EMC SOLUTION FOR AGILE AND ROBUST ANALYTICS ON HADOOP – DATA LAKE WITH PIVOTAL HDB

ABSTRACT

As companies increasingly adopt data lakes as a platform for storing data from a variety of sources, the need for simple management and protection grows. Increased analytic capabilities are needed to meet competitive demands. With breakthrough performance and SQL query optimization, Pivotal HDB supports a rich SQL dialect, including complex query and join operations. EMC Isilon Data Lake Storage provides a simple, scalable, and efficient repository that stores and protects massive amounts of unstructured data to the data lake. Together this enterprise data lake business analytics platform allows organizations to efficiently manage and protect their data lake and rapidly develop predictive analytics to unlock the power of their data.

REDEFINE

EMC WHITE PAPER

To learn more about how EMC products, services, and solutions can help solve your business and IT challenges, contact your local representative or authorized reseller, visit www.emc.com, or explore and compare products in the EMC Store

Part Number H15168

TABLE OF CONTENTS

Executive Summary

Today, every business must operate like a fast-moving, innovative software company to survive. The development of innovative software begins with data, including advanced analytics and machine learning of the data. For the new-generation applications of today, this data resides within the Hadoop Data Lake. Hadoop has drastically changed the economics and capabilities for capturing and storing today's unrelenting data growth. But as these data lakes rapidly expand with new and potentially revolutionary data, making use of the data quickly becomes a challenge. The difficulty of managing and protecting the data lake, and the need for a flexible, efficient, cost-saving multi-tenant architecture become clear. Additionally, to go beyond just storing data to understanding the "dark data," data lake organizations need to be able to rapidly integrate SQL with exceptional query performance.

EMC Isilon® Data Lake Storage provides a simple, scalable, and efficient repository that stores massive amounts of unstructured data to the data lake. It enables organizations to easily and quickly store current data, scale capacity, tune performance and ensure data protection as their data lake grows in the future. This all-inclusive data lake storage helps lower storage costs through efficient storage utilization by eliminating data silos of storage, and by lowering storage management costs for migration, security, and protection.

Pivotal HDB seamlessly fulfills the competitive business demand for business insights and predictive analytics. With breakthrough performance, Pivotal HDB supports a rich SQL dialect, including complex query and join operations. Pivotal HDB also incorporates a cutting-edge, cost-based SQL query optimizer (Pivotal Query Optimizer) that features dynamic pipelining technology.

Audience

This white paper is intended for business decision makers, IT managers, architects, and implementers. With EMC Isilon Data Lake Storage Solution with Pivotal HDB, enterprises can build robust storage infrastructure and rapidly develop predictive analytics to truly unlock the power of the data lake.

Introduction

EMC Isilon Data Lake Solutions for Hadoop combines EMC Isilon network-attached storage (NAS) with integrated support for Hadoop. Isilon is the only data lake platform natively integrated with the Hadoop Distributed File System (HDFS). Using HDFS as an over-the-wire protocol, customers can deploy a powerful, efficient, and flexible big data storage and analytics ecosystem. Isilon's in-place data analytics approach allows organizations to eliminate the time and resources required to replicate big data into a separate Hadoop infrastructure. The additional benefits of managing a single version of the data source include minimized data file security risks and better control of the data for data governance.

Pivotal HDB is an elastic SQL query engine that combines exceptional massively parallel processing, MPP-based analytics performance, robust ANSI SQL compliance, and integrated Pivotal Greenplum Database and open source PostgreSQL that operates natively in Hadoop. Pivotal HDB's robust SQL compliance, powerful cost-based query optimization capabilities, and support for advanced analytics enable companies to implement high-performance analytics solutions on data sets of any size, much more rapidly than traditional Hadoop SQL tools. Hadoop Native SQL provided by Pivotal HDB helps companies to unleash the power of Hadoop and drive significant business change.

Pivotal HDB with Isilon: An Integrated Solution for agile and responsive Hadoop analytics

By combining Pivotal HDB with EMC Isilon, a customer who wants to to unlock advanced 'Systems of Insight' from their data sources can run predictive analytics with higher accuracy utilizing the entire dataset, directly with an **enterprise-level, centralized Hadoop platform.** Additionally, organizations can execute near-real-time, ad-hoc queries at scale, and complete analytics tasks faster – in seconds or minutes, not hours or days. Using ANSI compliant Pivotal HDB, developers can rapidly model advanced statistical functions and algorithms using MADlib to spot relationships, patterns, and trends with speed and at scale, all within the data lake. Using HDB functionality, developers can interact with petabyte range data sets with the familiarity of a complete, standards-compliant SQL interface.

This winning combination is a cost-saving, flexible, enterprise-ready, multi-tenant architecture that provides consistent data protection, security, and efficient operations across all Hadoop architectures and quickly unlocks business insights with exceptional performance. This solution provides full SQL functionality and advanced query capabilities that derive new value from the data lake while they optimize, protect, and secure existing value currently residing in Hadoop repositories.

Challenges of traditional Hadoop Deployments

One of the top challenges data center and Hadoop administrators face today is managing the rapidly expanding and evolving data lake. Specifically those challenges arise when implementing a traditional Hadoop scale-out architecture on commodity servers.

Some of the primary challenges of managing traditional Hadoop scale-out architecture include:

- Hard to deploy and operate
- Poor utilization of storage and CPU
- Inefficient and underutilized direct attached storage (DAS) compounded by Hadoop mirroring of three times or more. This can multiply data center costs and is management-intensive.
- Time- and resource-consuming data staging and manually ingesting large-scale data sets into a separate Hadoop environment can lead to significant delays in reaping benefits or insight from the Hadoop analytics effort.
- Lack of multi-tenancy: traditional Hadoop architectures are isolated on specifically allocated hardware.
- Lack of enterprise-level data protection options including snapshot backup and recovery and data replication capabilities for disaster recovery purposes.

EMC Data Lake Isilon Storage Solution

The EMC Data Lake Isilon network-attached storage (NAS) platform addresses all of the IT challenges of the traditional Hadoop data lake. Powered by the distributed EMC Isilon OneFS® operating system, an EMC Data Lake Isilon cluster delivers a scalable pool of storage with a global namespace, advanced and efficient data protection, and unmatched flexibility not available in a traditional scale-out Hadoop architecture.

Data Protection and Efficient use of Resources

For data protection and high availability, traditional Hadoop DAS architectures require data replication of three times or more, and the Hadoop distribution contains no native replication. An EMC Isilon cluster, with the OneFS file system, uses erasure coding to protect data with greater than 80 percent storage efficiency, in contrast to traditional Hadoop distributed file system (HDFS) with 33 percent storage efficiency. Additionally, applying Isilon's SmartDedupe can further dedupe data in the data lake, making HDFS storage even more efficient. The highly resilient Isilon OneFS filesystem completely eliminates the single-point-of-failure risk with traditional Hadoop deployments.

The EMC Data Lake Isilon Storage allows for processing queries in parallel, one or more segment instances and dedicated CPU, memory and disk (at the Isilon compute layer).

Isilon solutions also offer robust security options including role-based access control (RBAC), SEC 17a-4-compliant, write-once-read-many (WORM) data protection, file system auditing, and data-at-rest encryption to help organizations address regulatory and compliance requirements.

Unmatched Flexibility within the Data Lake

The EMC Data Lake Isilon Storage solution supports multiple instances of Apache Hadoop distributions from different vendors simultaneously. The solution also supports multiple versions of the Hadoop distributed file system (HDFS). This allows organizations to use the specific analytics tools from any Hadoop distribution.

Lower Costs

Isilon's native HDFS integration means organizations can avoid the need to invest in a separate Hadoop infrastructure and eliminate the time and expense of moving large data sets. The unmatched efficiency and ease of use of Isilon solutions helps lower storage costs and simplify management.

Challenges of Developing Business Analytics on HDFS

Hadoop is a technology well-suited to tackling big data problems. The fundamental principle of the Hadoop architecture is to bring analysis to the data, rather than moving the data to a system that can analyze it. Instead of carefully filtering and transforming the source data through a heavy ETL process (for instance, the rows of a relational database) the original data source, with few or no modifications, is ingested into a central repository.

The Challenge of Tradition Hadoop analytics

There are two key technologies that allow users of Hadoop to successfully retain and analyze data: HDFS and MapReduce.
HDFS is a simple but extremely powerful distributed file system that stores data reliably at significant scale. MapReduce is parallel programming framework that integrates with HDFS. It allows users to express data analysis algorithms in terms of a small number of

functions and operators: chiefly, a map function and a reduce function. While MapReduce is convenient when performing scheduled analysis or manipulation of data stored on HDFS, it is not suitable for interactive use as it requires extensive Java programming and developers who understand how MapReduce works to solve data problems analytically.

Why Hive and Impala are insufficient

With the limitations of MapReduce, companies turned to Apache Hive and Cloudera Impala for solutions. These are SQL-like query engines that compile a limited dialect to MapReduce. While they do address some of the expressive shortcomings of MapReduce, performance remains an issue. Furthermore and most importantly, Impala and Apache Hive do not support all 99 of the standard TPC-DS queries.

The Advantages of Pivotal HDB

Pivotal HDB supports low-latency analytic SQL queries, coupled with massively parallel machine-learning capabilities, to shorten data-driven innovation cycles for the enterprise. Pivotal HDB enables discovery-based analysis of large data sets and rapid, iterative development of data analytics applications that apply deep machine learning. Leveraging Pivotal's parallel database technology, it consistently performs tens to hundreds of times faster than all other Hadoop query engines in the market.

Massively Parallel Processing (MPP) with the Data Lake

MPP Systems are distributed systems consisting of a cluster of independent nodes. MPP databases are excellent at structured data processing due to the architecture's fast query processing capability, embedded automatic query optimization, and industry standard SQL interfaces with BI tools. Pivotal HDB incorporates the performance advantages of MPP SQL engine into the Hadoop HDFS central repository environment, allowing customers the processing power of a relational database within a Hadoop repository without the strict schema requirements.

Exceptional Hadoop Native SQL Performance

Pivotal HDB provides superior performance compared to current open source SQL on Hadoop analytical tools. It can achieve fast performance for complex and advanced data analytics. With high speed interconnect for continuous pipelining of data processing, Pivotal HDB can produce near real-time latency.

- Pivotal HDB runs the set of TPC-DS queries roughly twice as fast as Impala. By design, Pivotal HDB also finishes most queries in a fraction of time it would take in Apache Hive.
- Pivotal HDB beats Impala by overall 454 percent in performance (as determined by measuring and comparing the geometric mean across the TPC queries).
- Compared to Hive, Pivotal HDB improves performance by an additional of 344 percent on complex queries.

Full ANSI SQL Compliance

Data consistency is a real issue for application developers. Pivotal HDB is fully ANSI SQL-92, SQL-99, SQL-2003 compliant, including OLAP extensions. Using Pivotal HDB, organizations can leverage existing SQL skills to ramp up and execute quickly within the Hadoop data lake. Additionally, there are no compatibility risks to SQL developers or SQL BI tools and applications. With full support for query roll-ups, dynamic partitions, and joins, enterprises can produce business-critical class analytics directly and natively within HDFS.

External Tables - Pivotal Extension Framework (PXF)

PXF enables SQL querying on data in the Hadoop components such as HBase, Hive, and any other distributed data file types. These queries execute in a single, zero-materialization and fully parallel workflow. PXF also uses the Pivotal HDB advanced query optimizer and executor to run analytics on these external data sources. PXF connects Hadoop-based components to facilitate data joins, such as between Pivotal HDB tables and HBase tables. PXF integrates seamlessly when using Isilon as the HDFS storage layer, and is the fastest way to load data or explore large datasets. PXF also integrates with HCatalog to query Hive tables directly. Additionally, users have added flexibility to HBase or Avro data files through the use of Pivotal HDB PxF services.

Integrated MADlib Machine-Learning

Machine learning algorithms allow organizations to identify patterns and trends in their big data datasets, and also enable them to make high-value predictions that can guide better decisions and smart actions in near real-time and without human intervention. Apache MADlib is an open source library for scalable in-database analytics. It provides data-parallel implementations of machine learning, mathematical, and statistical methods and is fully integrated with in the Apache HDB Hadoop Native SQL platform. MADlib uses the MPP architecture's full compute power to process very large data sets, whereas other products are limited by the amount of data that can be loaded into memory on a single node.

Pivotal HDB with Isilon: A Winning Combination

This comprehensive enterprise data lake business analytics platform provides a scalable, shared data storage infrastructure. Also, it has multiple protocol access to shared data and enterprise features like backup, replication, snapshots, file system audits, and data-at-rest encryption. Combined with ANSI compliant SQL and integrated machine learning, there are some unique capabilities and advantages over direct attached storage (DAS)which are outlined in the table below.

Table 1: Traditional Hadoop Environment vs EMC Isilon with Pivotal HDB

	Traditional DAS Hadoop Distributions with Hive	Hadoop with Pivotal HDB on Isilon
Data Type	**Files**	**Files**
Protection Overhead	**200%**	**20%**
NameNode Redundancy	**Active/Passive**	**Node-to-Node**
De-Duplication	x	□
File Edit Capability	x	□
NFS (v3,v4)/SMB/HTTP/FTP	x	□
Simultaneous Multi-Protocol	x	□
Independent Scaling (Storage/Compute)	x	□
File/Object Level Access Control Lists	x	□
Multi Tenancy	x	□
Simultaneous Hadoop Distributions	x	□
Hadoop Distributions	**1**	**A□□**
Data Tiering	x	□
WORM (SEC 17a-4)	x	□
POSIX Compliance	x	□
Snapshots	□	□
Encryption	x	□
ANSI SQL-92, SQL-99, SQL-2003 compliance	x	□
Integrated MADlib Machine Learning	x	□
Fully ACID compliant	**Partially**	□

Gaining the most from Pivotal HDB and EMC Isilon HDFS

To help our customers and partners get the most out of this proven solution, Pivotal's Platform Engineering team has tested all the common Pivotal HDB functionality used by customers with EMC Isilon as the HDFS layer. This section provides a description of the key findings, and lists some parameters used during the testing effort.

Environment and Test Overview

The test environment consisted of an EMC Isilon scale-out NAS platform with 16 commodity server nodes, 8 storage nodes and archetypical compute, memory, network, and disk equipment. The test used the industry standard 99 queries for TPC-DS benchmarking.

NOTE: Testing was performed by Pivotal only. The testing performed and results collected were not audited by or submitted to TPC.

Isilon Configuration

This section details the configuration parameters used to optimize the EMC Isilon X-series cluster utilized in the testing effort.

File Pool Data Access Patterns

A file pool policy is a way to assign filtering rules and operations that are both system- and user-defined. These data access settings are configured to optimize data access for the type of application accessing it. Protection settings and I/O optimization settings for file types can be set based on the specifications of the file pool filtering policy. EMC Isilon has three different access patterns that are available at a file pool level: concurrent, streaming and random. Concurrent optimizes for current load on the cluster featuring many simultaneous clients; streaming optimizes for high-speed streaming of a single file; and random optimizes for unpredictable access to the file with no cache prefetching.

Multiple iterations of the TPCDS query set were run and the best results were produced using **Streaming** as the access pattern on **Isilon with Pivotal HDB.**

Block Size

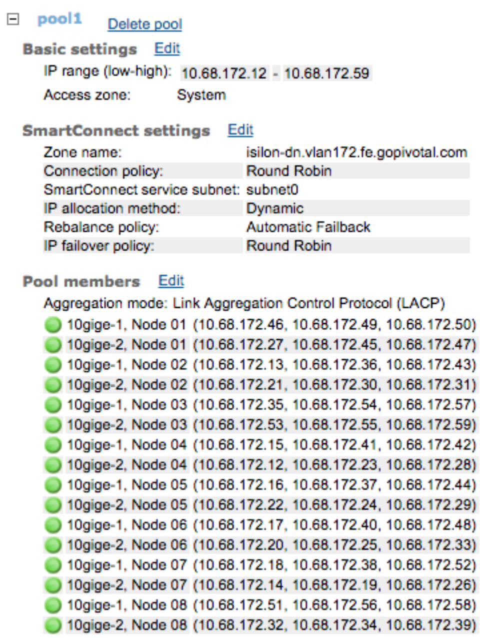
When selecting the correct Block Size for Isilon, Pivotal Hadoop Distribution (PHD) and HDB, it is important to set Isilon's HDFS (isi_hdfs_d daemon), HDFS on the Pivotal HD cluster, and the block size for Pivotal HDB to the same parameters.

Best query runtime results were achieved when block sizes were set to **128MB** or **512MB**. For Pivotal HD, Apache Ambari admin UI can be used to make this change. For EMC Isilon, this is a change that can only be applied using the CLI.

Balancing Traffic

In order to get a balanced throughput out of the Isilon cluster, separate the Isilon cluster into two IP Pools—one for DataNode traffic and the other for NameNode Traffic. For the DataNode IP Pool, configure three IP addresses per network interface in each Isilon node as shown from the EMC Isilon OneFS Management UI below:

Figure 1: EMC Isilon OneFS Management UI: IP configuration



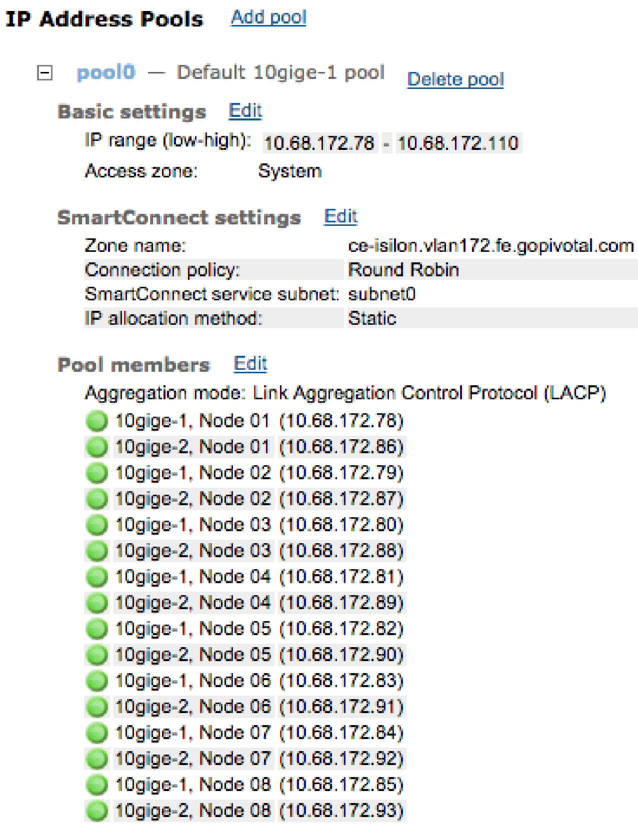
Round robin is recommended for the pool connection policy. The IP allocation method needs to be static —in case of Isilon node or network interface failures, another Isilon node can service requests to the failed DataNode IP address in the pool, and this provides redundancy and load balancing at the DataNode level. Using the isi hdfs racks command, you can create a rack that will now use this new pool as its DataNodes as shown below:

isi hdfs racks create --client-ip-ranges= --ip-pools=pool1 --rack=/rack0

In the above command, replace with the actual IP range of the compute environment.
A separate IP pool to be used for NameNode traffic also needs to be created for each 10Gb Network interface per Isilon node in the

Isilon cluster as shown below:

Figure 2: EMC Isilon OneFS Management UI: IP Address Pools



The connection policy for the NameNode pool needs to be round robin so that each request goes to a different NameNode IP address. The IP allocation for the NameNode pool should be dynamic—as SmartConnect itself will provide the redundancy needed in case of Node or network interface failures.

Pivotal HDB Configuration Parameters

This section details the configuration parameters used to optimize Pivotal HDB for the testing effort.

Data Distribution

Pivotal HDB 2.0 supports distributing table data either randomly or by a specific key. We found that selecting **RANDOM** distribution produced the best results when running Pivotal HDB 2.0 with EMC Isilon as its HDFS storage layer. Additionally, the truncate functionality works as expected in Pivotal HDB 2.0.

Data Format

Pivotal HDB supports multiple formats for creating table data. We compared Append Only tables with Parquet file tables and found that **Parquet tables with snappy compression** produced the best results when running Pivotal HDB with Isilon as its HDFS storage layer. Additionally we verified that the truncate functionality works in Pivotal HDB as expected when running with Isilon as its HDFS storage layer.

Table Partitions

Partitioning data in Pivotal HDB has significant advantages. With it, you can look only at specific partitions of data for rows or columns, and you do not have to incur the expensive scan on the entire table when looking for specific values in a large table. Partitioning improves runtimes when EMC Isilon is the HDFS storage layer. We compared partitioning large tables annually, quarterly, monthly, weekly, and daily in our test scenarios. We found that partitioning large tables by either **quarterly partitions (partitioned every 90 days) or monthly partitions (partitioned every 30 days)** would isolate table scans to specific partitions, reduce the amount of data scanned, and speed up requested results.

Concurrency

The testing effort covered the entire 99 TPC-DS query test suite sequentially and monitored it comparatively when 5, 10, 15, or 20 users were running concurrently. Linear growth in the runtimes occurred as users were added. Pivotal HDB in the EMC Data Lake Isilon configuration was able to distribute the TPC-DS query load evenly across the resources available without running out of resources.

To fine-tune the environment further, Pivotal HDB can leverage resource queues at the group or user level. Depending on workload, additional resources can be granted to specific users and groups to schedule resources efficiently.

Below are the Pivotal HDB and Isilon configuration parameters that produced optimum results.

Table 2: Summary: EMC Isilon and Pivotal HDB tuning recommendations

Component	Configuration	Action	Notes
Isilon	Access Patterns	`isi file pool default-policy modify --data-access-pattern=streaming`	Streaming
Isilon	Block Size	`isi hdfs settings modify --default-block-size=128M`	128M or 512M
Pivotal HDB	Block Size	Ambari UI: edit the HDFS block-size	Match Isilon "Block Size"
Pivotal HDB	Block Size	Edit dfs.default.blocksize property in `/usr/local/HDB/etc/hdfs-client.xml`	Match Isilon "Block Size"
Isilon	Thread Count	`isi hdfs settings modify --server-threads=auto`	Auto or 64
Isilon	Data Node Traffic	OneFS Management UI	3 IP addresses per node load balanced Round Robin; 1 IP per interface; IP Allocation: Static
Isilon	Name Node Traffic	OneFS Management UI	Load balanced Round Robin; 3 IPs per interface; IP Allocation Dynamic

Pivotal HDB	Data Format	Example:
		`CREATE TABLE tpch.region`
		`WITH`
		`(appendonly=true,orientation=parquet,`
		`compresstype=snappy) AS`
		`SELECT * FROM ext_tpch.region`
		`DISTRIBUTED BY (R_REGIONKEY);`
			Parquet with Snappy Compression

Pivotal HDB	Partitioning	Example:
		`CREATE TABLE p_orders (LIKE orders)`
		`WITH (appendonly=true,`
		`compresstype=snappy,`
		`orientation=parquet)`
		`DISTRIBUTED BY (o_orderkey)`
		`PARTITION BY RANGE (o_orderdate)`
		`(START ('1992-01-01') END ('1999-01-01')`
		`EVERY (interval '1 year'));`
			More partitions equate performance but diminished load time to faster query

Lab Testing Environment

The lab environment consisted of eight compute servers (two masters and six workers), ten 900 GB SAS drives for local scratch/spill space, 64GB of memory in each node, and Pivotal HDB version 2.0 on Pivotal HD 3.0. Pivotal HDB was configured to use spill space on each of its local drives and to have ten Pivotal HDB segment instances per node. The configuration options included random access for distribution tables, parquet data format, snappy compression, and monthly partitioning.

Networking was configured to be a 10GB storage network where both the compute and Isilon cluster connected to the same switch. The compute rack has two 10Gb internal switches, which were configured with a MLAG configuration using two ports for the internal switch to the storage network. This configuration allows the Isilon cluster to be a shared resource on the 10GB network. Data can be accessed it via HDFS or any other Isilon supported protocol.

Storage was provided by four 410 nodes in the Isilon cluster with two 800GB SSDs and thirty-four 1TB SATA drives. The software version on the Isilon cluster was OneFS v8.0. The Isilon storage configuration utilized SSDs for data and metadata. The Name Node pool was set to dynamic, Data Node Pool was set to static, and streaming access was enabled.

Summary

The advantages of using Pivotal HDB within the EMC Data Lake Isilon Storage Solution are numerous. From fast data ingestion with EMC Data Lake Isilon Storage’s native HDFS to rapid SQL development with Pivotal HDB, the list of benefits to the big data lake environment is impressive. Enterprises can leverage existing SQL skillsets and avoid additional training with the fully ANSI SQL complaint analytic environment. Organizations gain the full advantages of massive data lake scalability, Hadoop distribution flexibility, and data protection and encryption of all their big data assets.