SDC
STORAGE DEVELOPER CONFERENCE
SNIA ■ SANTA CLARA, 2014

Implementation of Hadoop Distributed File System Protocol on OneFS

Tanuj Khurana
EMC Isilon Storage Division

Outline

- HDFS Overview
- OneFS Overview
- HDFS protocol on OneFS
- HDFS protocol server implementation
- References
- Q&A

HDFS Overview

- Distributed File System
  - Inspired by Google's GFS
  - Designed for scalability and fault tolerance
  - Fast streaming data access
  - Minimal data motion
- Master Slave Architecture
  - NameNode (Master)
  - DataNodes

http://hadoop.apache.org/docs/r1.2.1/hdfs_design

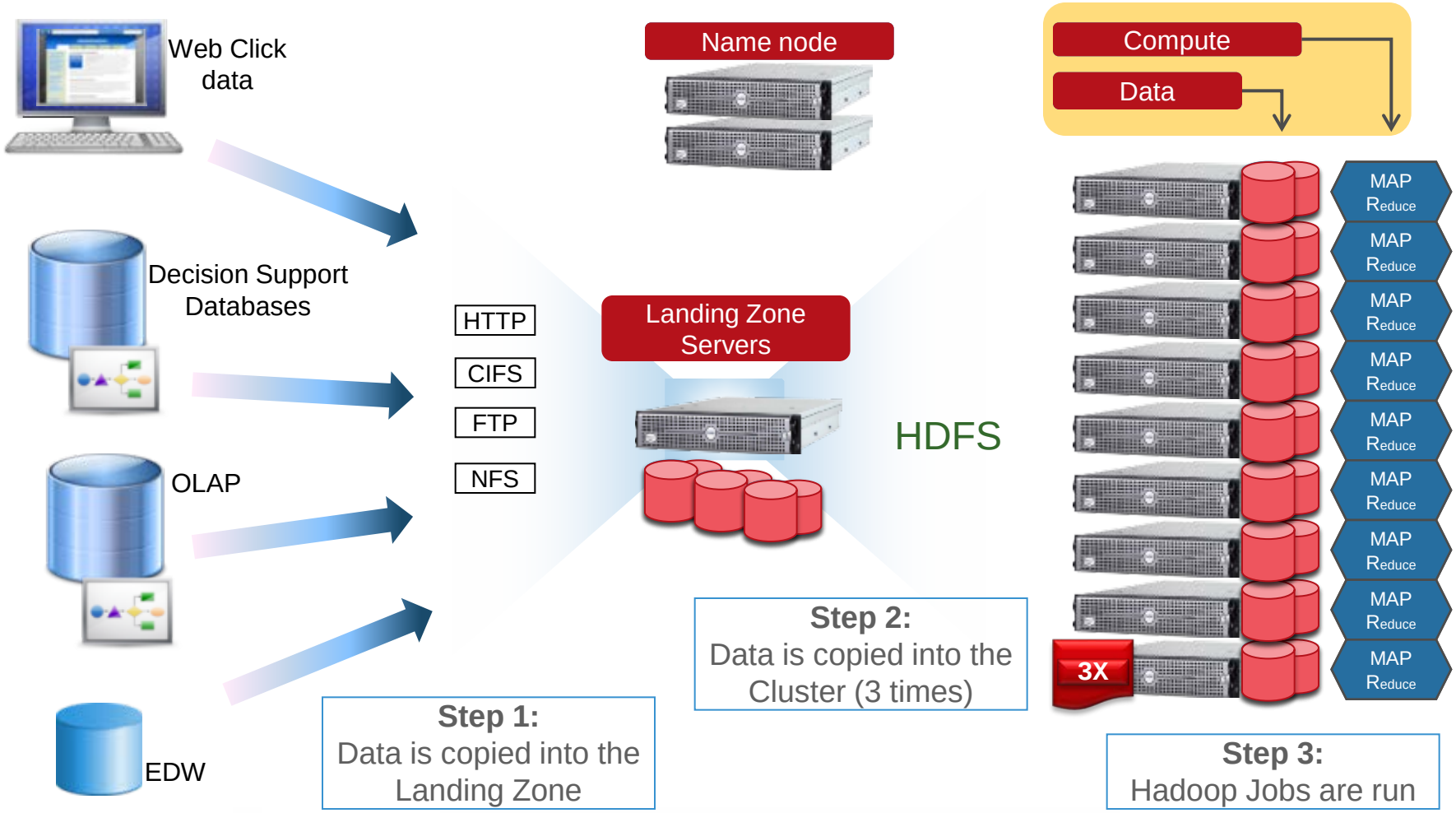
HDFS Overview: NameNode

- Manages the file-system namespace
- Stores all metadata in the RAM
- File names, owners, group, access info
- Maintains file to blocks mapping
- Manages block replication

HDFS Overview: DataNode

- Stores blocks of files on top of native host OS file-system (e.g. EXT3, ZFS)
- Same block is replicated on multiple data nodes for redundancy (typically 3X)
- Has no “awareness” of data blocks living elsewhere (only the NameNode does)

HDFS Overview: Workflow



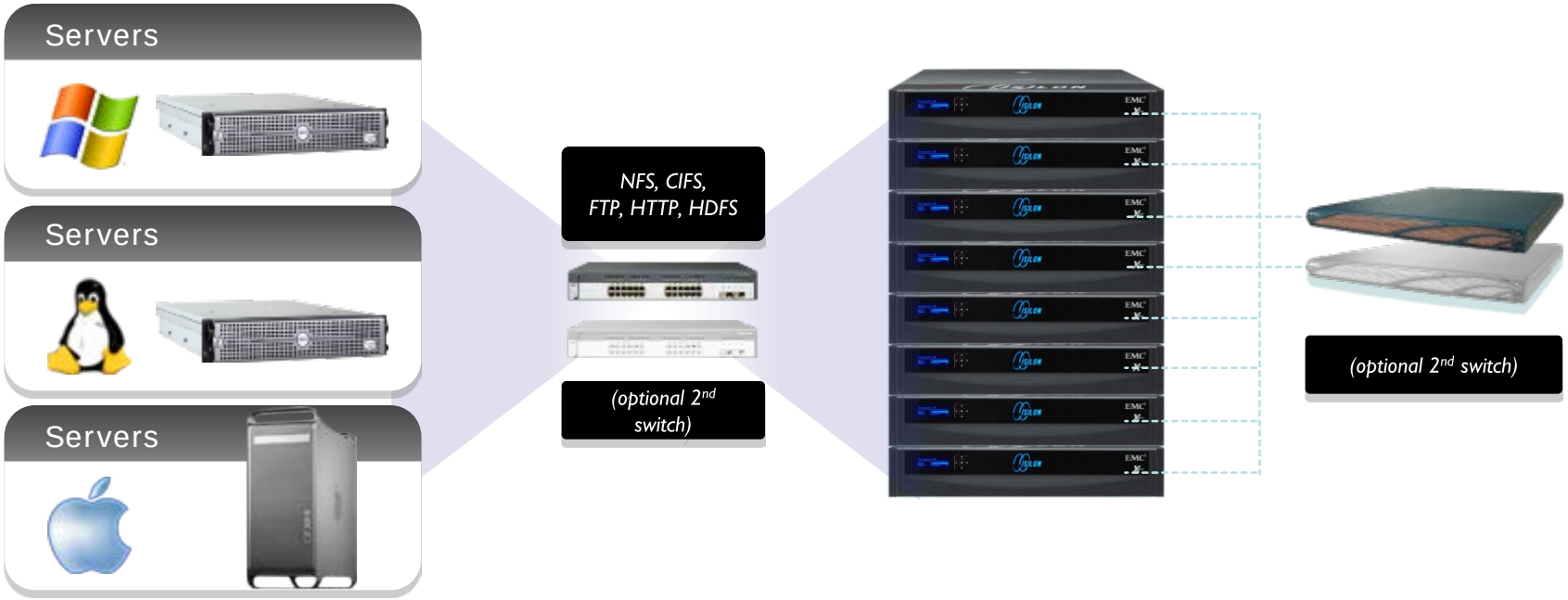
OneFS Overview

- Built from the ground up on FreeBSD
- Distributed scale-out file system
  - Posix compliant
  - Built in support for Data Protection, Snapshots, DR, Audit, Deduplication
- Support for multiple protocols
  - SMB, NFS, HTTP, SWIFT, HDFS

OneFS Overview: Semantics

- Symmetric cluster architecture
  - Metadata distributed across all nodes
- Globally coherent file system access
  - Distributed lock manager
  - Two-phase commit for all write operations
- Reed-Solomon FEC used for data protection

OneFS Overview: Architecture

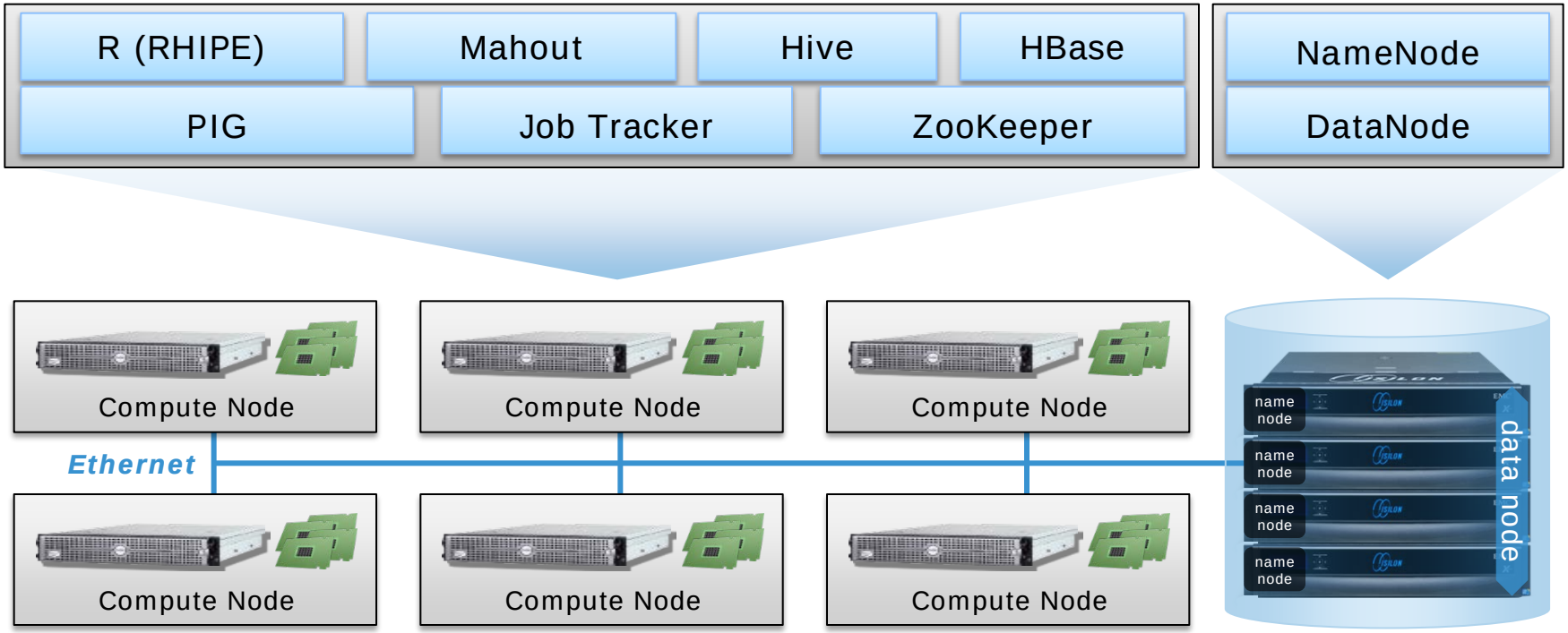


Client/Application Layer	Ethernet Layer	Isilon IQ Storage Layer	Intracluster Communication Infiniband

HDFS protocol on OneFS

- Implements the HDFS interface for Client-NameNode and Client-DataNode
- Each Isilon node runs a NameNode and DataNode service
- Underlying file system is OneFS

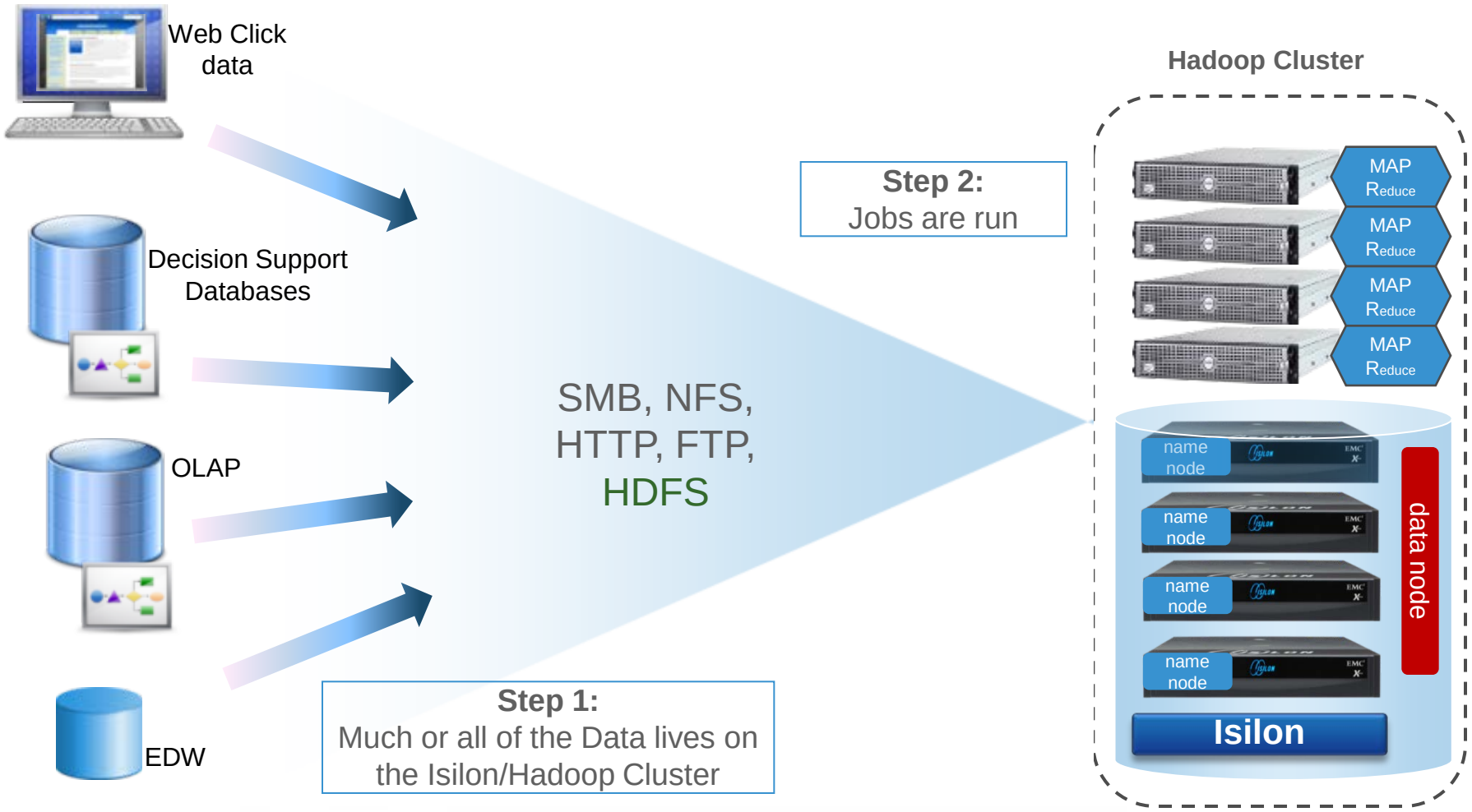
HDFS protocol on OneFS: Architecture



HDFS protocol on OneFS: Benefits

- Multi-protocol access
  - No data ingestion, faster time to results
  - Single repository for all data
- Scale compute and data independently
- Higher storage efficiency (OneFS: 80% usable)
- Active-Active NameNode architecture
- Simultaneous multi-distribution and multi-Hadoop version support
- More data management options (Snapshots, DR, Audit etc …)

HDFS Workflow on OneFS



HDFS Protocol Impl: NameNode

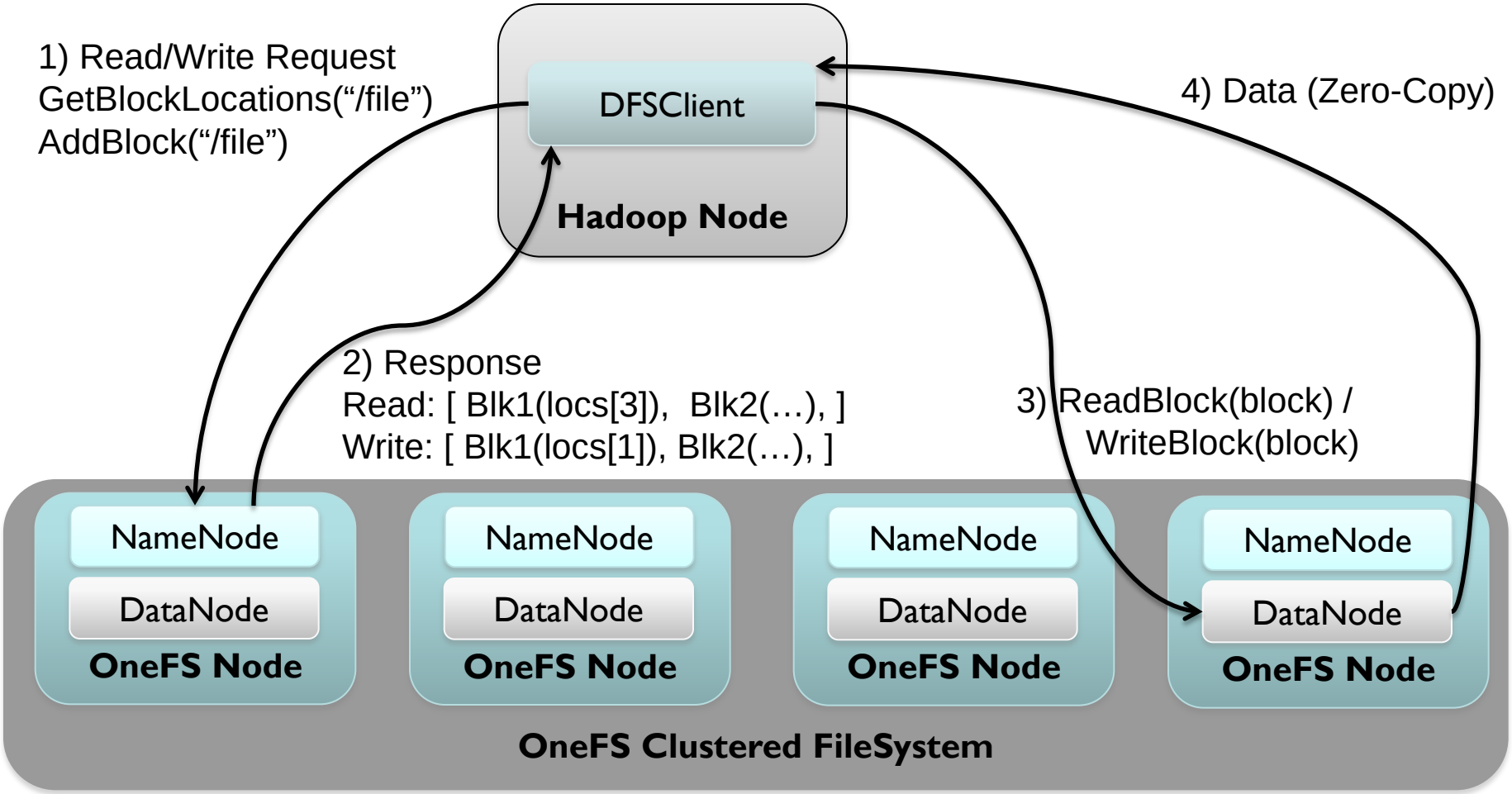
- Most RPCs translate to POSIX system calls
  - setPermission() → chmod(…)
  - setTimes() → utimes(…)
  - create() → open(…, O_CREAT, …)
- Other RPCs need creative interpretation
  - getBlockLocations(), addBlock(), abandonBlock()
  - renewLease(), recoverLease()
- Implements multiple versions of the protocol
  - V1, V2 and V2.2
  - Versions have different wire formats

NameNode Connection Routing

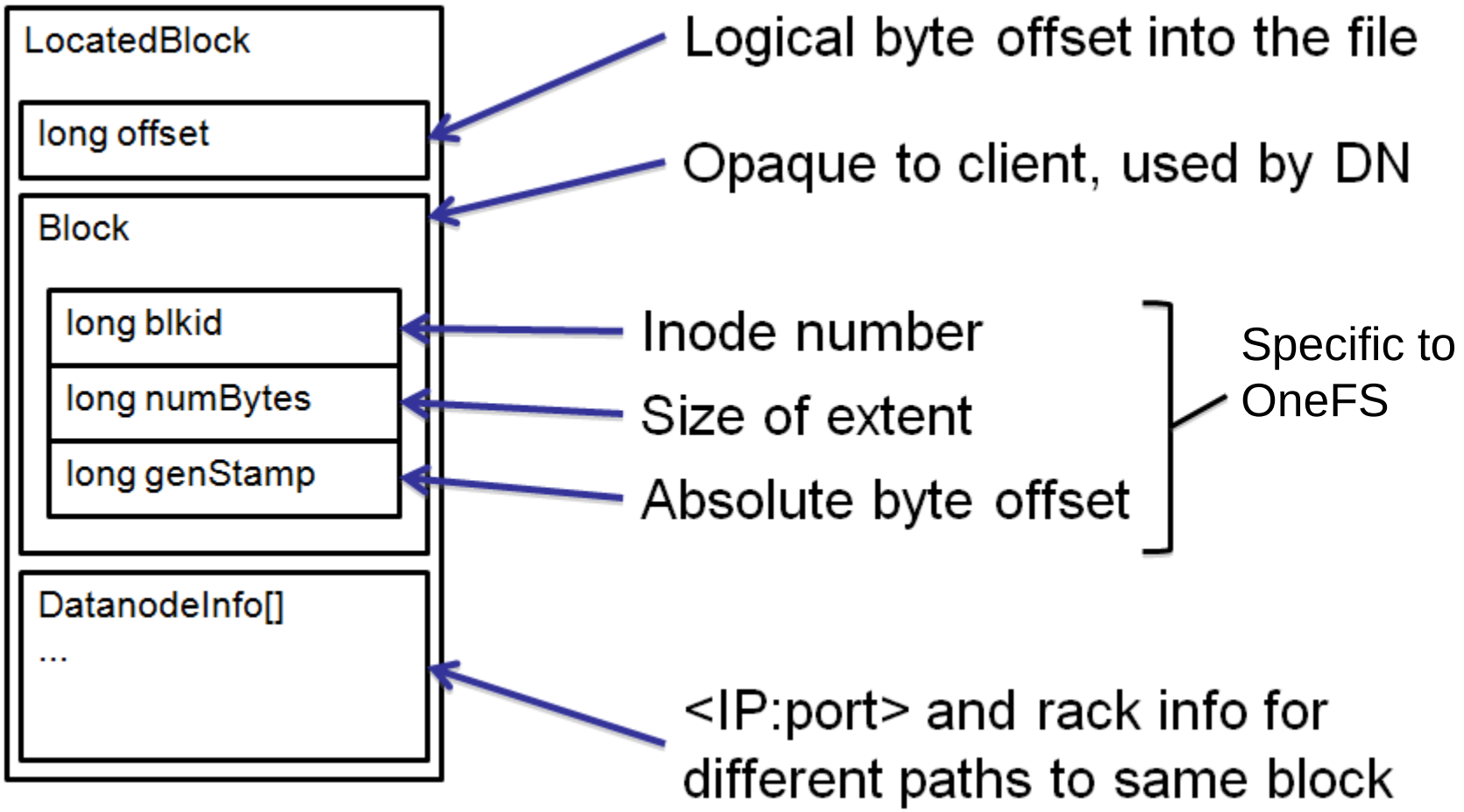
- NameNode is configured as single URL
  - Easy configuration:

    Set fs.defaultFS to hdfs://smartconnect.isilon.com:8020/
- DNS round-robin to distribute across nodes
  - Metadata IOPs get spread out
  - OneFS maintains cross-node consistency
- IP Failover plus client retries for resiliency

HDFS protocol Impl: Data Path

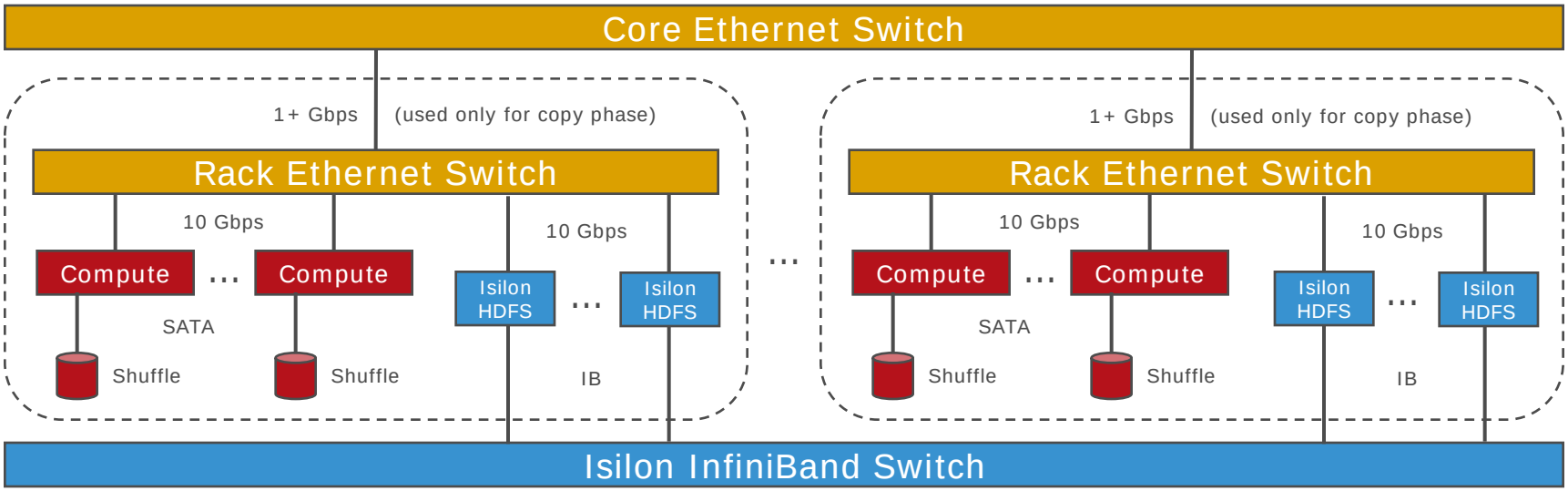


HDFS protocol Impl: Data Path



HDFS protocol impl: Rack locality

- Configure racks to limit cross switch contention



HDFS I/O ALWAYS comes through a rack-local Isilon node which collects data blocks from all other Isilon nodes across the InfiniBand fabric

HDFS protocol Impl: Authentication

- Simple Authentication
  - Username sent in clear-text on wire, requires name resolution on every access
  - Integrated with different directory services (AD, LDAP, NIS)
- Kerberos Authentication
  - One hdfs service SPN for the cluster
  - Kerberos “provider” manages the keytab and SPNs for both MIT/AD KDC
  - Impersonation supported via “proxyusers”

HDFS protocol Impl: Leases

- HDFS implements a single-writer, multiple-reader model
- Only one client can hold a lease on a file opened for writing, other clients can still read
- Clients periodically renew lease by sending requests to NameNode
- Leases “expire”
- On OneFS, leases are cluster aware because of distributed NameNode architecture
  - Built on top of OneFS Distributed Lock Manager

HDFS protocol Impl: WebHDFS

- RESTful API to access HDFS
  - Popular for scripting, toolkits and integration
  - Used by Apache Hue, a popular HDFS file browser client
- Runs within the hdfs daemon
  - Communicates with Apache web server over a unix domain socket using the FastCGI interface
- Supports both HTTP/HTTPS
- Supports SPNEGO via Kerberos

HDFS protocol Impl: Access Zones

- OneFS solution to Multi-Tenancy that ties together:
  - Cluster network configuration (IP Pools)
  - Authentication providers
  - File protocol access
- Zone context determined based on the cluster IP address the client connects to
- Logically partition cluster into self-contained units

Access Zones + HDFS

- Per-zone HDFS root directory
  - Limits the file-system namespace view
  - Virtualize all file path accesses (e.g. /home/user1 -> /ifs/zone1/home/user1)
- Per-zone HDFS security settings
  - Simple_only / Kerberos_only / All
- Per-zone authentication services (AD, LDAP…)
- Key enabler for HDFS as a Service solution

References

- EMC Isilon OneFS Overview

  http://www.emc.com/collateral/hardware/white-papers/h10719-isilon-onefs-technical-overview-wp.pdf

- EMC Isilon Hadoop White Paper

  http://www.emc.com/collateral/software/white-papers/h10528-wp-hadoop-on-isilon.pdf

- Isilon Hadoop Best Practices

  http://www.emc.com/collateral/white-paper/h12877-wp-emc-isilon-hadoop-best-practices.pdf

- EMC Hadoop Starter Kit

  https://community.emc.com/docs/DOC-26892

Questions ?