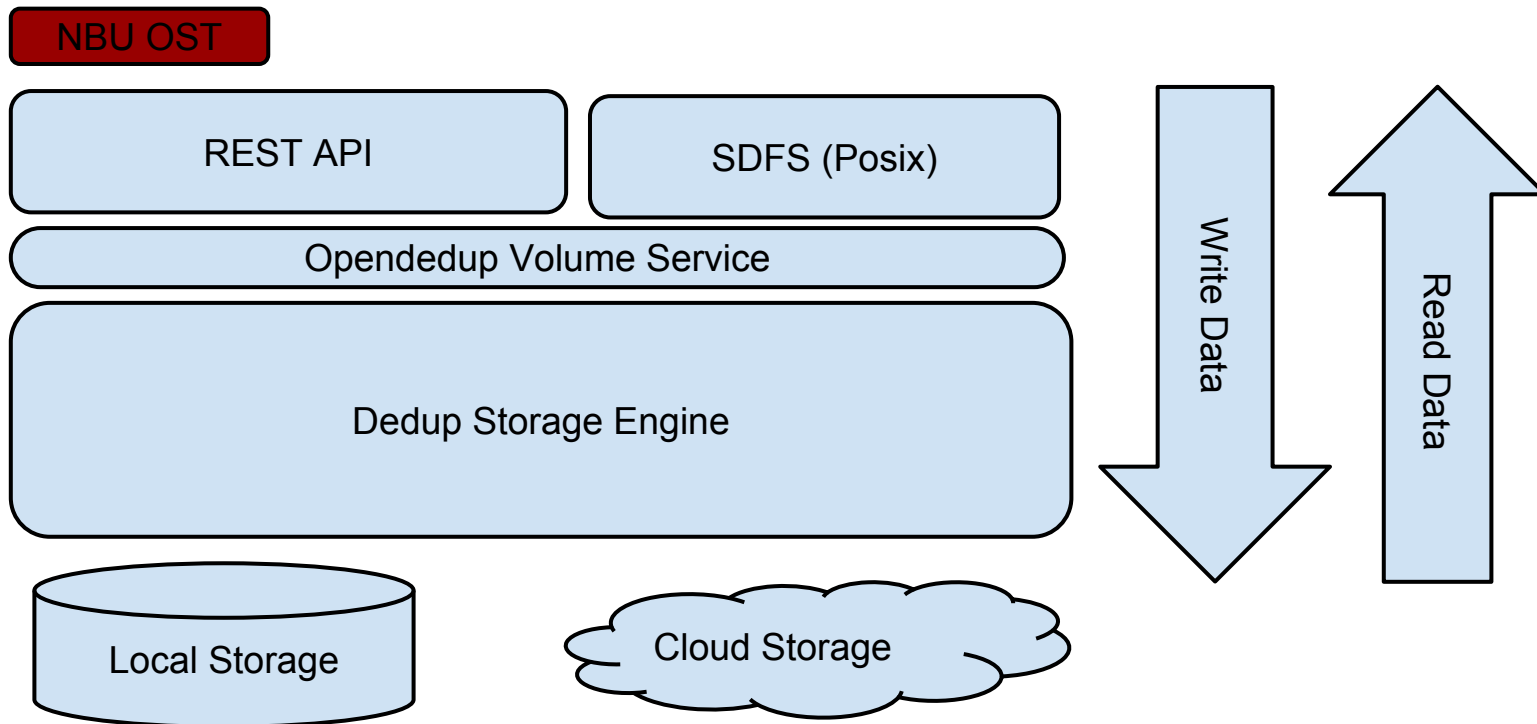
A storage gateway to scale out storage

- Scale out block
- Object storage - Cloud Storage
- Hybrid

# What is it used for?

- Local Block or Cloud Object
- Backup Target
- Archive Target
- VM Storage
- SSD Data Reduction

# Architecture



# General Cloud Capabilities

- **Local Performance** - Most recently used data can be cached locally
- **Security** - All data is locally encrypted with AES-256-CBC before it is sent to the cloud or on disk
- **Data Reduction** - All data is deduplicated and compressed before it is sent to the cloud.
- **Bandwidth Control** - Cloud Storage IO can be throttled for upload and download speeds
- **Replication** - WAN Efficient Replication between Cloud Storage Gateways
- **Glacier, Azure, S3, Google, Swift, block**
- **Instant Recovery** - in the cloud or on prem

# Steps for Deduplication

1. Write to Filesystem
2. Incoming Write is cached to buffer
3. When buffer is full, times out, or sync variable block chunking occurs
4. Chunks compared to hash table
5. Unique chunks spooled to blocks of 60MB
6. Blocks uploaded to cloud after sync, timeout, or 60 MB

# Performance and Scale

- Validated to 260TB of Unique Storage
- At 98% Deduplication rate
  - 1800 MB/s per CPU (16 Core)
- At 0% Deduplication
  - IO and Network Bound
  - S3 performance observed at 450 MB/s
  - Theoretical Max is 1600 MB/s per CPU (16 Core)

# Object Storage is a Great fit for Archive data

- Low Cost Storage
- Built in Replication
- Built in HA
- Infinite Scalability

# Cloud Storage Challenge

- Limited WAN Bandwidth to cloud providers
- Storage costs still a concern
- Encryption in public cloud
- HTTP Puts and Gets can get expensive
- Data Retrieval can take time
- No simple way to integrate legacy applications



# Today's Solutions

Use deduplication and compression to leverage to cost dynamics of object storage.

# 1st Generation Gaps

- Does not leverage the resiliency capabilities of Object storage because data is only available in one place at one time.
- Deduplication required finite scalability because of math/physics
  - Hashtable Requires RAM
  - Metadata Requires Local Disk

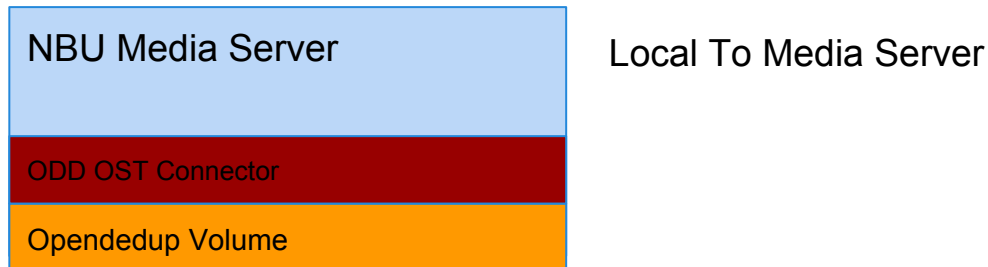
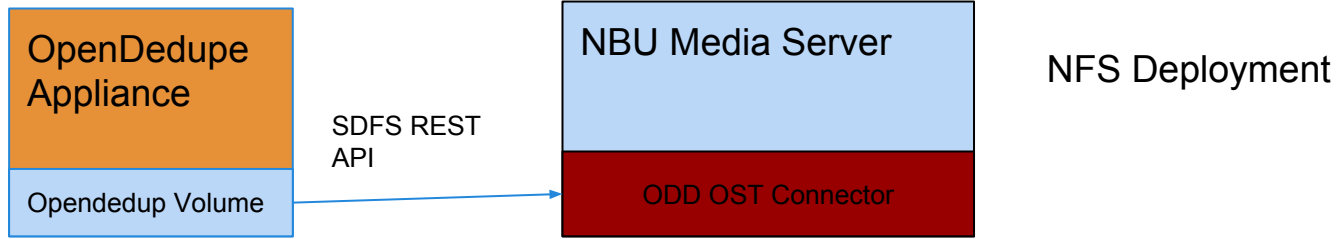
# OpenDedupe

- Shared Instant Access to protected data
- Infinite scalability
- All the same deduplication benefits as a 1st generation target solution

# Why OpenDedupe

- Can be leveraged as a storage platform
- Works natively on all form factors and products
  - 5330/5230
  - NetBackup Media Server
  - Backup Exec
  - Enterprise Vault
- No Special Hardware Required
  - No SSD
  - No Second Server
- Works with all major Vendors and Tiers
  - Glacier
  - IA
  - Azure, S3, Swift, Google

# Netbackup Deployment Patterns



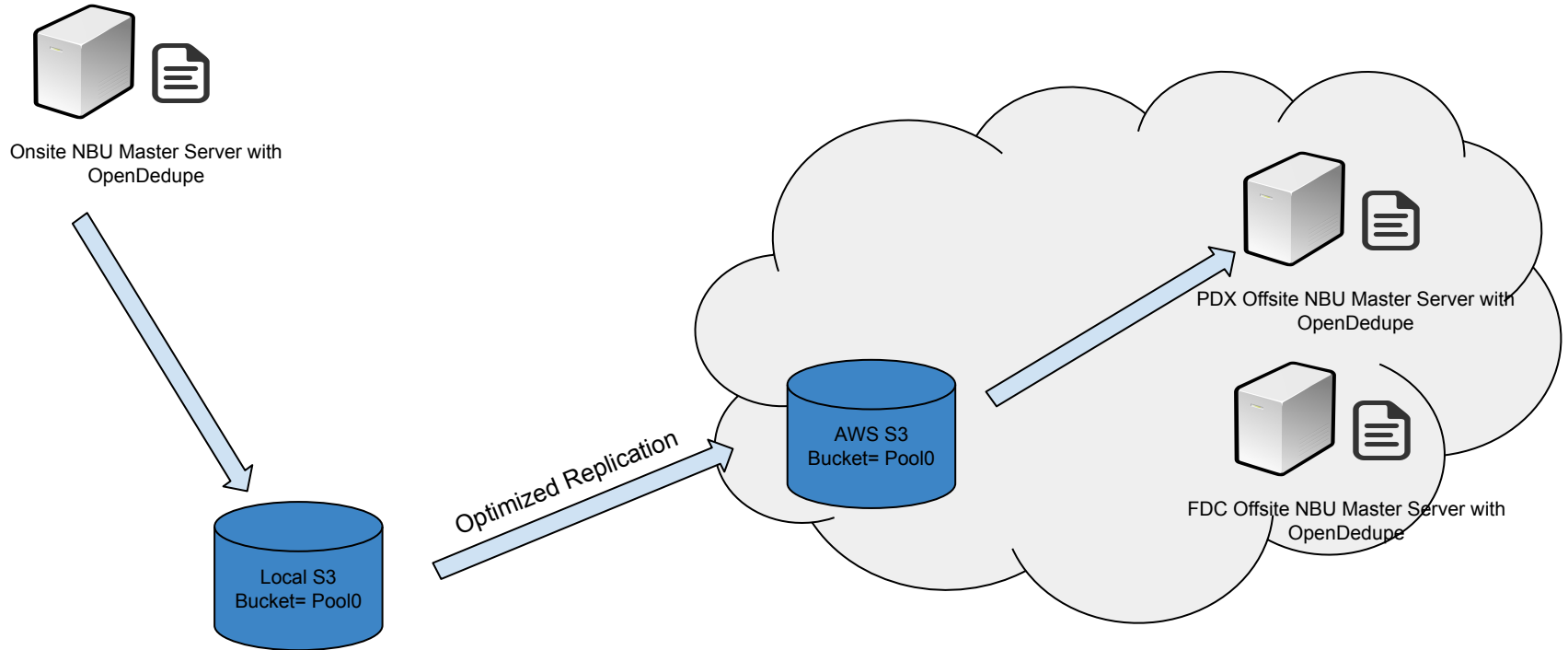
# OpenDedupe Air

- Indirect -> Traditional Replication between data stores
- Direct -> Zero Datamovement Air

# Indirect

- Replicate between Object Storage Types
  - On-Prem S3 to Cloud
  - On-Prem S3 to DR S3
  - Azure - Amazon
  - ...

# Indirect Data Movement Process

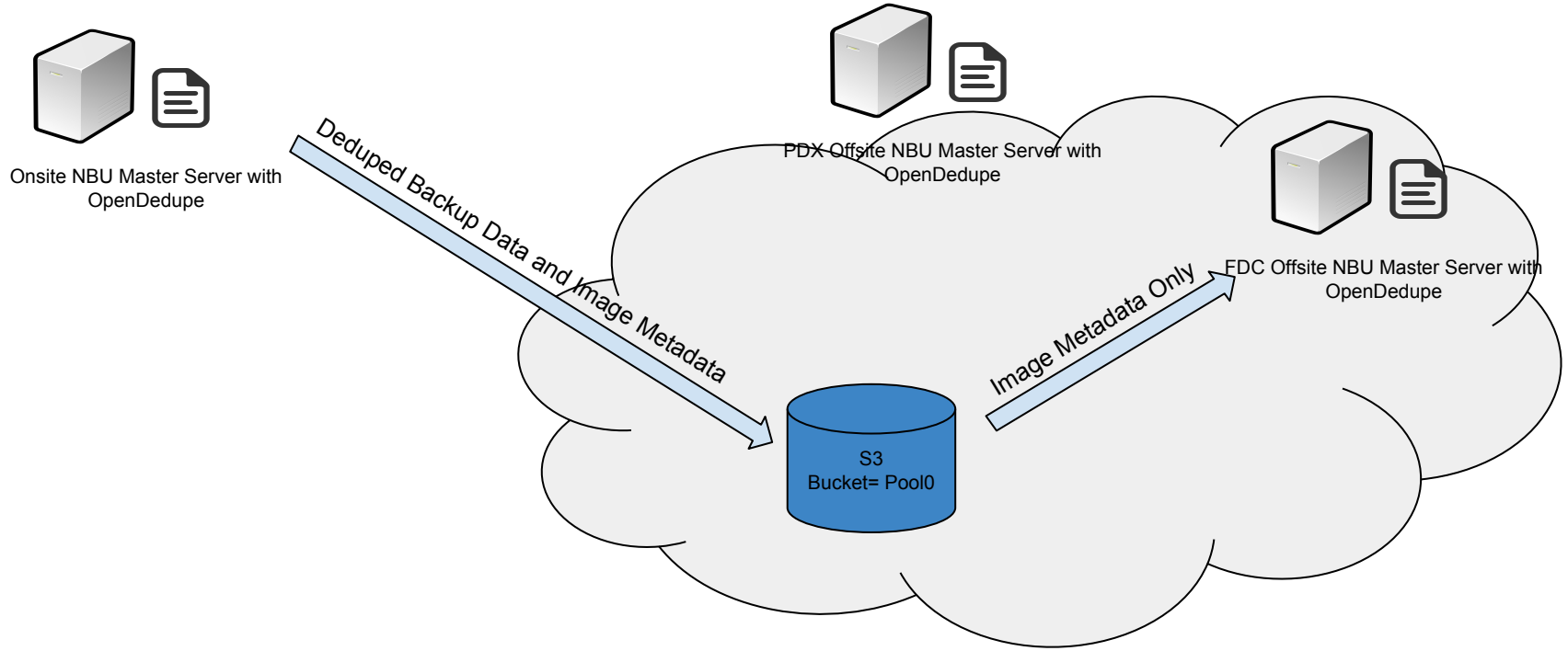




# Zero Data Movement AIR

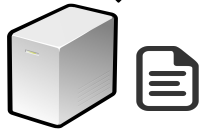
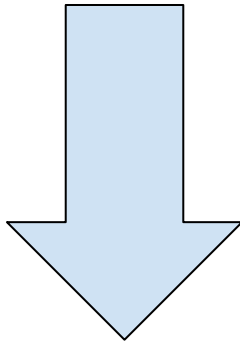
- S3 Bucket Shared Between Source and Target NBU Domain
- Source Domain performs backup
- Netbackup AIR Process Initiated
- Target Domain downloads small metadata elements (.2% size of target backup) to perform restore

# Zero Data Movement Process



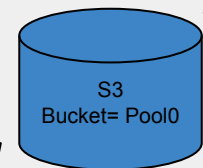
# Demo Setup

Backup Clients

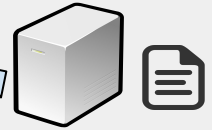


Onsite NBU Master Server with OpenDedupe

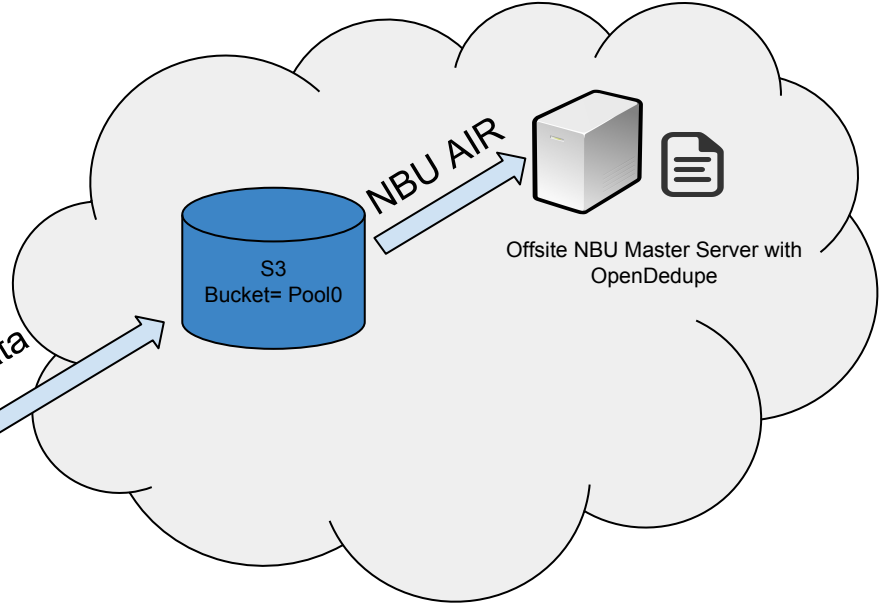
Deduped Backup Data



NBU AIR



Offsite NBU Master Server with OpenDedupe

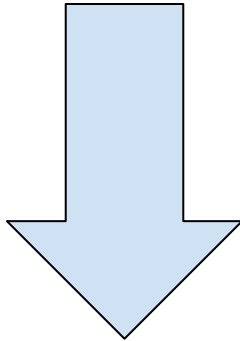


# Benefits and Use Case

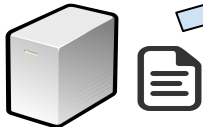
- Benefit - Zero Data Movement for DR Consistency
- Use Cases
  - LTR - Images Can be removed on Source side and kept on target for LTR
  - Cloud DR - Backup datacenter and use Cloud for DR Recovery

# Demo Setup

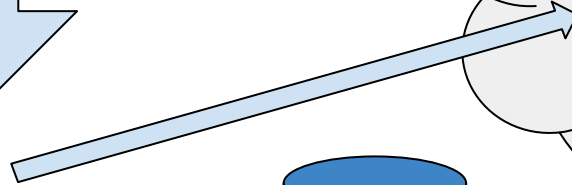
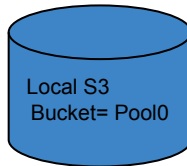
Backup Clients



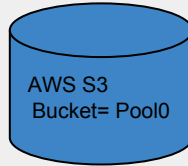
Onsite NBU Media Server with OpenDedupe



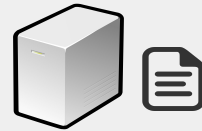
Local S3 Bucket= Pool0



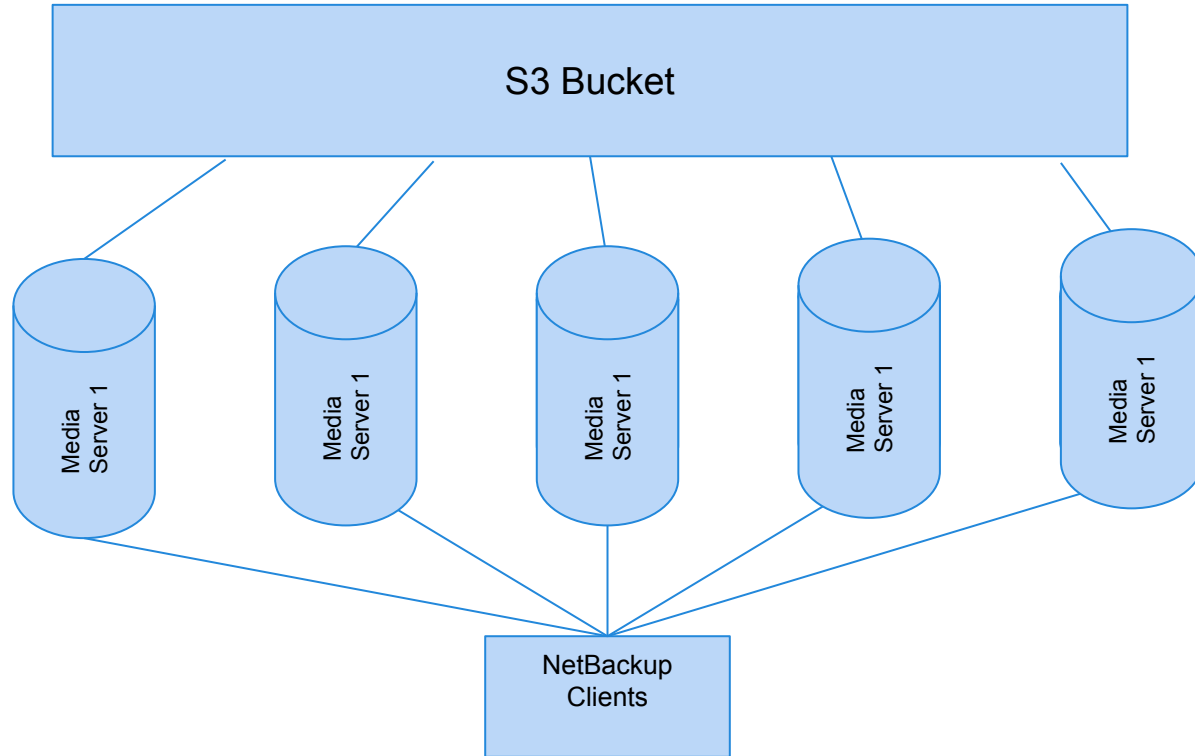
AWS S3 Bucket= Pool0



Offsite NBU Master Server with OpenDedupe



# Clustered Architecture



# Truly Shared Storage

- All Media Servers share images
- Global Deduplication
  - All media servers share the same dedup table and storage
  - An image backed up on one media server is deduped against data backed up on another media server
- Any Media Server can restore image in the cloud or on premise regardless:
  - Where they were backed up
  - What media server backed them up

# It's all about the Data

- OpenDedup Store stores all its data in the object store
  - Hashtable - used for local deduplication during writes
  - Unique Block - Actual Data - Compressed and Encrypted
  - File Metadata - Attributes + location of data in object store. Metadata = 2% of the original file before compression.
  - Any file can be read just from its metadata



# SDFS as A Solution

- Deduplication and Compression
  - Reduces bandwidth
  - Minimizes HTTP Gets and Puts
  - Lower storage footprint
- Local Caching of hot data
  - Reduces bandwidth
  - Minimizes HTTP Gets
  - Increases Read Speeds
  - Data Encryption in flight using AES256 CBC
  - Data is secure at rest
  - Data is secure in transit
- Legacy Integration through Virtual Filesystem Layer
  - Posix
  - NFS
  - ISCSI
  - Windows

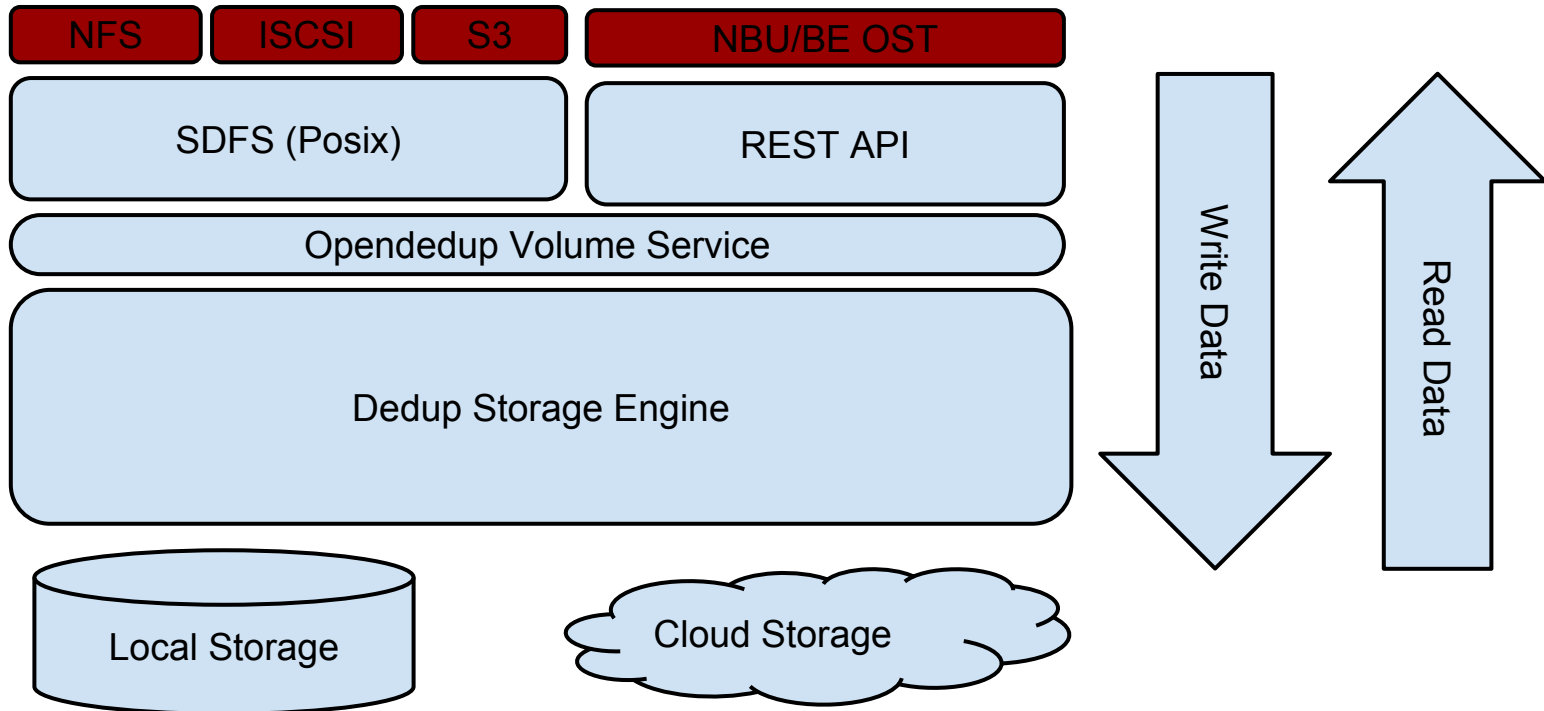
# General Capabilities

- **Multi-OS Support**
  - Linux
  - Windows
- **Flexibility**
  - Designed to support random IO
  - Built in Posix Compliant Filesystem (SDFS)
  - Block Device Support
  - OST support could be added
- **Scalability**
  - Active instances with 100TB of backend storage per node
  - Multi-Threaded
  - N+1 Node Scale out
- **WAN Efficient Replication**
  - Granular to File/Folder level
  - Compression
  - Only unique blocks
  - Encrypted and authenticated transport
- **Deduplication**
  - Inline
  - Fixed Block 4K-128K
  - Variable Block using Rabin border detection
  - Default Murmur3-128 bit hashing
- **Storage**
  - Built in AES-256 CBC Encryption
  - Block level compression
  - Flexible Storage
    - DAS
    - Cloud
    - Clustered Nodes

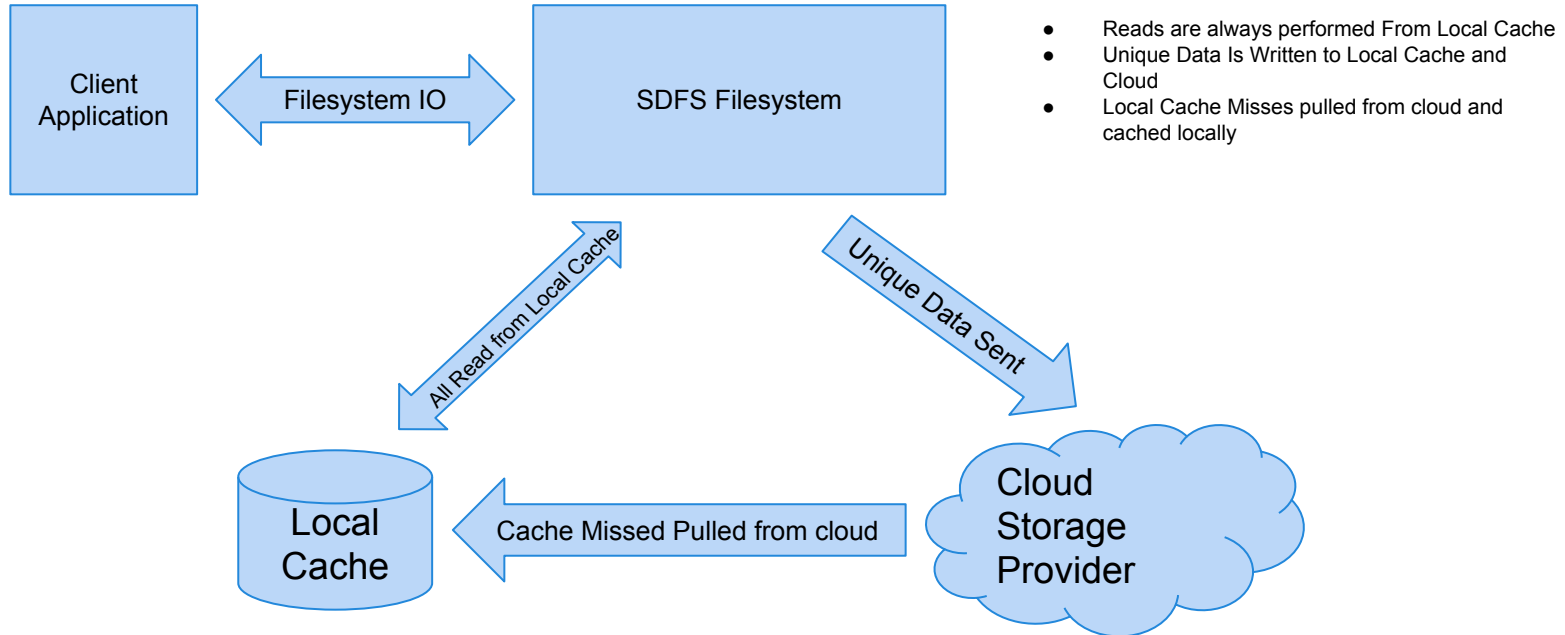
# Enhanced Cloud Capabilities

- Flexible Storage
  - Azure
  - AWS S3
  - AWS Glacier
  - Google
  - Swift
  - Any S3 compliant backend
- Multi-Threaded - Configurable write/read thread throttling
- Configurable upload block size
- Resilient
  - Auto upload restart after crash
  - Hash DB fully recoverable from cloud
  - File Metadata backup recoverable from cloud
- Local active block Caching in MRU capacity
  - Size Configurable local cache pool
  - Stores unique chunks
- Variable Block Performance (Per Cloud Instance CPU)
  - 100% Unique no compression R/W Performance 75-100 MB/s
  - 10% Unique no compression Write Performance 300 MB/s

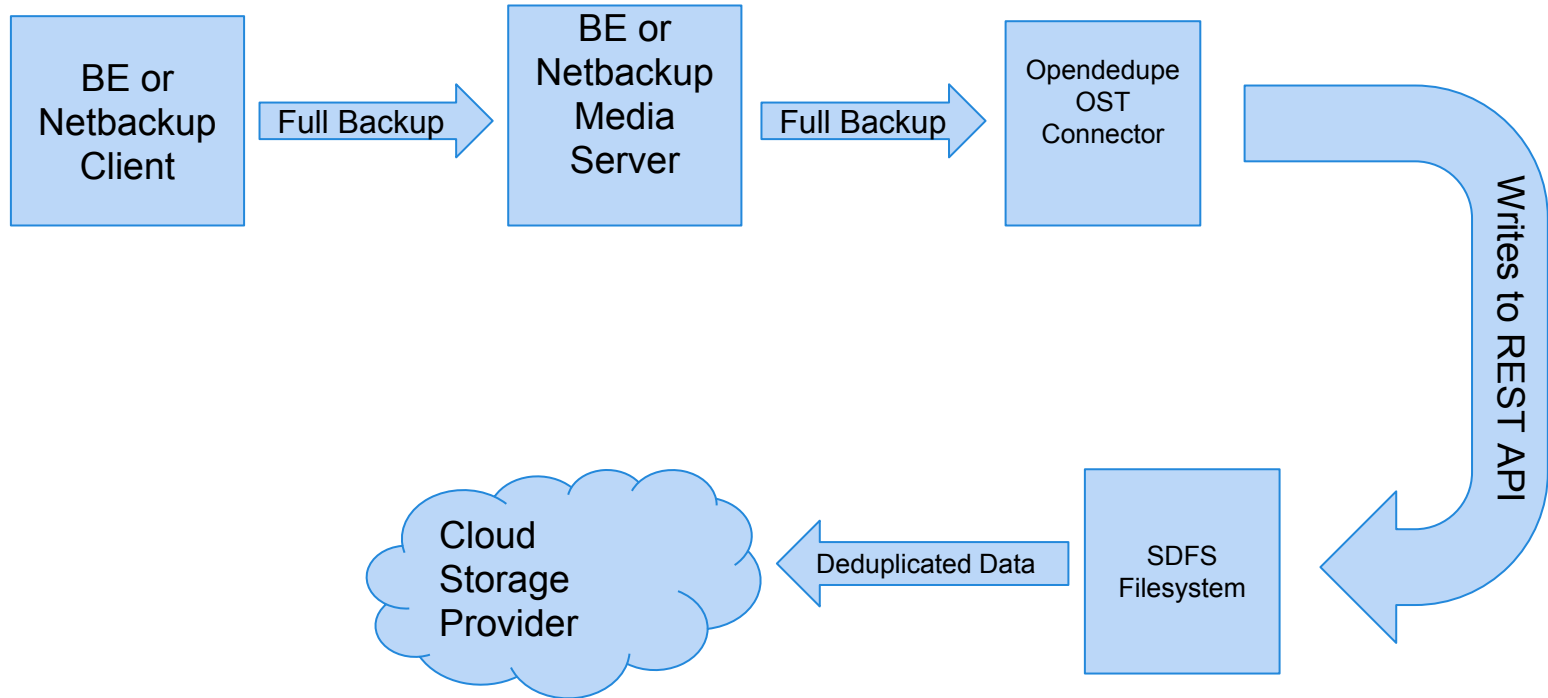
# Architecture



# SDFS Data Flow



# OST Integration



# NetBackup Integration

- Advanced Disk
  - Works on Windows/Linux Media Servers
  - Supports Read/Write
- OST Connector
  - Developed for RHEL 7
  - Supports NBU 7.7+
  - Functions Supported
    - Read
    - Write
    - Accelerator
- Accelerator Performance up to 2000 MB/s

# Components

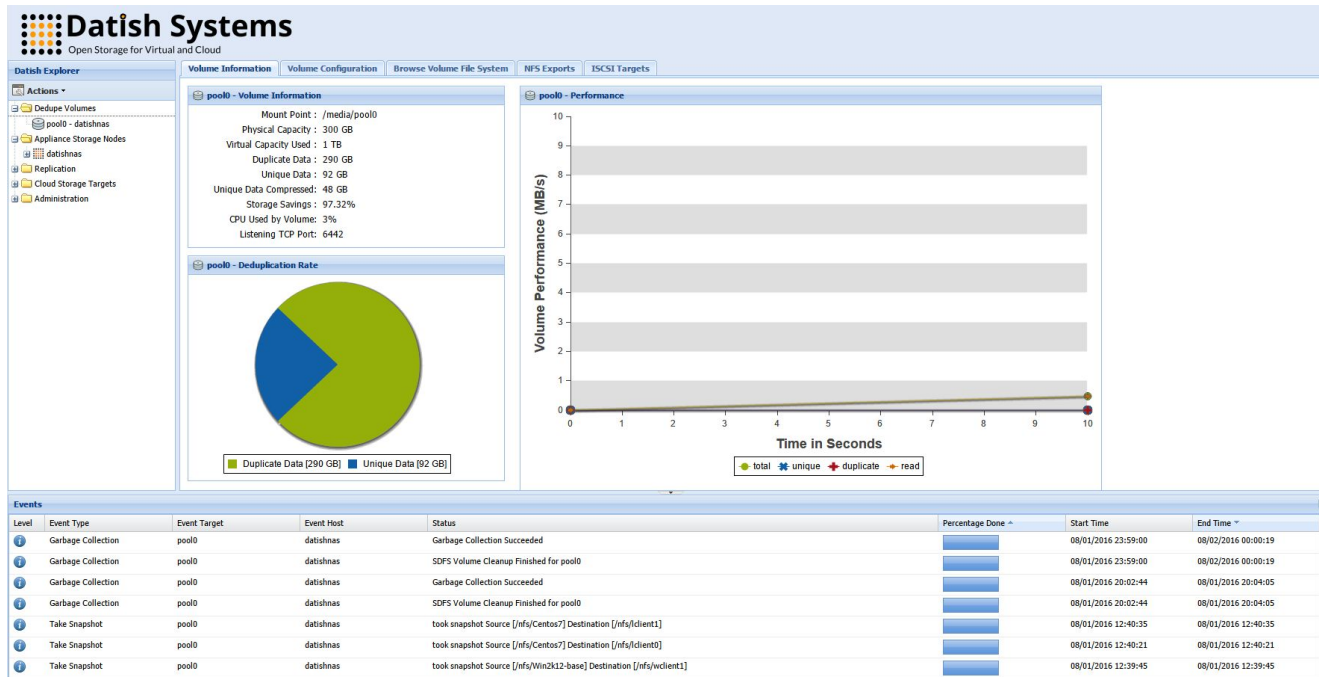
- SDFS - Provides FS Emulation
- Volume Service
  - Hashes Data
  - Stores File Metadata
  - Manages Random IO
- Dedup Storage Engine
  - Manages Unique Hash Lookup Table
    - Stores Hash and block reference location
  - Manages backend storage for unique blocks
    - Pluggable storage layer
    - All data is associated to unique hash



# Minimum System Requirements

- 300MB of memory per TB of Unique storage
  - 3GB per TB to keep entire hashtable in RAM
  - Assumes average block size is 8KB
- 2.5GB of local storage per TB of Unique Storage Capacity
  - Persist hashtable to disk

# Virtual Appliance



## Benefits

- Easy to Setup
- ISCSI or NFS
- Built in Replication
- Central Management