TDWI RESEARCH

TDWI CHECKLIST REPORT

WHERE HADOOP FITS IN YOUR DATA WAREHOUSE ARCHITECTURE

By Philip Russom

Sponsored by

TERADATA. THE BEST DECISION POSSIBLE™

tdwi.org

tdwi

JUNE 2013

TDWI CHECKLIST REPORT

WHERE HADOOP FITS IN YOUR DATA WAREHOUSE ARCHITECTURE

By Philip Russom

TABLE OF CONTENTS

1201 Monster Road SW, Suite 250
Renton, WA 98057

T 425.277.9126
F 425.687.2842
E info@tdwi.org

tdwi.org

FOREWORD

Business intelligence (BI) and data warehousing (DW) professionals are aware of Hadoop's general benefits for BI and DW. They even know that Hadoop's most credible use cases focus on analytics and managing multi-structured and no-schema data. This is good news for the integration of Hadoop into mainstream BI and DW practices. However, the Hadoop ecosystem of products is still quite new, so most BI and DW professionals haven't yet determined where to begin their integration of Hadoop with established platforms for BI, DW, data integration (DI), and analytics.

The first step toward successful integration is to determine where Hadoop fits in your data warehouse architecture. Hadoop is a family of products, each with multiple capabilities, so there are multiple areas in data warehouse architectures where Hadoop products can contribute. At the moment, Hadoop seems most compelling as a data platform for capturing and storing big data within an extended DW environment, plus processing that data for analytic purposes on other platforms.

This TDWI Checklist Report discusses adjustments to DW architectures that real-world organizations are making today, so that Hadoop can help the DW environment satisfy new business requirements for big data management and big data analytics.

NUMBER ONE
ROLES FOR HADOOP IN DW ARCHITECTURES

THE HADOOP ECOSYSTEM

Apache Hadoop is an open source software project administered by the Apache Software Foundation (ASF). The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers, each offering local computation and storage.

The Hadoop family of products includes the Hadoop Distributed File System (HDFS), MapReduce, Pig, Hive, HBase, Mahout, Cassandra, YARN, Ambari, Avro, Chukwa, and Zookeeper. These products are available as native open source from ASF, as well as from several software vendors. In this report, the term *Hadoop* refers to the entire Hadoop family of products as licensed by Apache.

Note that MapReduce products do not require HDFS and some run atop a relational DBMS. MapReduce is a general-purpose execution engine that handles the complexities of parallel programming for a wide variety of hand-coded logic and other applications, which includes (but is not restricted to) analytics.

HADOOP'S LIMITATIONS

Hadoop is comparable to traditional massively parallel processing relational database management systems (MPP RDBMSs) in that both are highly scalable, shared nothing, parallel environments that support analysis. Yet, the original design of Hadoop by Internet search engine companies had little need for capabilities important to other enterprises:

Database functions. As its name states, HDFS is a distributed file system and therefore lacks capabilities we would associate with a database management system (DBMS), such as indexing, random access to data, support for standard SQL, and query optimization. That's fine, because HDFS does things DBMSs do not do as well, such as managing and processing massive volumes of file-based, unstructured data. For minimal DBMS functionality, users can layer HBase over HDFS, as well as the query framework called Hive.

Low-latency queries. As a high-latency, batch-oriented platform, Hadoop is predicated on scanning all the data in the file system. Hence, the selective random access to data and iterative queries we take for granted with RDBMSs are alien to Hadoop. (HBase is a possible exception, though designed for OLTP, not analytics.) Furthermore, Hadoop lacks mature query optimization and the ability to place "hot" and "cold" data on a variety of storage devices with different levels of performance. For these features, the emerging best practice is to process data on Hadoop and move the results to an RDBMS.

Ease of access. Hadoop evolved at Internet firms in cultures that encourage hacking and hand coding. Hive has certainly improved the ease of data access, but the lack of ANSI SQL support limits the range of users, tools, and applications that can access Hadoop data.

Data integration. Hadoop was built for capturing and analyzing Web data, and it has been stretched to handle other non-traditional data domains. Hence, Hadoop is not efficient at supporting joins and complex SQL, but it can be programmed for other data transformations.

Data integrity. The ACID properties (atomicity, consistency, isolation, durability) guarantee that database transactions are processed with integrity. Hadoop is not a DBMS, so it is not ACID compliant, and therefore is not appropriate where inserts and updates are required.

Fine-grained security. Hadoop lacks security features common in RDBMSs, such as row- and column-level security, and it lacks mature user-level controls, authentication options, and encryption.

PROMISING USES OF HADOOP IN DW CONTEXTS

Due to their new capabilities and low cost, HDFS and other Hadoop products show great promise for transforming some areas within the data platform landscape:

Data staging. A considerable amount of data is processed in a DW's staging area to prepare source data for specific uses (reporting, analytics) and for loading into specific databases (DWs, marts). Much of this processing is done by homegrown or tool-based solutions for extract, transform, and load (ETL). Hadoop allows organizations to deploy an extremely scalable and economical ETL environment.

Data archiving. Traditionally, enterprises had three options when it came to archiving data: leave it within a relational database, move it to tape, or delete it. Hadoop's scalability and low cost enable organizations to keep all data forever in a readily accessible online environment.

Schema flexibility. Relational DBMSs are well equipped for highly structured data (from ERP, CRM) and stable semi-structured data (XML, JSON). As a complement, Hadoop can quickly and easily ingest any data format, including evolving schema (as in A/B and multivariate tests on a website) and no schema (audio, video, images).

Processing flexibility. Hadoop's NoSQL approach is a more natural framework for manipulating non-traditional data types and enabling procedural processing valuable in use cases such as time-series analysis and gap recognition. Hadoop supports a variety of programming languages, thus providing more capabilities than SQL alone. In addition, Hadoop enables the growing practice of "late binding"—instead of transforming data as it's ingested by Hadoop, structure is applied at runtime.

NUMBER TWO
TRENDS IN DATA WAREHOUSE ARCHITECTURES

For decades, data warehousing relied almost exclusively on relational methods, as seen in RDBMSs and ANSI SQL. These are still valuable, so they are not going away. The trend is that relational methods and platforms are being complemented by new ones, such as Hadoop and other NoSQL approaches. As organizations embrace big data and other new sources of data, they need new, non-relational data platforms that are designed and optimized for the new forms of data. In similar trends, new platform types (both relational and non-relational) are useful for the growing number of workloads in analytics and real-time operation.

Workload-centric DW architecture. One way to measure a data warehouse's architecture is to count the number of workloads it supports. As seen in Figure 1, a little more than half of user organizations surveyed support only the most common workloads, namely those for standard reports, performance management, and online analytic processing (OLAP). The other half also supports workloads for advanced analytics, detailed source data, and real-time data feeds. The trend is toward the latter. In other words, the number and diversity of DW workloads is increasing, because organizations are embracing big data, multi-structured data, real-time or streaming data, and data processing for advanced analytics.

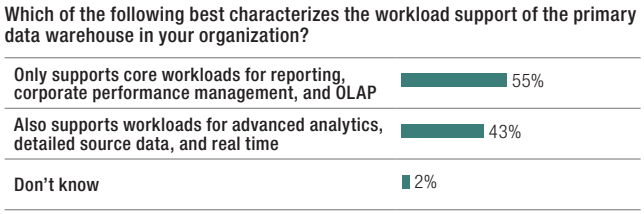

Figure 1. *Based on 114 respondents in 2012.*[1]

Distributed DW architecture. The issue in a multi-workload environment is whether a single-platform data warehouse can be designed and optimized such that all workloads run optimally, even when concurrent. More DW teams are concluding that a multi-platform data warehouse environment is more cost-effective and flexible. Plus, some workloads receive better optimization when moved to a platform beside the data warehouse. In reaction, many organizations now maintain a core DW platform for traditional workloads but offload other workloads to other platforms. For example, data and processing for SQL-based analytics are regularly offloaded to DW appliances and columnar DBMSs. A few teams offload workloads for big data and advanced analytics to HDFS,

[1] See the discussion around Figure 8 in the 2012 TDWI Best Practices Report *High-Performance Data Warehousing*, available free on tdwi.org.

discovery platforms, MapReduce, and similar platforms. The result is a strong trend toward distributed DW architectures, where many areas of the logical DW architecture are physically deployed on standalone platforms instead of the core DW platform.

From the EDW to the multi-platform unified data architecture. A consequence of the workload-centric approach (coupled with a reassessment of DW economics) is a trend away from the single-platform monolith of the enterprise data warehouse (EDW) and toward a physically distributed unified data architecture (UDA).[2] A modern UDA is a logical design that assumes deployment onto multiple platform types, ranging from the traditional warehouse (and its satellite systems for marts and ODSs) to new platforms such as DW appliances, columnar DBMSs, NoSQL databases, MapReduce tools, and even a file system on steroids such as HDFS. The multi-platform approach of UDA adds more complexity to the DW environment; yet, the complexity is being addressed by vendor R&D to abstract the complexity and take advantage of the various capability and cost options.

A data warehouse benefits from a reference architecture. Building a data warehouse based on UDA involves expanding your portfolio of enterprise software to include multiple DBMSs and other workload-specific data management platforms. That's not enough, though. Without a strong data warehouse architecture, the complexity of the environment could drive it into chaos. For example, DW professionals must design a logical architecture with compartmentalized areas (for data staging, archiving, data types, analytic processing, domains, marts, dimensions, cubes, time series). This way, areas of the architecture will translate well into distributed physical deployments. Moving data around is inevitable in a multi-platform UDA, so there needs to be a well-defined data integration architecture as well. Even so, an assumption behind UDA is that data structures and their deployment platforms will integrate on the fly (due to the exploratory nature of analytics) in a loosely coupled fashion. So the architecture should define data standards, preferred interfaces, and shared business rules to give loose coupling consistent usage.

Vendor contributions to new DW architectures:

- BI vendors are integrating with Hadoop
- MPP RDBMS vendors are integrating with Hadoop to provide greater abstraction and productivity tools
- MPP RDBMS vendors are borrowing technology from Hadoop (MapReduce, late-binding) to bring big data features together with the advantages of an RDBMS

NUMBER THREE
DATA STAGING

The big data phenomenon is driving many organizations to restructure the data staging areas within their DW architectures to cope with data's increasing volume, sources, types, structures, processing requirements, and feed speeds.

Consider HDFS as a bigger and better platform for some kinds of data staging. Most organizations that perform hefty data processing in and around a data staging area have already separated staging from the warehouse proper. When the staging area has its own DBMS instances and other data platforms (such as HDFS), it provides options to pair the ETL job with the right level of platform features and economics. As an example, offloading workloads to the staging area frees resources on the core DW so it can take on more analytically oriented workloads.

Hadoop has proven to be valuable in staging and refining big data sets, such as Web log data, where the files can be ingested quickly into HDFS, and then cleansed and transformed (sessionized, stripped of superfluous tags) using HiveQL to create a load-ready file for a relational database. Additionally, data types that were once out of reach—such as structured signals from pictures and audio files—can now be processed and loaded into a relational database for integration and fast response times. As another example, an insurance company can detect fraud based on claims reported over the phone by analyzing the client's voice for signs of lying within Hadoop, then integrate with other data (about past case history, size of claim) within a relational environment to further analyze and take action.

Carefully evaluate the best environment for ETL routines. Hadoop is a powerful and economical platform for ETL processing where ETL requirements favor Hadoop's strengths around straightforward calculations and sums, stripping out "noise" data (as is common with data from logs, machines, and streams), and flexible processing to bring structure to no-schema data types. On the flip side, many ETL routines are better suited for transformations within relational environments where the requisite reference data and source data persist, and where joins and complex transformations favor SQL. This is more ELT than ETL, where processing is completed in the target warehouse. As an example, consider a marketing ROI calculation where multiple subject areas (e.g., campaign history, product master, sales, customer master) are all referenced in an integrated fashion to indirectly infer uplift for hundreds of campaigns each day.

[2] *Unified data architecture* is a Teradata term, but it is consistent with concepts TDWI has espoused for years, such as distributed data warehouse architecture and the data warehouse environment. All of these assume an integrated logical design that is made manifest by a multi-platform physical environment.

NUMBER FOUR
DATA ARCHIVING

There are good reasons data warehouse environments need a data archive.

An archive can enable generations of analytic applications. Organizations experienced with analytics know they cannot anticipate all the ways in which they'll need to process data for analytics in the future. Therefore, they prefer to maintain source data in its original form until a new analytic need arises. With the advent of Hadoop, it is both technically and economically feasible to archive raw data, then transform it in new ways as new business analytic requirements arise.

Though DW professionals may not call it an archive, a store of detailed source data strongly resembles an archive, in that data is retained in its original form. Many organizations have stretched their DW and staging areas to accommodate massive stores of detailed source data. They should consider moving such stores to HDFS, which can handle this data at a low price point, while keeping the data live on spinning disks (unlike offline archives) and available to a wide range of tools and other platforms within the DW environment.

Many organizations measure data in years, not terabytes and petabytes. They'd like to have more years of data available for analytics and reporting so they can have a longer view of corporate performance on which to base strategic decisions. They may have legal, regulatory, or contractual obligations that require availability of more years of data than their current systems can support. HDFS, as a live archive, can help users cost-effectively scale to more years of data, not just more terabytes and petabytes.

DW teams need to practice information life cycle management (ILM). That's where data is placed on platforms and media that are consistent with the value of individual data sets. For example, on a modern relational DW platform, a data set that is "hot" (i.e., being accessed often by many users) should be moved to solid-state drives (SSDs) or server memory, where it gets the high performance that its high value merits. At the other extreme, "cold" data sets should be compressed or moved to low-cost media, consistent with their low value. Hadoop's low cost per terabyte makes it an appropriate platform for cold data. Unlike magnetic tape (a common archive medium), data in Hadoop is accessible immediately without a lengthy "restore" process.

NUMBER FIVE
MULTI-STRUCTURED DATA

Build a multi-structured data environment. Disciplines for BI, DW, DI, and analytics need to drive aggressively toward multi-structured data environments where workload-specific platforms can capture, manage, and process the full range of data, from structured and relational data to evolving and no-schema data types. Multi-structured data environments are required if organizations are to get business value from the full range of available data. After all, limiting the range of data sources can limit desirable business goals. Historically, data warehouses focused on "transactions" (e.g., customer accounts, orders, supply chain, marketing communications, customer service calls) that can now be augmented with "interactions" (e.g., movement within a website, activity within an online game, health vital signs) to provide a more complete and valuable view of a business.

Fill the non-structured data gaps with Hadoop. Hadoop is a very efficient refinery for bringing structure to and culling signals from large and messy data sets. When these valuable signals from big data sets are cleansed and integrated with other corporate data assets in a high-performing and easy-to-use environment, value is created by businesses with the culture and people who are equipped to exploit it.

There are, however, cases of value creation where the analysis is done wholly within the Hadoop environment, when little value is gained from integration and reuse, and huge value is gained from deep exploration and flexibility that result in the desired output. Primarily, these use cases involve analyzing provisional data sets where the primary value of the data is not derived through integration with other data sets.

This doesn't mean Hadoop is the best place for all data or that it's a credible replacement for a data warehouse. The modern data warehouse environment includes multiple platforms so that diverse data can find the best home based on storage, processing, and budgetary requirements. Relational DBMSs are still the best home for integrated, cleansed, and reused data, where low latency and service-level agreements are important.

NUMBER SIX
ADVANCED ANALYTICS

Many organizations rely heavily on SQL as the primary approach to advanced analytics. After all, BI professionals know standard SQL, and almost all tools and DBMSs support it. In a practice that TDWI calls *extreme SQL*, an experienced BI professional iteratively creates complex SQL programs, and these work well with big data that's SQL addressable. Extreme SQL is typically applied to highly detailed source data, still in its original schema (or lightly transformed). The SQL is "extreme" because it's creating multidimensional structures, complex data models, and analytic models on the fly without remodeling and transforming the data in advance. Extreme SQL also enables broad data exploration.

Relational and analytic DBMSs are best for SQL-based analytics. Instead of expecting HDFS and add-on tools such as Apache Hive and HBase to rise to the challenges of extreme SQL, it's best (with today's state of technology) to depend on data platforms and tools within the data warehouse environment that are better suited to extreme SQL, namely relational DBMSs (which are typically at the heart of a core DW) or standalone analytic DBMSs, which may be packaged as data warehouse appliances, columnar DBMSs, analytic accelerators, in-memory DBMSs, multi-tool analytic platforms, and cloud-based or SaaS-based platforms.

To be sure, both MPP RDBMSs and Hadoop support analytics. The modern data warehouse recognizes both as environments for analytics to support in different use cases:

- **Data needs.** As discussed, an MPP RDBMS will contain cleansed and integrated data, whereas a Hadoop cluster will contain raw data in quantity. As such, the data requirements of the end user will naturally dictate the environment in which the analytics is best performed.
- **Analytical tool needs.** Statistical software packages are all-powerful analytical tools that have mature integration with best-in-class MPP RDBMS vendors. Those same packages are becoming prevalent on Hadoop as well, but lack maturity when compared to RDBMS integration at this time. The Apache Mahout project is a collection of open source statistical algorithms that run on top of Hadoop. Additionally, the ability to apply low-level programming languages (such as Java) and MapReduce against Hadoop provides yet another option in constructing an analytical algorithm.
- **Latency needs.** Another key driver for data platforms in analytical use cases is the requirement of query latency. As discussed, MPP RDBMSs provide significant performance and throughput advantages over Hadoop. In some cases, analytics is a series of trials and errors, where low latency is critical to completing the task in a quality manner. In other cases, the length of time in which the analytics is developed or runs is not as critical to other feature benefits within Hadoop.

Manage analytic big data in Hadoop, but take it to more mature tools and platforms for analysis, when appropriate. As more seasoned BI/DW professionals start using Hadoop products, the emerging best practice is to treat HDFS as just another data store—albeit one for multi-structured data. Using a variety of tools (whether from software vendors or open source from Apache), HDFS data is scanned, extracted, and transformed for specific BI and analytic purposes, as you would with any data store, archive, or staging area. Note that the data processed from HDFS travels to other, more mature or workload-specific platforms within the extended data warehouse environment for reporting and analysis.

ABOUT OUR SPONSOR

teradata.com

Teradata is the world's largest company focused on integrated data warehousing, big data analytics and discovery, and business applications. Our powerful solutions are the foundation on which we've built our leadership position in business intelligence and are designed to address any business or technology need for companies of all sizes. With the addition of Teradata Portfolio for Hadoop, Teradata provides the most trusted and flexible path for companies looking to integrate Apache™ Hadoop® as part of their enterprise data architecture.

Only Teradata gives you the ability to integrate your organization's data, optimize your business processes, and accelerate new insights like never before. The power unleashed from your data brings confidence to your organization and inspires leaders to think boldly and act decisively for the best decisions possible. Learn more at teradata.com.

ABOUT THE AUTHOR

Philip Russom is director of TDWI Research for data management and oversees many of TDWI's research-oriented publications, services, and events. He is a well-known figure in data warehousing and business intelligence, having published over 500 research reports, magazine articles, opinion columns, speeches, Webinars, and more. Before joining TDWI in 2005, Russom was an industry analyst covering BI at Forrester Research and Giga Information Group. He also ran his own business as an independent industry analyst and BI consultant and was a contributing editor with leading IT magazines. Before that, Russom worked in technical and marketing positions for various database vendors. You can reach him at prussom@tdwi.org, @prussom on Twitter, and on LinkedIn at linkedin.com/in/philiprussom.

ABOUT TDWI RESEARCH

TDWI Research provides research and advice for business intelligence and data warehousing professionals worldwide. TDWI Research focuses exclusively on BI/DW issues and teams up with industry thought leaders and practitioners to deliver both broad and deep understanding of the business and technical challenges surrounding the deployment and use of business intelligence and data warehousing solutions. TDWI Research offers in-depth research reports, commentary, inquiry services, and topical conferences as well as strategic planning services to user and vendor organizations.

ABOUT THE TDWI CHECKLIST REPORT SERIES

TDWI Checklist Reports provide an overview of success factors for a specific project in business intelligence, data warehousing, or a related data management discipline. Companies may use this overview to get organized before beginning a project or to identify goals and areas of improvement for current projects.