SPECIAL REPORT

WINTER CORPORATION

Big Data

What Does It Really Cost?

The Large Scale Data Management Experts

WinterCorp

V1.08 - 8/5/13

WINTER CORPORATION

Big Data

What Does It Really Cost?

RICHARD WINTER
RICK GILBERT
JUDITH R. DAVIS

For comments & questions on this report,
email: tcod@wintercorp.com

WinterCorp

245 FIRST STREET, SUITE 1800
CAMBRIDGE MA 02145
617-695-1800

visit us at www.wintercorp.com

V1.08 - 8/5/13

SUMMARY

There are two major platform architectures for implementing big data analytics: the data warehouse and Hadoop. A key challenge today is choosing which of these big data solutions to use for a specific analytic application.

The established architecture uses parallel data warehouse technology based on the relational model of data. The organization creates a database to capture, integrate, aggregate and store data from multiple sources for query, reporting, analysis and decision-making purposes. Mature data warehouse architectures perform all data operations in parallel, routinely applying hundreds or thousands of processors at once to speed response time.

The newer architecture employs Apache Hadoop, an open-source file system that supports distributed processing of large datasets across clusters of computer servers. Via MapReduce, a programming framework, Hadoop can process the data in parallel across all of the servers in a cluster. Hadoop supports clusters of virtually any size from a single server to tens of thousands of servers, thus enabling the storage and processing of extremely large volumes of data applying a different approach to the tasks of parallel database management.

Hadoop has garnered much industry attention as a low-cost, open-source alternative to the data warehouse platform. A commonly quoted cost figure for the acquisition of a Hadoop cluster is less than $1,000 US per TB -- several times lower than the average list price for a data warehouse platform.

The central premise of this WinterCorp Report is that the customer will want to estimate a ***total solution cost*** for an analytic data problem. Such an estimate must include not only all system-related costs but also the software related cost of developing the analytic business solution. This include such items as the cost of developing the applications, queries and analytics that use the data for the intended purpose.

Acting on this premise, WinterCorp developed a framework for estimating the ***total cost of data***, or TCOD. The goal is to enable organizations to compare the approximate total costs of different solutions and understand where each big data platform architecture works best.

Business Examples. To illustrate use of the TCOD framework, the present report describes two big data application examples, and shows how each would be implemented using (1) data warehouse technology and (2) Hadoop. The report then calculates the five-year TCOD for each solution based on a set of cost assumptions which are documented in full.

The first example, refining the sensor output of large industrial diesel engines, favors a Hadoop solution. Requirements for this example include 500 TB of stored data and rapid, intensive processing of a small number of closely-related datasets. In addition, the analysis reads the entire dataset, life of the raw data is relatively short, and a small group of experts collaborates on the analysis. The five-year TCODs for the data refining example, summarized in Figure 1, show that Hadoop ($9.5 million) is a far more cost-effective solution than a data warehouse appliance ($30 million).

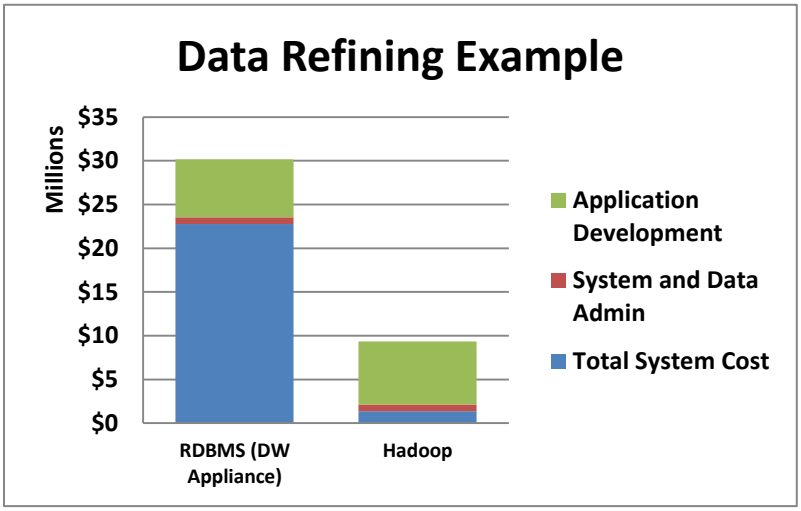


Figure 1. Comparison of 5-year TCODs for engineering example

The system cost for the data warehouse appliance is the dominant factor in this case. Note our inclusive concept of system cost and its breakdown in Figure 1A, where just $5.5 million of the $22.7 million system cost for the data warehouse appliance is incurred in the first year.

	Data Warehouse Appliance	Hadoop
Volume of Data	500 TB	500 TB
System Cost[1]	$22.7	$1.4
Initial Acquisition Cost	$5.5[2]	$0.2[3]
Upgrades at 26% CAGR	$8.4	$0.3
Maintenance/Support[4]	$8.2	$0.2
Power/Space/Cooling	$0.6	$0.7
Admin	$0.8	$0.8
Application Development	$6.6	$7.2
Total Cost of Data	**$30 million**	**$9.3 million**

Table 1A: Cost Breakdown for data refining example[5]

Our second example is an enterprise data warehouse (EDW) in a large financial institution. Here, the analytic requirements favor a data warehouse platform. There are a large number of data sources, users, complex queries, analyses and analytic applications; needs for data integration and integrity; need for reusability and agility to accommodate rapidly changing business requirements and long data life. As shown in Figure 2, the five-year TCODs for a 500 TB EDW show that the data warehouse platform ($265

[1] The "System Cost" figures in are inclusive five year estimates for the total of system acquisition, support/maintenance, upgrades at 26% CAGR in system capacity and power, space and cooling.

[2] Average list price per TB for three widely deployed data warehouse appliances is used for acquisition and upgrade price, reduced by an illustrative discount of 40%, roughly indicative of discounts that are likely to be available with a purchase of approximately 500 TB capacity. An aggressive growth rate, in which system size doubles every three years, is based on the authors' experience with big data programs generating high business value. Note that some vendors offer data warehouse appliances at lower performance and price points.

[3] The widely estimated $1000/TB for acquisition of a Hadoop do-it-yourself cluster is used here, with an illustrative vendor discount of 10% for the relatively large purchase of 500TB capacity.

[4] Annual maintenance and/or support for a data warehouse appliance is estimated at 20% of discounted (acquisition+upgrade) price for the data warehouse appliance. This factor is estimated for years 2-5 as it is usually included in the acquisition price for year 1.

[5] Due to rounding, not all totals tie in. Download the spreadsheet at www.wintercorp.com/bigdata-spreadsheet for additional precision.

million) is far more cost-effective than a Hadoop solution ($740 million) in this example, even though the system cost of the data warehouse platform is higher than that of the Hadoop platform.

.

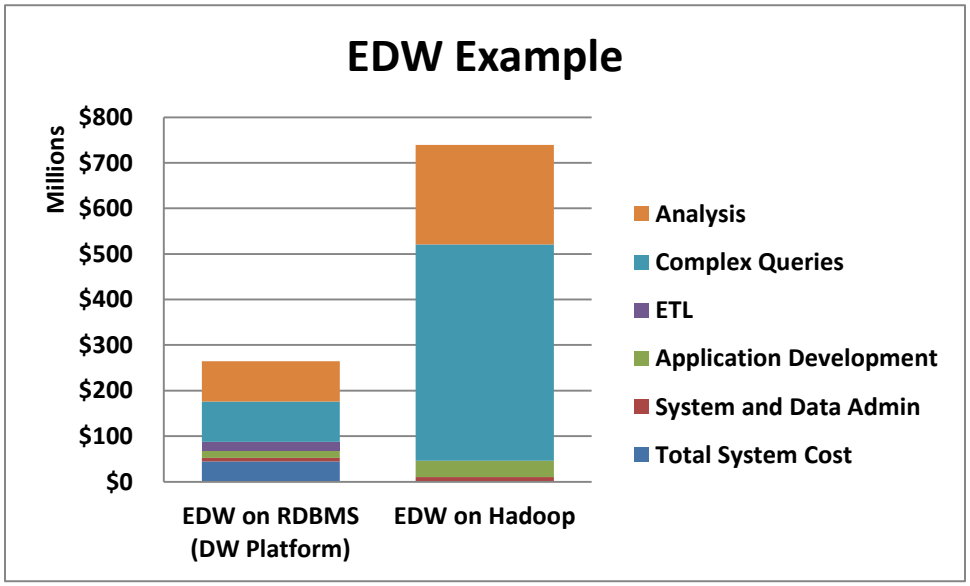


Figure 2. Comparison of 5-year TCODs for EDW example

system cost plays a relatively small part in the total cost picture: system cost is 17% of the total when RBDMS technology is used and less when Hadoop technology is used. As is evident from the bar chart in Figure 2, the development of complex queries and analytics are the dominant cost factors in the example.

Table 2A shows each component of total cost in millions of dollars and breaks system cost down numerically into its major elements. Note that our figure for ***system cost*** is again an inclusive five-year cost figure, and initial acquisition cost plays a yet smaller role than total system cost. Of the $44 million estimated for EDW system cost, $10.8 million is the initial acquisition cost – about 4% of TCOD. While it is common to focus on the first major outlay in the project, the acquisition of a platform -- the total cost of the project is far more important, and other factors greatly outweigh all the system costs combined. Choosing the data warehouse platform in this case lowers the overall cost by a factor of 2.8 times.

	Data Warehouse Platform	Hadoop
Volume of Data	500 TB	500 TB
System Cost[6]	$44.6	$1.4
Initial Acquisition Cost	$10.8[7]	$0.2[8]
Upgrades at 26% CAGR	$16.4	$0.3
Maintenance/Support[9]	$15.9	$0.2
Power/Space/Cooling	$1.5	$0.6
Admin	$7.7	$8.5
Application Development	$16.5	$36
ETL	$18.4	$0
Complex Queries	$88.7	$475
Analysis	$88.7	$219
Total Cost of Data (TCOD)	**$265 million**	**$740 million**

Table 2A: Cost Breakdown for EDW example[10]

Sensitivity analysis confirms that these results in the EDW example are valid even if changes are made to key cost assumptions regarding frequency of data usage, development cost and data volume. For example, we get essentially the same result for an EDW ranging in size from 50 TB to 2 PB.

Business Value. While not the subject of this report, the business value of a big data initiative is always important. Typically, substantial big data investments – like all investments – are made when the expected business value is substantially greater than the expected cost. In this report, we use examples in which the investments are in the

[6] The "System Cost" figures in are inclusive five year estimates for the total of system acquisition, support/maintenance, upgrades at 26% CAGR in system capacity and power, space and cooling.

[7] Average list price per TB for three widely deployed enterprise data warehouse platforms is used for acquisition and upgrade price, reduced by an illustrative discount of 40%, roughly indicative of discounts that are likely to be available with a purchase of approximately 500 TB of user data capacity. An aggressive growth rate, in which system size doubles every three years, is based on the authors' experience with big data programs generating high business value. Note that some vendors offer data warehouse appliances at lower performance and price points.

[8] The widely estimated $1000/TB for acquisition of a Hadoop do-it-yourself cluster is used here, with an illustrative vendor discount of 10% for the relatively large purchase of 500TB of user data capacity.

[9] Annual maintenance and/or support for a data warehouse appliance is estimated at 20% of discounted (acquisition+upgrade) price for the data warehouse appliance. This factor is estimated for years 2-5 as it is usually included in the acquisition price for year 1.

[10] Due to rounding, not all totals tie in. Download the spreadsheet at www.wintercorp.com/bigdata-spreadsheet for additional precision.

millions of dollars over five years. While the total costs in the examples are large, readers are urged to bear in mind four points in examining the examples:

a. **The total cost figures in our illustrations may be unfamiliar.** In their daily work, most executives do not perceive the total cost estimates we present because such costs as the query and analysis of data are spread out over multiple departments and budgets. Nonetheless, we believe the costs are real and are representative of business initiatives that leverage data and analytics for strategic advantage.
b. **We believe that the investments we describe in the examples are made because they do indeed produce returns in proportion to their scale.** The extraordinary successes in recent years of analytic business strategies bear this out. We encounter many situations in our work with clients in which big data investments produce returns valued in the billions of dollars in the course of a few years.
c. Choosing the most appropriate big data platform for an initiative, as illustrated in our examples can **reduce the real total cost by a factor of three** or more, thus having a large impact on ROI. **The best choice can also reduce business risk and time to value;** this is all the more important when critical business problems and opportunities are at hand;
d. We have chosen to illustrate the framework with large scale examples, where data volumes start out in the hundreds of terabytes and grow rapidly. However**, the same principles and tradeoffs apply at smaller scale.** The sensitivity analyses in Section 4 of the report demonstrate this point.

Conclusions. The TCOD cost analyses presented in full in the text of this report highlight important guidelines for selecting a big data analytic solution:

- Carefully identify and evaluate the analytic and data management requirements of each application before choosing a big data platform.

- Systematically evaluate costs according to well-defined framework, such as the one presented here.
- While initial platform acquisition cost can be a consideration, do not underestimate or ignore the real, long-term costs of a big data solution. **Businesses win by optimizing total cost, risk and time to value.**
- Employ a flexible analytic architecture that implements both data warehouse technology and Hadoop in order to take advantage of the strengths of each.

Effective analytics have become a critical success factor for every organization. Rapidly increasing data volumes and analytic workloads make it more important than ever that all enterprises consider the economics of any large-scale initiative. The TCOD framework will help guide users of data and analytics in making effective choices as they develop big data architectures and solutions.

Contents

1 What is Big Data? 1
1.1 Overview 1
1.2 The Data Warehouse Platform for Big Data 3
1.3 The Hadoop Platform for Big Data 5
1.4 The Total Cost Concept 7
2 Total Cost of Data (TCOD) Framework 9
2.1 Overview 9
2.2 Using the TCOD Framework 11
3 Big Data Examples 14
3.1 Engineering Analytic Example: Data Refining 14
3.2 Business Example: Financial Enterprise Data Warehouse 18
3.3 Comparison of Big Data Solutions 26
4 TCOD Sensitivity Analysis 30
4.1 Frequency of Data Use 30
4.2 Cost of Development in Hadoop Environment 31
4.3 Data Volume 33
5 Conclusions 35
Appendix: Additional Cost Factors To Consider 37

1 What is Big Data?

1.1 Overview

The extraordinary growth of data volumes and data variety in recent years has given rise to the widely discussed phenomenon of "big data," presenting new challenges and new business opportunities for virtually every enterprise.

In addition to large and rapidly growing data volumes, there are three key technical aspects to the big data phenomenon. ***First, we live in an increasingly data-intensive digital world***. Devices and machines that create and/or transmit digital data are everywhere around us, permeating the world in which we live and work. These include mobile devices, such as laptops, smartphones, tablets, GPSs, medical implants, cars, airplanes, barcode scanners, etc.; and fixed devices such as sensors in machines and buildings, and cameras on roadways. As these data-intensive devices decline in price, there will be ever more of them. And each class of device will incorporate more computing power each year, enabling it to gather or generate and transmit yet more information. At some point in the future, nearly every manmade object will contain a device that transmits data.

Second, much of the data stored online and analyzed today is more varied than the data stored in recent decades. Until recently, most enterprises could only afford to retain, organize and analyze the tabular data associated with transactions, customers and suppliers and closely-related business operations. But the accumulated technology changes of the past decade have made it practical to capture and retain digital information that is far more varied and complex than in the past. The increasing reliance on the Internet for communication, research, online shopping, social interaction and other daily uses creates an incentive for organizations to analyze the billions of web clicks, emails, tweets and blogs plus photos, images, videos and other complex types of data generated by the Internet and the digital devices described above.

Third, more of our data arrives in near-real time. Because most data now is generated on or by electronic devices and transmitted immediately, the data is increasingly "fresh" in terms of its context and value. In today's environment, data often describes what happened in the last second or minute, not what happened yesterday, or last week. Mobile devices contribute to this with the ability to identify who is where and when in real time. An example is the potential to know right now that a prospective customer is outside one of your stores. In fact, when a customer stops to look at a sign outside the store or at a display in the store window, it may be possible to identify the specific item on which the customer is focused. As a result, you have the opportunity **for a moment** to give that customer an incentive to step inside the store, and the incentive can be related to the customer's interest at that moment.

Organizations want to take advantage of this growing big data environment with analytic solutions that can derive new information about key business entities (customers, products, suppliers, manufacturing processes etc.), enhance the value of data already gathered and lead to a more complete and comprehensive view of the business. In the case of big data, the elephant in the room, so to speak, is Hadoop, the open source platform for big data storage and processing. The issue is when to use Hadoop and when to use data warehouse technology for a specific big data analytic solution. Each platform has advantages that make it best suited for some situations.

For many enterprise executives and strategists, big data now suggests a large business opportunity. They see exciting, new opportunities tempered by the continuing challenge to design a cost-effective solution that meets critical and frequently changing business requirements. The benefit of success is more effective business decisions and, ultimately, increased agility and competitiveness.

This WinterCorp Report briefly describes the two major platform architectures for implementing big data analytics: data warehouse technology and Hadoop. Each is distinctive in terms of the time and effort required (1) to set up the analytic environment and (2) to support ongoing usage over time.

The report also presents a framework for assessing the total cost of each platform, illustrates how to apply the framework in two application examples and summarizes guidelines for choosing a big data solution.

1.2 The Data Warehouse Platform for Big Data

The established platform architecture for big data analytics is parallel data warehouse technology, a broad category that includes enterprise data warehouses, data marts and analytic databases. The organization creates a database to capture, integrate, aggregate and store data from multiple sources for query, reporting, analysis and decision-making purposes (Figure 3).

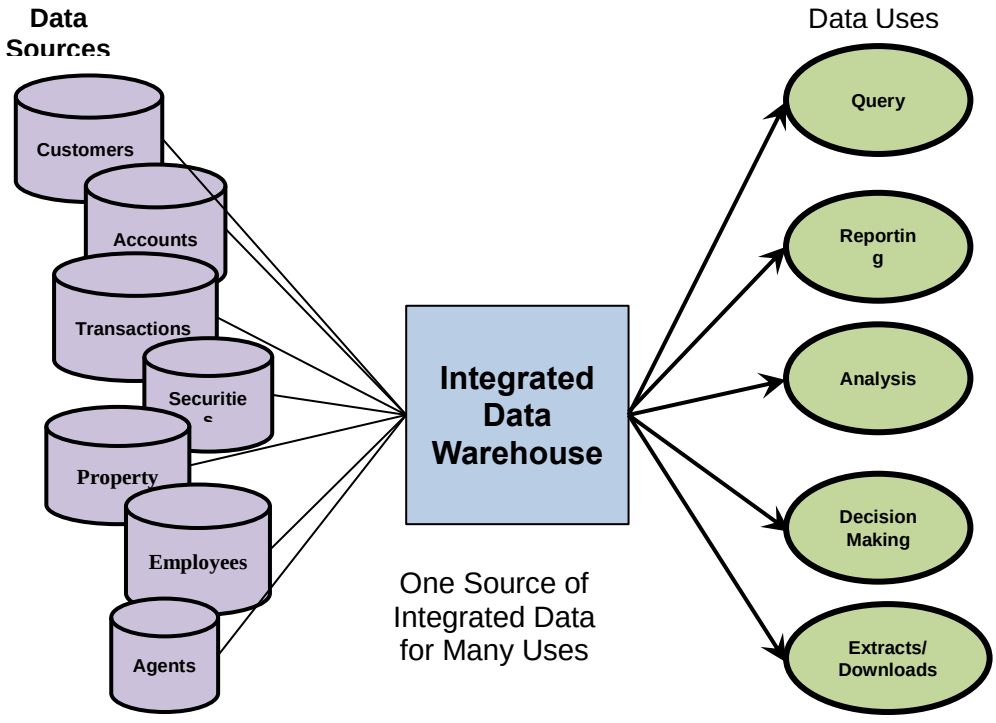


Figure 3. A data warehouse integrates many data sources for many uses and facilitates use of the integrated data

Because data warehouse platforms embrace the need for ongoing management and integration of data in addition to data analysis, they are typically built on a relational

database management system (RDBMS). As a result, data warehouse platforms have a schema layer to describe data structure, integrity constraints and relationships; support for SQL; a sophisticated query optimizer for high-performance query execution; indexing options to speed access to data; workload management facilities and a highly parallel architecture.

To fully enable its data management objectives, the data warehouse requires a significant upfront development effort to create the infrastructure for repeatable use of the data. Typically, before using the data in production, one must:

- Create a business model of the data.
- Create the logical data definition, called the schema.
- Create the physical database design.
- Create extract/transform/load (ETL) processes to cleanse, validate, integrate and load the data into the data warehouse in a format that conforms to the model.
- Create business views of the data for data delivery.

It is impractical to realize the full value of the data warehouse in support of repeatable processes without first completing these steps. Once they are completed, users can begin to access the data warehouse for business purposes, e.g., create and execute queries and analyses in SQL using the underlying data definitions.

Critical advantages of data warehouse technology are its infrastructure and the fact that SQL is a declarative language. The user simply specifies in the SQL query the data to retrieve. The user does not have to specify where to find the data, how to integrate it or how to manipulate it. This is all handled automatically by the data warehouse platform, dramatically reducing the cost of creating any new query, analysis or analytic application that accesses the same data.

For most large organizations, business requirements, data sources and data structures change frequently. Such changes are often implemented by modifying the data definitions, without changing most or all of the applications and queries that use the data.

In summary, the major advantages of a data warehouse platform for analytics are:

- The data management infrastructure that describes the data to the business, maintains the integrity of the data and controls and optimizes access to data.
- The ability to use SQL for both simple and complex queries and analyses.
- The ability to easily change the data definitions to accommodate changes to business requirements and data sources, without changing the applications or user views that access the data.

While data warehouse technology has many advantages, there is effort and cost required upfront to set up the data infrastructure for support of repeatable processes. In addition, large-scale data warehouses typically involve a substantial cost to license the necessary database management software.

1.3 The Hadoop Platform for Big Data

Hadoop (officially the Apache Hadoop project) is an open-source software library that supports distributed processing of large datasets across clusters of computer servers (see Figure 4). Hadoop clusters, running the same software, can range in size from a single server to as many as tens of thousands of servers.

Hadoop employs a distributed file system called HDFS, which automatically distributes files across the cluster and automatically retrieves data by file name. HDFS has built-in support for data replication. For example, no data is lost in the event of a disk drive failure.

MapReduce, a programming framework for analytic data processing, enables Hadoop to process large datasets in parallel across all nodes in the clusters. The primary advantage often cited for Hadoop is its low cost for applications involving large data volumes, data variety and high data velocity.

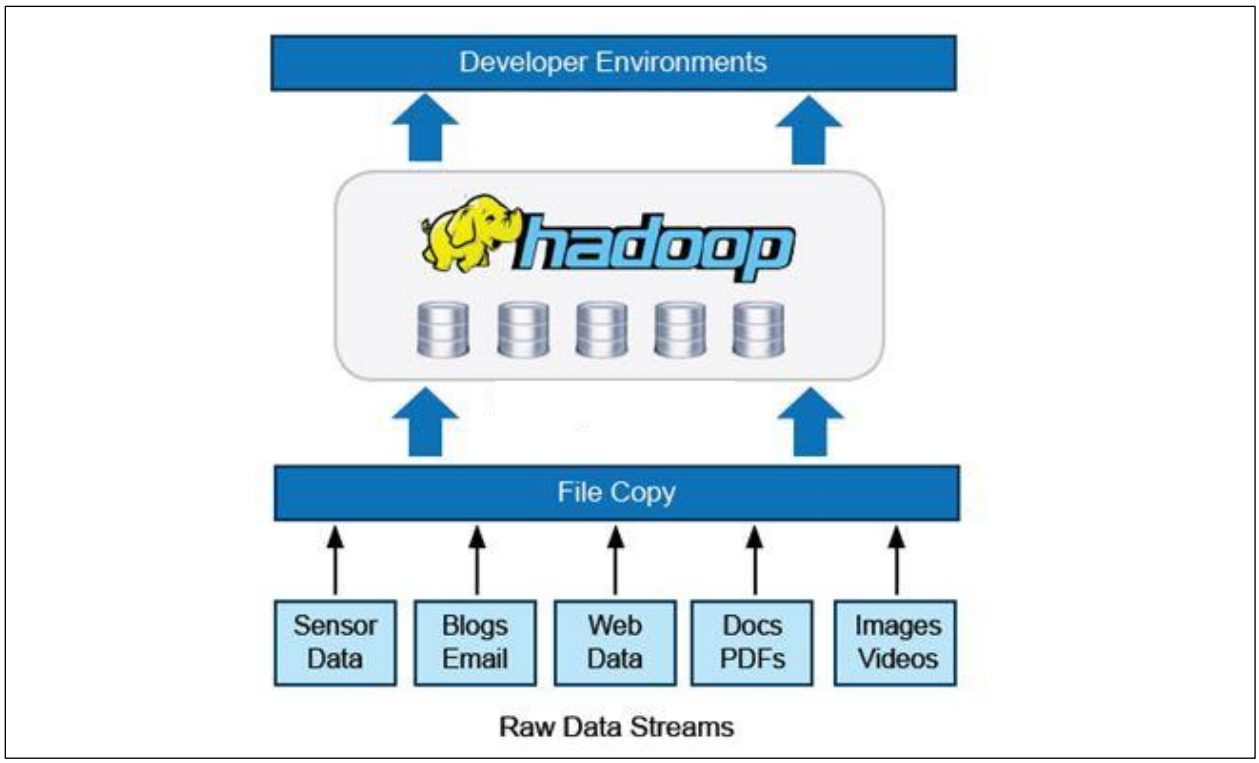


Figure 4. Hadoop supports distributed processing of large data volumes and many different data types across clusters of servers. (Source: Apache Foundation)

Setting up an analytic environment on Hadoop does not require the same initial investment as data warehousing. There are no software license fees for Hadoop itself. Once the organization acquires the hardware for the Hadoop cluster(s), it can install HDFS and MapReduce, move the data onto the cluster and begin processing the data. Specialized skills may be required to configure and implement the cluster for demanding applications, but when those skills are available, the process can be completed quickly.

On the other hand, using Hadoop for ongoing analytics can require significant development effort. Hadoop is not a data management platform and does not include many of the RDBMS components described above, such as a data schema layer, indexing or a query optimizer. HDFS does not change files once they are written. Thus, if any data in a file changes, the entire file must be rewritten. Application programs for Hadoop are typically written in Java or other procedural languages, and these programs must include all specifications for data structure, location, integration and processing.

There is a language, HiveQL, for writing simple queries without programming. The component of Hadoop that processes these queries, Hive, also supports schemas (a schema is a definition of the data stored in the system). However, Hive and HDFS do not enforce the schema or any other integrity constraints on the data, nor can HiveQL be used for complex queries. In addition, Hive performs limited query optimization when compared to an advanced data warehouse platform.

In summary, Hadoop's key advantages include the following:

- Low system acquisition cost.
- Fast, parallel processing of large volumes of data and many different data types.
- Access to data without extensive upfront data modeling and database design.
- Automatic, system-maintained data replication.
- Ability to support a very large number of servers in a single cluster.

1.4 The Total Cost Concept

A common estimate for the list price to acquire a Hadoop cluster, including hardware and open source software, is less than $1,000 US per TB of data stored. This is several times less than the average list price of widely-used, highly-parallel data warehouse platforms. Thus, many people assume that an analytic data solution implemented on Hadoop will always cost less than the same solution on a data warehouse platform. Some might imagine that system acquisition price is the only important factor.

However, getting the business value from data is not only about "system" price. To understand the actual cost of any business solution, one needs to determine the cost of constructing and maintaining the solution as well as the ongoing cost of using the data for its business purpose.

WinterCorp has developed a cost framework for big data solutions to help customers understand the larger cost picture around using data for a business purpose, and how those costs can be vastly different in data warehouse and Hadoop environments for the same application.

The next section of this report describes the framework and discusses how organizations can use it to develop a better understanding of what a particular analytic solution will actually cost over time.

2 Total Cost of Data (TCOD) Framework

2.1 Overview

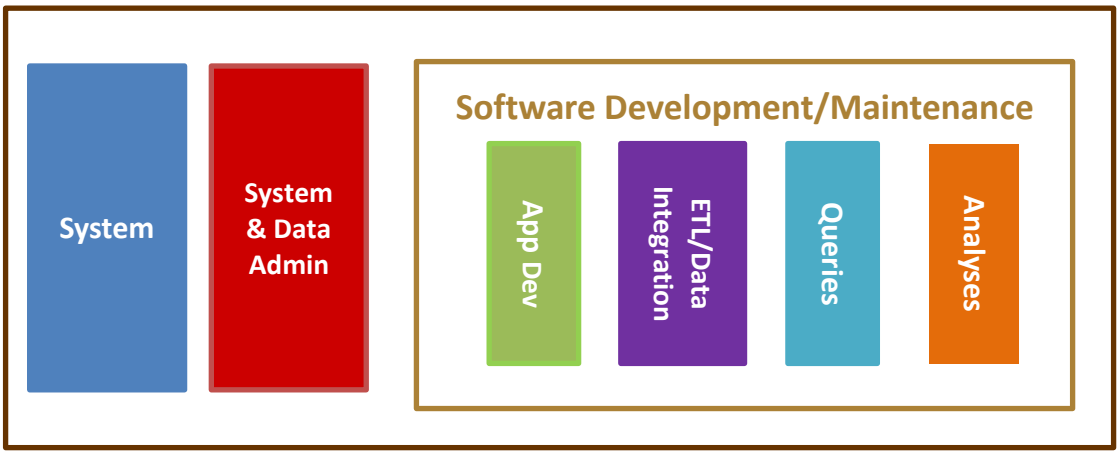


Figure 5. Components of Total Cost of Data (TCOD)

To capture the total cost of a big data solution, we propose a framework for estimating the ***total cost of data,*** or TCOD. In addition to ***system cost***, TCOD takes into consideration the ***cost of using the data*** over a period of time. This includes the cost of developing and maintaining the business solutions – complex queries, analyses and other analytic applications – built on top of the system. These solutions deliver the real value of data to the organization. Thus, it is critical for an organization to identify and understand the total cost picture before selecting and implementing any technology for a big data analytic solution. The major cost components of TCOD (see Figure 5) are:

- *System cost* – The cost to acquire, maintain/support and upgrade the hardware and system software plus the cost of space, power and cooling. In the case of a data warehouse, this also includes the cost of the database management software.
- *Cost of system and data administration* – The cost of expert staff to administer the system and the data it stores.

- *Cost of data integration* – The cost of developing or acquiring an ETL (extract, transform and load) or ELT solution to prepare data for analytic use. This is the cost of developing a process to cleanse the source data, reorganize it as necessary and store it in the database in accordance with an integrated database design.
- *Cost to develop queries* – The cost of developing queries that can be expressed in SQL.
- *Cost to develop analyses* – The cost of developing procedural programs that perform data analyses too complex to express in SQL.
- *Cost to develop analytic applications* – The cost of developing substantial application programs that use the data to support repeatable processes such as campaign management or loan approval in a financial institution.

An analogy from everyday life is deciding whether to drive or fly from Boston to San Diego for an extended vacation in the winter. If the cost of a round-trip airline ticket is high, driving might look attractive as a potentially less expensive alternative. But driving isn't only about the cost of car maintenance and fuel (i.e., the "system cost"). Driving cross-country involves significantly more time (i.e., "labor cost") than flying, plus the cost of overnight stays and meals. On the other hand, driving avoids the cost of renting a car in San Diego. The total cost of the two alternatives would depend on several variables, including:

- The number of people traveling ("users")
- The total estimated time required for the trip (number of days driving compared to time flying, which depends in turn on the number of miles driven each day vs. the flight schedule). Getting there in less than one day might be worth the airline ticket cost depending on the cost/value of each traveler's time. Alternatively, a week of driving each way might be worth it if one is spending a relatively long time in California, and/or there is value in the experience of driving cross-country.
- The cost of airline ticket(s) and baggage vs. the cost of fuel, car maintenance, hotels and meals.

- The cost of renting a car vs. using one's own car while in California.

This example clearly shows that assessing the total cost of travel options is more complex than it appears on the surface, and involves more than just the obvious out-of-pocket costs.

The TCOD framework enables an organization to compare the anticipated long-term costs of implementing different technologies to meet specific big data analytic requirements. The example relevant to this report is evaluating a Hadoop cluster versus a data warehouse platform in an effort to understand where each technology solution works best. The key goal is to help organizations develop and deliver cost-effective big data analytic solutions while minimizing the risk of making an expensive mistake in the process. Ignoring a component of total cost does not, unfortunately, make the cost go away. The organization will pay that cost sooner or later.

2.2 Using the TCOD Framework

The TCOD framework consists of a set of cost variables and values that can be customized for specific big data solution requirements. These values are then aggregated over a specific time period to produce a TCOD profile for a particular solution or technology.

As described in Section 3 below, WinterCorp has used this framework to develop and analyze the relative five-year TCODs of using Hadoop and a data warehouse platform to implement each of the two big data application examples presented. The accompanying spreadsheet (see www.wintercorp.com/bigdata-spreadsheet) documents the cost assumptions and detailed cost calculations used to develop these TCOD profiles. Here is a summary of the assumptions that apply to both of the big data application examples:

- The *cost horizon* is five (5) years.
- The *system acquisition cost* for each platform is based on an average list price per TB for user data storage across widely-used products plus a 30% illustrative

vendor discount for data warehouse platforms and a 10% illustrative vendor discount for a Hadoop cluster.

- The *data compression ratio* is assumed to be two (2) on all platforms. This is the raw data volume divided by the compressed data volume and determines the actual user data storage required. The data compression ratio affects the system acquisition cost; system maintenance and support cost; and space, power and cooling cost. An organization's actual data compression ratio will vary according to its data profile and big data platform choice.
- *System upgrade costs* are based on a 26% compound annual growth rate in system capacity. This results in a doubling of system capacity every three years. We consider this reasonable given the anticipated fast pace of growth in big data environments. Vendor discounts apply to upgrade costs.
- *System maintenance and support costs* are assumed to be 10% of acquisition cost for Hadoop and 20% for data warehouse platforms, annually, including upgrades. In the case of DW platforms, we assume that first year maintenance is included in the acquisition cost. Vendor discounts also apply to system maintenance and support costs.
- *Space, power and cooling costs*, like system acquisition cost, are based on an average annual cost per TB across widely-used products. We use vendor estimates for these figures.
- *Total system cost* is the sum of system acquisition cost; system upgrade costs; system maintenance and support costs and space, power and cooling costs.
- *ETL/ELT cost* is estimated from the size of the full-time equivalent (FTE) staff required to develop and maintain ETL routines. In our examples, we assume no ETL cost for Hadoop-based solutions.
- *Salary data* for various programmer/developer, system and data administrator and ETL developer positions are national averages from www.indeed.com. Total cost

of employment for one full-time equivalent in each position is salary plus 50% to cover overhead costs.

- All *cost-per-line-of-code* assumptions combine the *salary data* with data on *developer productivity* (lines of code per staff month) for business and engineering applications of varying sizes from Quantitative Software Management, Inc. (QSM), McLean VA (www.qsm.com). QSM developed its data from 10,000 projects it has tracked since 2000. (As a note, QSM also develops and publishes productivity data based on function points, available at the same web site.)
- *Business days in a year* is 250 (50 weeks * 5 business days/week)

3 Big Data Examples

Let's look at two very different examples of big data environments; how the cost framework works for each one when implemented on a Hadoop cluster versus a data warehouse platform; and what conclusions can be drawn about the best uses of each solution.

3.1 Engineering Analytic Example: Data Refining

Many types of industrial equipment are currently manufactured with built-in sensors, as described in General Electric (GE)'s vision report on the rise of the Industrial Internet. For example, diesel engines are instrumented to generate data many times per second when the engine is operating. According to GE, a single jet engine produces data at approximately 10TB/hour. GE estimates there are over 2.3 million diesel engines in use worldwide in rail, air and marine systems. The manufacturers, users and maintainers of these engines all have an interest in the insights that can result from analyzing this operational data. The data can then be used to optimize maintenance, troubleshoot problems, maximize safety, engineer better products in the future, plan logistics and accomplish many other purposes. However, the volume of data can be so huge that analyzing it poses a challenge. Even with modern systems expressly designed for big data, it is economically feasible to store all of the raw data for only a limited period of time. Furthermore, most of the data is not interesting. If an engine sensor reports many times a second that everything is exactly as intended, how valuable is it to retain all the detailed data? Clearly, the goal is to retain for longer-term analysis only the readings that are in some way out of the ordinary. These are the readings that can be used to predict failures before they occur, stage replacement parts where they will be needed and engineer better products for the future.

As a result, many industrial, sensor-based big data applications present a *data refining challenge.* The need is to process a huge volume of raw data through a system that can filter out only the data worth retaining for deeper analysis and/or longer-term

storage. Determining what to save, however, may depend on a complex combination of conditions. For example, each of several readings – temperature, pressure, vibration and humidity – may be acceptable but the *combination* of readings may not be. Or the readings from different parts of an engine, taken together, may suggest a problem is developing. The data-refining process itself can be analytically complex and, because of the steady inflow of raw data at a high rate, it can also present stringent throughput requirements.

In this example, we assume a small group of engineering experts designs and writes custom Java programs to analyze the sensor data from multiple instrumented pieces of equipment. Each analysis can deal with terabytes or petabytes of raw data. Only data related to specific events or trends of interest is retained long-term; the rest is discarded.

Cost with Hadoop. Let's apply the cost framework to this engineering analytic example as it would be implemented on Hadoop. In addition to the general assumptions described earlier, we also assume this example would require an estimated data volume of 500 TB, 300,000 lines of new Java/MapReduce code per year (a proxy for acquired applications plus developed applications) and staffing of one full-time equivalent (FTE) to maintain and support a 500 TB Hadoop cluster (see Figure 6).

Using these assumptions, the five-year total system cost is $1.4 million, system and data administration cost is $0.75 million and application development costs are $7.2 million for a TCOD of $9.3 million. In this case, Hadoop is a good fit and a very cost-effective solution.

	Cost Factor	Value	Comment
A	Cost modeling horizon	5	Years
B	Lines of Java/MapReduce code for data refining example	300,000	Per year; proxy for acquired + developed apps
C	Loaded cost per year for a Hadoop developer	150,000	Salary + overhead (from Salary Data worksheet)
D	Source lines of code (SLOC)/person-month, large (100k lines) data refining example	529	From Effort Per SLOC worksheet
E	Cost of a line of Java/MapReduce code, large (100K lines) data refining example	$24	Uses effort per line of code for a large (100k lines) data refining example
F	Hadoop cluster acquisition cost, data refining example	$1,000	Per TB, list; open source software is free
G	Vendor discount for Hadoop cluster, data refining example	10%	Illustrative discount
H	Hadoop cluster maintenance & support cost, data refining example	$100	Per TB/year, list; calculated as 10% of F; applies to years 1 through 5 (assume first year maintenance NOT included in F)
I	Hadoop cluster space, power and cooling cost, data refining example	$301	Per TB/year
J	Data volume for data refining example	5,000	TB
K	Rate of annual growth in system capacity, data refining example	26%	Capacity grows to keep up with growth in data volume and analytical workload; applies to years 2 through 5
L	Data compression ratio, Hadoop cluster, data refining example	2	Raw data volume divided by compressed data volume; determines actual storage capacity required
M	Loaded cost per year for a Hadoop administrator	$150,000	Salary + overhead (from Salary Data worksheet)
N	Staffing required to maintain and support a PB-scale Hadoop cluster, data refining example	1	Full-time equivalents (FTEs)/year

Figure 6. Cost assumptions for the data refining example on Hadoop

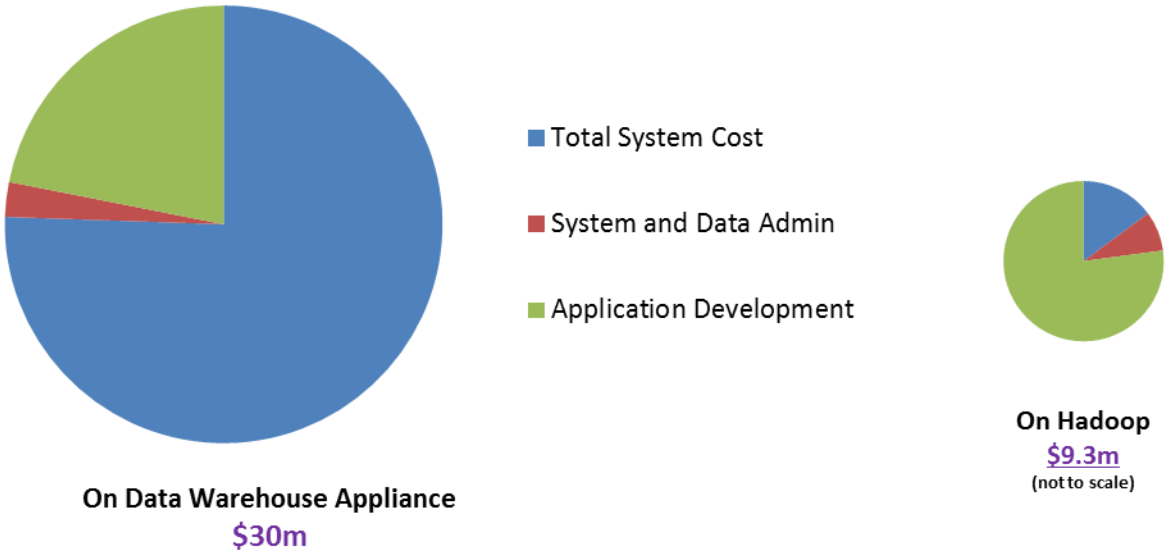
Cost with data warehouse technology. Implementing the engineering example using data warehouse technology requires additional cost assumptions, shown in Figure 7. We assume the system is a data warehouse appliance. Many data warehouse vendors now provide data warehouse appliances designed to handle a less complex big data workload at a lower cost than an enterprise-level data warehouse product.

	Cost Factor	Value	Comments
O	Data warehouse appliance acquisition cost	$37,000	Per TB, list; based on an average of widely-used data warehouse appliance products
P	Vendor discount for data warehouse appliance	40%	Illustrative discount
Q	Data warehouse appliance maintenance & support cost	$7,400	Per TB/year, list; calculated as 20% of O; applies to years 2 through 5 (first year maintenance & support included in O)
R	Data warehouse appliance space, power & cooling cost	$291	Per TB/year
S	Data compression ratio, data warehouse appliance	2	Raw data volume divided by compressed data volume; determines actual storage capacity required
T	Loaded cost per year for a data warehouse DBA	$154,500	Salary + overhead (from Salary Data worksheet)
U	Loaded cost per year for a SQL developer	$139,500	Salary + overhead (from Salary Data worksheet)
V	Cost of a line of Java/SQL code, SQL developer, large (100k lines) engineering example	$22	Uses effort per line of code for a large (100k lines) engineering example
W	Staffing required to maintain and support the data warehouse for engineering example	1	Full-time equivalents (FTE)/year

Figure 7. Additional cost assumptions for the data refining example on a data warehouse appliance

With data warehouse technology, the solution would be far more expensive, about $30 million in TCOD compared to $9.3 million with Hadoop. Figure 8 shows a comparison of these TCOD profiles.

The data warehouse appliance solution is more expensive than Hadoop because the total system cost of $23 million is so much higher. System cost now dwarfs the $7 million application development cost. In addition, the engineering application would not derive significant incremental benefits from the capabilities of the data warehouse appliance, since the requirements for large data volume and fast processing are satisfied by Hadoop.



	Data Warehouse Appliance	Hadoop
Volume of Data	500 TB	500 TB
System Cost	$22.7	$1.4
Initial Acquisition Cost	$5.5	$0.2
Upgrades at 26% CAGR	$8.4	$0.3
Maintenance/Support	$8.2	$0.2
Power/Space/Cooling	$0.6	$0.7
Admin	$0.8	$0.8
Application Development	$6.6	$7.2
Total Cost of Data	$30 million	$9.3 million

Figure 8. Comparison of TCODs for the engineering example

Clearly, Hadoop provides a big opportunity here, and there are analogous applications in the business environment, such as analyzing web log data for website optimization.

Many commercial problems in finance, petroleum exploration and other industries have analytic requirements similar to those of our data refining example:

- High volume of data (hundreds of TBs to PBs).
- Intensive analysis of a small number of closely-related datasets.
- Analysis reads the entire dataset.
- Raw data life is typically short, a few hours to a few weeks; only small subsets are retained.
- A small group of experts collaborates on the analysis.

In these scenarios, the cost equation favors Hadoop. There are many other excellent Hadoop use cases as well. Two other proven examples are using Hadoop as: (1) the initial landing zone for all data coming into the enterprise and (2) an online, searchable and queryable data archive.

3.2 Business Example: Financial Enterprise Data Warehouse

At the opposite end of the analytic data spectrum is implementing an enterprise data warehouse (EDW) in a large company. In this case, an EDW in a financial institution, such as a bank or insurance company, uses data warehouse technology to help thousands, or even millions, of users query, analyze and examine similarly large volumes of business data daily.

The EDW presents different challenges with respect to managing data. First, there is the need to integrate many data sources, typically hundreds or thousands in a large organization. For example, there could be a hundred or more different sources of information just about customers. This customer data must be consolidated into a single view for analysis and reporting. Getting the integration right requires a comprehensive data model and database design.

In a financial institution, every detailed transaction must be accurate and have integrity. For example, accounts have to balance; transactions must be reconciled to the right account; and and accounts must be matched to the right individuals, households or

business entities. This requires complex, sophisticated data validation routines and system-enforced constraints to maintain integrity across applications and over time.

Typically, the data and analytic applications live for years or decades and must be managed throughout their entire lifecycle.

In a healthy business, new business needs arise daily and data sources change frequently. If there are 1,000 data sources and each one changes only once a year, that results in an average of 20 changes per week. Other aspects of the EDW environment are dynamic as well, such as business rules, workloads and types of queries. The solution has to be flexible and agile enough to handle these changes efficiently. Otherwise, it cannot deliver value to the business and its costs will become prohibitive.

In an EDW there can be thousands or even millions of users sharing data both inside and outside the organization: employees, customers, prospective customers, suppliers, etc. Ongoing usage of the EDW can involve hundreds of analytic applications, thousands of one-time analyses and tens of thousands of complex queries. These all have to be developed, managed and maintained; and they, too, rely on shared data that is fed by many changing data sources. If users and applications are not buffered from changes in the data sources, the enterprise faces an untenable burden.

Even a change in the distribution of values for a single data attribute can undermine the performance of an application unless the query optimizer in the data warehouse platform automatically adapts its execution plan. Thus, the enterprise relies on the infrastructure of the data warehouse platform – the system-enforced schema, the high-level declarative language (SQL), the query optimizer and the workload manager – to maintain the usability of the integrated collection of data and the performance of its applications over any extended period of time.

Cost with data warehouse technology. Applying the TCOD framework to the EDW requires additional cost assumptions, including the upfront costs of developing the EDW infrastructure (see Figure 9). We assume the following:

- An initial EDW data volume of 1 PB.
- A staff of 25 ETL developers to develop and maintain ETL routines.
- Complex queries – A complex query is a query that (1) can be expressed in SQL and will be performed with satisfactory efficiency on the data warehouse platform and (2) is not a simple query. (A simple query is one that can be expressed in HiveQL and processed with satisfactory efficiency on Hadoop.) We assume a complex query is 100 lines of SQL code on average with 10 new distinct complex queries per business day by EDW users.
- Analyses – These are analyses too complex to be expressed in SQL, or those that will not be performed with reasonable efficiency in SQL. Therefore, they are expressed in a procedural language such as Java. We assume an analysis consists of a complex query (100 lines of SQL) to select the data plus a Java/SQL (or similar language) program of 900 lines to process the data. There is 1 new distinct analyses per business day.
- There are 300,000 lines per year of new analytic application code (a proxy for acquired applications plus developed applications).
- System and database administration requires 10 FTEs per year. This staffing figure includes database design, data modeling, schema development and maintenance, performance management and system administration.

	Cost Factor	Value	Comments
X	Source lines of code (SLOC)/person-month, large (100k lines) business app	1,052	From Effort Per SLOC worksheet
Y	Source lines of code (SLOC)/person-month, small (1k lines) business app	164	From Effort Per SLOC worksheet
Z	Cost of developing a line of complex SQL for a complex query or analysis, EDW environment on DW platform	$71	Uses effort per line of code for a small (1k lines) business app
AA	Lines of SQL in a complex query	100	Cost of development per complex query is (Z*AA) = $7,100
BB	Lines of procedural code in a complex analysis, EDW environment on DW platform	900	After data has been selected via a complex SQL query
CC	Cost of developing a line of procedural code, small (1k lines) business app, EDW environment on DW platform	$71	For use in a business analysis
DD	Cost of developing a line of Java/MR procedural code, small (1k lines) business app	$76	For use in a business analysis or complex query
EE	Cost of developing a complex business analysis, EDW environment on DW platform	$71,000	(Z*AA) + (BB*CC)
FF	EDW system acquisition cost	$72,000	Per TB, list; based on an average of widely-used EDW platform products
GG	Vendor discount on EDW system	30%	Illustrative discount
HH	EDW system maintenance & support cost	$14,400	Per TB/year, list; calculated as 20% of FF; applies to years 2 through 5 (first year maintenance & support included in FF)
II	EDW system space, power and cooling cost	$692	Per TB/year
JJ	Data volume for EDW example	1,000	TB (in a large financial institution); one PB system initial user data storage capacity with enterprise workload
KK	Rate of annual growth in system capacity, EDW example [2]	26%	Capacity grows to keep up with growth in data volume and analytical workload; applies to years 2 through 5
LL	Data compression ratio, EDW system	2	Raw data volume divided by compressed data volume; determines actual storage capacity required
MM	Database and system administration in the EDW environment on DW platform	10	FTEs/year, DW professional (in large financial institution); this includes data modeling, database design and performance management
NN	Loaded cost per year for an ETL Developer, EDW environment on DW platform	$147,000	Salary + overhead (from Salary Data worksheet)
OO	ETL in the EDW environment on DW platform	25	FTEs/year, ETL Developer (in large financial institution)
PP	Cost of a line of application code using SQL for data access, large (100k lines) business app, in the EDW environment on DW platform	$11	Uses effort per line of code for a large (100k lines) business app
QQ	Lines of new application code, SQL based, EDW example	300,000	Per year (in large financial institution); proxy for acquired + developed apps
RR	Business days in a year	250	50 weeks * 5 business days per week (assumes 10 holidays)
SS	Distinct new complex queries, EDW example	10	Per business day
TT	Distinct new data analyses, EDW example	1	Per business day

Figure 9. Additional cost assumptions for the EDW using data warehouse technology

Using data warehouse technology for the EDW results in a five-year total system cost of $44.6 million. Of this, the initial acquisition cost is estimated at $10.8 million.

The organization will spend about $265 million in TCOD over five years to achieve its business goals with data (see Figure 10). In this example, system cost comprises about 17% of the TCOD and initial acquisition represents about 4% of the TCOD. The larger costs are those incurred in developing ETL processes, complex queries and analyses. Thus, what drives the business equation here is the cost of ***using*** the data and developing the artifacts that produce the business payoff. These costs add up to much more than the system cost.

	Data Warehouse Platform
Volume of Data	500 TB
System Cost	$44.6
Initial Acquisition Cost	$10.8
Upgrades at 26% CAGR	$16.4
Maintenance/Support	$15.9
Power/Space/Cooling	$1.5
Admin	$7.7
Application Development	$16.5
ETL	$18.4
Complex Queries	$88.7
Analysis	$88.7
Total Cost of Data (TCOD)	**$265 million**

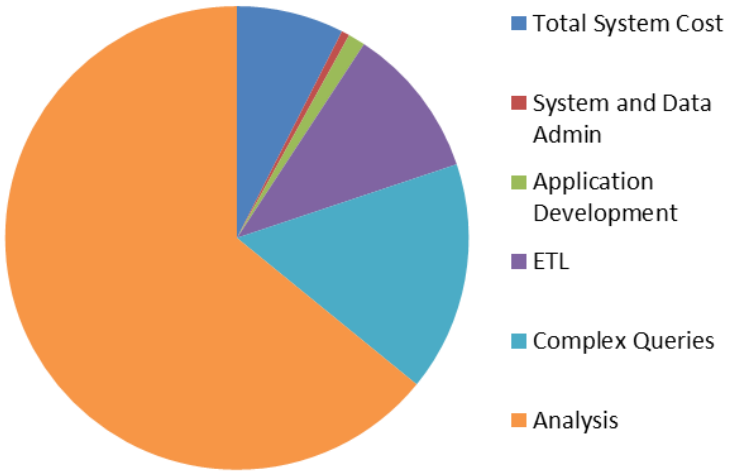


Figure 10. TCOD for the EDW using data warehouse technology

Cost with Hadoop. Now let's look at what is involved if the organization builds the EDW on Hadoop, with the following assumptions:

- Data is described in Hive schemas.
- Simple queries are written in HiveQL.
- Complex queries, analyses and analytic applications are written in Java programs using MapReduce.
- No ETL development is done in the Hadoop EDW environment; data is processed and analyzed in the form in which it is initially received.

An important point here is that Hadoop, as a distributed file system, lacks the functionality of a DBMS and the infrastructure components of the data warehouse platform. Many functions and capabilities – for example, how data is accessed, integrated and processed – now have to be included in the application code itself (i.e., the Java programs written for each query, analysis or analytic application). (Note: While many BI tools interface to Hadoop, the extensive optimization provided by the data warehouse does not exist in Hadoop, as discussed below. Therefore, only simple queries are practical via these tools when used with Hadoop.)

In an EDW there are many large tables that represent different entities and relationships among them. Financial organizations have customers, accounts, transactions, households, etc. Healthcare companies have claims, patients, providers, diseases, prescriptions and diagnoses. It is not possible to leverage data across a complex business by selecting just one way to join these together, a single path through the data for queries and analysis. Without a query optimizer in Hadoop, the programmer has to choose the most efficient way to access and process the data.

Thus, development of complex queries, analyses and analytic applications takes more effort and requires more skill in Hadoop. This is reflected in our additional development cost assumptions for this scenario (see Figure 11):

- Complex queries written in Java/MapReduce require five (5) times as many lines of code as in SQL (i.e., 500 lines vs. 100 lines).
- Procedural code in the Hadoop EDW environment requires twice as many lines of code as in the EDW environment. Therefore, an analysis involves a total of 2,300 lines of code. This is based on 500 (5 x 100) lines of code for the complex query to select the data plus 1,800 (2 x 900) lines of code for the algorithm to process the data. Analytic applications require twice as much code (600k vs. 300k lines).

Finally, staffing for data administration and for system administration/maintenance/support in the EDW environment on Hadoop each require one FTE per year.

Cost Factor			Comment
UU	Cost of developing a line of Java/MR code, large (100k lines) business app, EDW example	$12	Uses effort per line of code for a large (100k lines) business app
VV	Lines of code ratio for Java/MR vs. SQL in a large (100k lines) business app, EDW example	2	A large business app written in Java/MR in an EDW environment will require more lines of code than the same app written in SQL
WW	Lines of code ratio for Java/MR vs. SQL in a complex query, EDW example	5	A complex query written in Java/MR in an EDW environment will require more lines of code than the same app written in SQL
XX	Lines of code in a complex business analysis, EDW on Hadoop	2,300	Calculated as (WW*AA) + (VV*BB) for an analysis written in Java/MR.
YY	Hadoop cluster acquisition cost, EDW example	$1,000	Per TB, list; open source software is free
ZZ	Vendor discount for Hadoop cluster, EDW example	10%	Illustrative discount
AAA	Hadoop cluster maintenance & support cost, EDW example	$100	Per TB/year, list; calculated as 10% of YY; applies to years 1 through 5 (assume first year maintenance NOT included in YY)
BBB	Hadoop cluster space, power and cooling cost, EDW example	$301	Per TB/year
CCC	Data compression ratio, Hadoop cluster, EDW example	2	Raw data volume divided by compressed data volume; determines actual storage capacity required
DDD	ETL in the EDW environment, Hadoop cluster	0	FTEs/year, Hadoop Developer (in large financial institution)
EEE	Data administration, EDW on Hadoop	1	FTEs/year, Hadoop Administrator; includes admin of Hive schemas
FFF	Staffing required to maintain and support a PB-scale Hadoop cluster, EDW example	1	FTEs/year, Hadoop Administrator

Figure 11. Additional cost assumptions for the EDW on Hadoop

Using these assumptions, the TCOD shows that it will cost hundreds of millions of dollars more to create and maintain a complex data warehouse using Hadoop compared to data warehouse technology (see Figure 12). With Hadoop, the system cost is a small percentage of the total cost and disappears on the pie chart, while the costs of developing complex queries and analyses increase greatly. The cost of developing and implementing every new business requirement is now very high, and the savings in the cost of the Hadoop system do not compensate for this additional cost. At a total cost of $739 million, Hadoop cannot match the cost-effectiveness and functionality of the data warehouse technology for the EDW.

	Hadoop
Volume of Data	500 TB
System Cost	$1.4
Initial Acquisition Cost	$0.2
Upgrades at 26% CAGR	$0.3
Maintenance/Support	$0.2
Power/Space/Cooling	$0.6
Admin	$8.5
Application Development	$36
ETL	$0
Complex Queries	$475
Analysis	$219
Total Cost of Data (TCOD)	**$740 million**

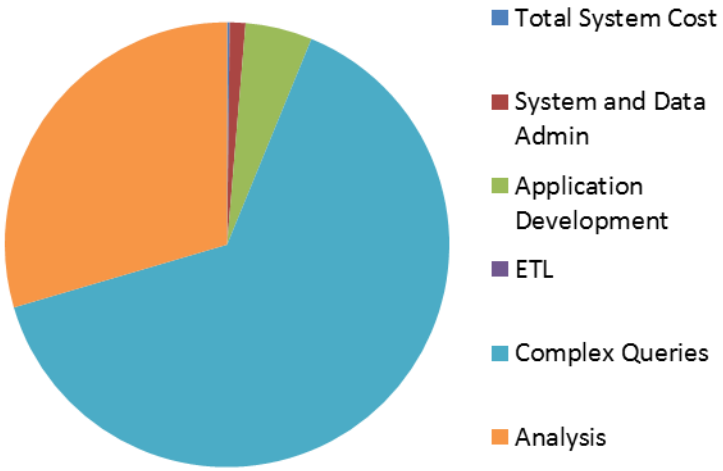


Figure 12. TCOD for the EDW using Hadoop

We further investigated the conclusion that Hadoop is the more expensive EDW solution by testing the sensitivity of the TCOD framework to changes in key cost assumptions: frequency of data usage, the cost of EDW development in the Hadoop environment and data volume. These tests and our results are described in Section 4.

3.3 Comparison of Big Data Solutions

These two big data implementations have very different data management requirements and TCOD profiles depending on the solution platform (see Figure 13). Comparing and contrasting them is instructive in understanding the relative strengths of the underlying platforms and the types of applications for which each is best suited. The TCOD framework shows that in the engineering example, the Hadoop solution saves over $20 million in total cost over five years compared to the data warehouse appliance solution. In this example, Hadoop reduces the system cost by over 90% when compared to a data warehouse appliance.

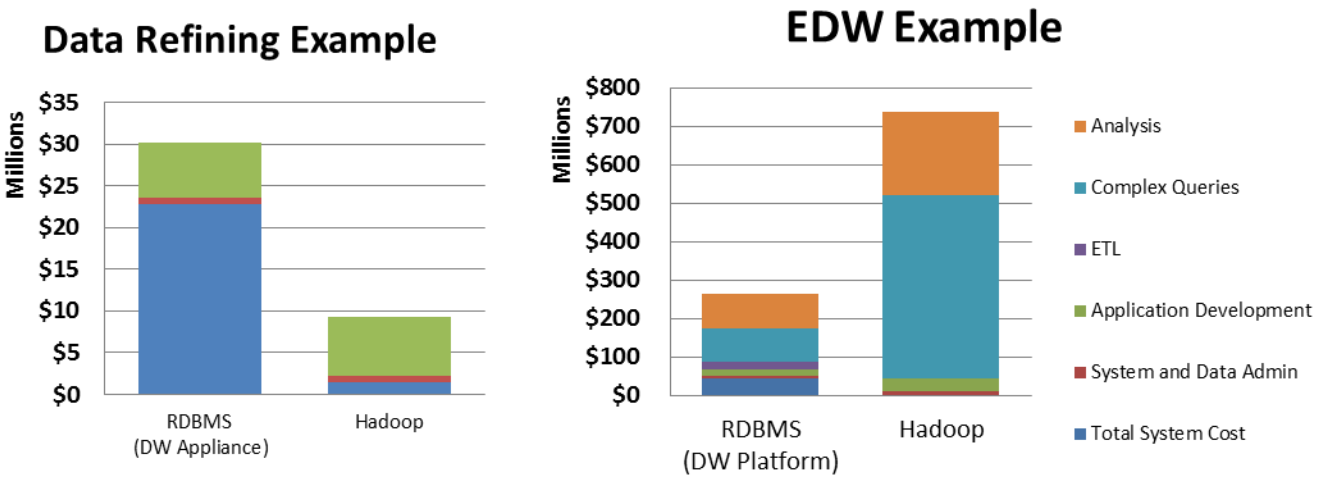


Figure 13. Summary comparison of five-year TCODs for both examples

In the EDW implementation, if one looks at system cost alone, data warehouse technology appears to be much more expensive than Hadoop. However, other cost components of the EDW have a larger impact on the total cost calculation. In fact, a Hadoop solution in the EDW example costs a total of $474 million dollars more than using data warehouse technology ($739 million vs. $265 million), or 2.8 times as much.

In general, the TCOD shows that data warehouse technology produces a more favorable total cost profile in situations where there substantial numbers of:

- Distinct complex queries
- Distinct analyses
- Custom analytic applications to be developed

These characteristics result in a large return on the investment in data warehouse technology.

Hadoop technology results in cost advantages in situations with:

- High data volume, which leverages the low storage cost of Hadoop platforms.
- Minimal or no requirement for upfront data modeling or database design.
- High intensity, algorithmically complex computation that does not benefit from the application of SQL.
- Rapid, high volume processing of data that takes advantage of Hadoop's highly parallel architecture and support for large clusters of servers.

In addition, though not addressed directly in the cost model, Hadoop is advantageous where there is high data variety, including data formats and types that do not benefit directly from the relational data model used by data warehouse platforms. Examples include free-form text, images, scans, video and other so-called "unstructured" digital objects.

Figure 14 presents one characterization of the space of all analytic applications. The vertical axis represents data volume. The horizontal axis represents application and data complexity and the need for data integration.

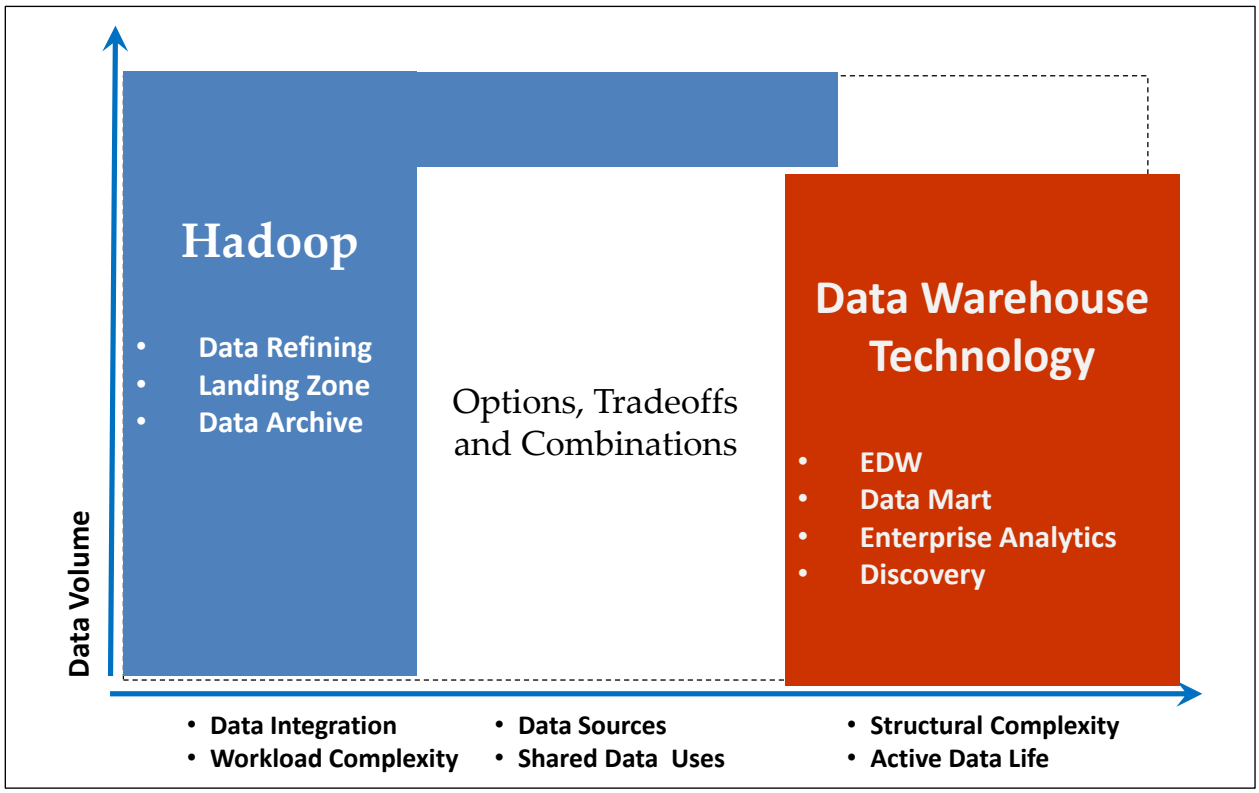


Figure 14. The big data analytic application space

The more complex the big data environment, the higher the payoff from data warehouse technology. In terms of complexity and integration, the engineering example described above falls on the far left and the EDW example on the far right on this chart. Between these extremes, organizations have options for a solution depending on specific requirements.

One example that could fall into this "middle space" is a data mining or scoring application. One characteristic of this type of application is that it often reads all the data. If an organization is scoring customers, it often scores all of them. Because Hadoop provides the ability to read entire files rapidly and process them efficiently, it could be a good fit here.

A larger question is what else the organization does with the data used by the scoring application. If the data represents customers and their purchases, it may need to be in a data warehouse for other uses. In this case, is the incremental cost of scoring the data in

place (in the data warehouse) more or less than the incremental cost of moving all of the data to a separate system (Hadoop) and scoring it there? Moving the data to Hadoop could reduce the cost of executing the scoring application, but may raise questions about how much time and effort it takes to move the data, the cost of maintaining two copies of the data, etc. Either big data solution could be the right one depending on the answers to these questions.

4 TCOD Sensitivity Analysis

The TCOD calculations show that implementing the EDW is much more cost-effective with data warehouse technology than with Hadoop. To further examine this conclusion, we tested the cost assumptions on which it is based by varying the values that represent three of the major, and largest, cost components of the EDW TCOD: the respective costs of developing analytic applications, complex queries and analyses. We looked at both 1) frequency of data use in the EDW and (2) the cost of developing these applications, queries and analyses in the Hadoop environment. The goal was to see how sensitive the TCOD results are to changes in the underlying assumptions, and how that affects the relative cost differential between the two big data solutions. We also tested the cost assumptions by varying the size of the data to see if smaller volumes changed the relative cost profiles.

4.1 Frequency of Data Use

The TCOD calculation for frequency of data use in the EDW example (i.e., the volume and frequency of new application, complex query and analysis development) is based on developing 300,000 lines of application code per year plus 25 distinct complex queries and 10 distinct analyses per day.

Level	A/D (lines/year)	Distinct Complex Queries (Per Day)	Distinct Analyses (Per Day)	TCOD-5 EDW $million	TCOD-5 Hadoop $million
9	600K	20	2.0	$432	$1,411
8	525k	18	1.8	$387	$1,236
7	450k	15	1.5	$342	$1,061
6	375k	13	1.3	$297	$886
5	*300k*	*10*	*1*	*$265*	*$739*
4	225k	8	0.8	$206	$535
3	150k	5	0.5	$161	$360
2	75k	3	0.3	$116	$185
1	0	0	0	$71	$10

Current Examples

Figure 15. Alternative assumptions for frequency of data use for the EDW

The six scenarios illustrated in Figure 15 vary from 200% of our base assumptions (which represent the middle of the range of values) to zero. The resulting TCOD for data warehouse technology and for Hadoop is shown for each scenario. Figure 16 charts the resulting TCODs and shows that data warehouse technology continues to be more cost effective by a large factor, even as the intensity of data use declines.

The assumptions required to make the TCOD of the Hadoop solution the same as that of data warehouse technology are clearly unrealistic for an intensively used EDW: about 1 distinct complex query per day, 2 distinct analytics per week and 5,000 lines of application development per year.

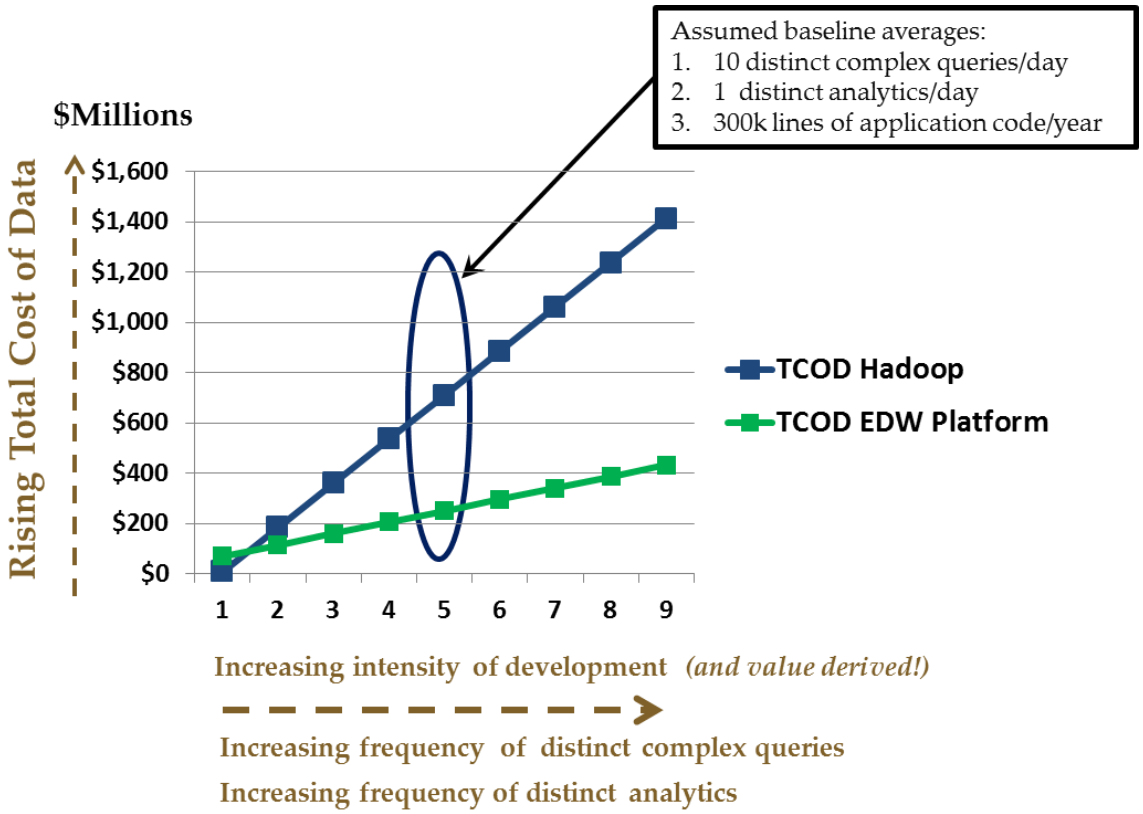


Figure 16. TCOD based on alternative assumptions for frequency of data use for the EDW

4.2 Cost of Development in Hadoop Environment

The TCOD calculation for the cost of EDW development in the Hadoop environment is based on 500 lines of Java/MapReduce in a complex query and 2,300 lines of

Java/MapReduce in an analysis. These values will vary with the project. Again, they represent the middle of the range of values for our sensitivity analysis shown in Figure 17.

Level	Query Code Ratio (Java/MR to SQL)	Lines of Code in a Complex Query (Java/MR)	Analysis Code Ratio (Java/MR to SQL)	Lines of Code in an Analysis (Java/MR)	TCOD-5 EDW (Millions)	TCOD-5 Hadoop (Millions)
10	**10**	1000	3.25	3,925	$265	$1,576
9	**9**	900	3.00	3,600	$265	$1,426
8	**8**	800	2.75	3,275	$265	$1,275
7	**7**	700	2.50	2,950	$265	$1,24
6	**6**	600	2.25	2,625	$265	$974
Current Examples						
5	***5***	***500***	***2.00***	***2,300***	***$265***	***$739***
4	**4**	400	1.75	1,975	$265	$672
3	**3**	300	1.50	1,659	$265	$521
2	**2**	200	1.25	1,325	$265	$371
1	**1**	100	1.00	1,000	$265	$220

Figure 17. Alternative assumptions for cost of Hadoop development for the EDW

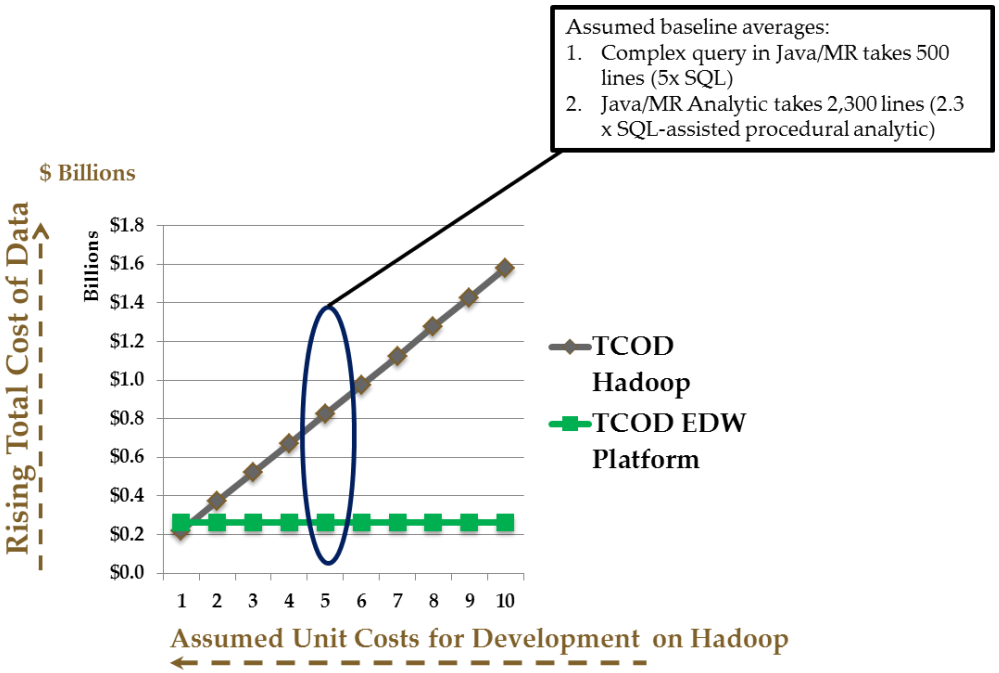


Figure 18. TCOD based on alternative assumptions for cost of development on Hadoop for the EDW

Figure 18 graphs the resulting TCODs and again shows that data warehouse technology is more cost effective across a wide range of assumptions.

4.3 Data Volume

For our enterprise data warehouse example in Section 3, we chose an initial data volume of one petabyte (PB). While today's larger data warehouses range from one to approximately 50 PB, many organizations have considerably lower data volumes.

So we also tested the cost assumptions on a range of data volumes to see what happens to the TCOD. In Figure 19, the assumed EDW data volume varies from 50 TB to 1,000 TB (1 PB).

Data Volume (TB)	TCOD-5 EDW (Millions)	TCOD-5 Hadoop (Millions)
50	$224	$738
167	$235	$747
333	$250	$747
500	***$265***	***$748***
667	$280	$748
833	$295	$749
1,000	$309	$749

Current Examples

Figure 19. Alternative assumptions for data volume for the EDW

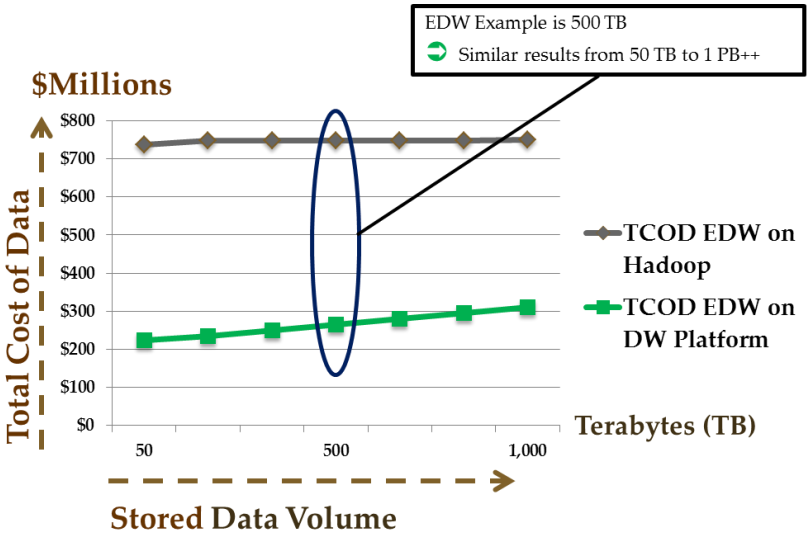


Figure 20. TCOD based on alternative assumptions for data volume for the EDW

As shown in Figure 20, changing the data volume by a factor of twenty – from 50 TB to 1 PB – has little effect on the TCOD calculation.

In this range, what matters most in the TCOD is the cost of complex queries, analyses and analytic application development, not the volume of stored data in the data warehouse. Therefore, organizations with smaller data warehouses can be confident that the TCOD framework also applies to their environments.

5 Conclusions

The TCOD analysis described in this WinterCorp Report highlights several important guidelines for selecting a big data analytic solution.

Carefully identify and evaluate the analytic and data management requirements of each application before choosing a big data platform. Our examples show that data warehouse technology and Hadoop are each effective big data platforms for specific types of analytic applications. Hadoop is far more cost effective in the data refining example. The data warehouse platform is much more cost effective in the EDW example.

In the EDW example, the system cost is lower with Hadoop, but the ongoing software costs – especially the costs of developing complex queries and analyses – are much higher. It is these costs that matter more over time. Fundamentally, by investing in the data warehouse platform and infrastructure (e.g., data model, schema, ETL), one realizes a large return on investment in developing queries, developing analyses and responding to rapidly changing business requirements.

Do not underestimate or (worse) ignore the real, long-term costs of a big data solution. It is critical to recognize that there can be six major cost components of any big data solution. In addition to the obvious system cost, the organization must also assess the ongoing costs of system and data administration, meeting data integration/ETL requirements and developing and maintaining complex queries, complex analyses and analytic applications. Over several years, each of these can be greater than the system cost and, in the case of a data warehouse requirement, the sum is almost always more than the system cost. An organization ignores these costs at its peril. An organization that does not take into account all six components can easily make an error that ultimately costs hundreds of millions or even billions.

Employ a flexible analytic architecture that takes advantage of the strengths of both data warehouse technology and Hadoop. When a new technology arrives on the scene, like Hadoop, there is often the tendency to believe it provides a better solution for every problem. We believe that Hadoop will have a significant role in every large analytic data environment and we have described three important use cases for it: refining data, providing an initial landing zone for enterprise data and providing online access to a data archive. While Hadoop is more cost effective than a data warehouse platform in certain analytic applications, some of the data in these scenarios may still need to go into a data warehouse. And based on our cost analysis, data warehouse technology is clearly the better platform on which to build an enterprise data warehouse. WinterCorp sees many large organizations creating an overall analytic architecture that includes both platforms as options with the opportunity to deploy each in situations where it is best suited.

Effective analytics are fast becoming a critical success factor for the entire organization. Our cost framework shows that an organization that chooses the right solution for each analytic application can be better off by a very large factor in terms of what it spends to accomplish a given analytic objective. Ultimately, the decision on how to implement big data analytics may mean the difference between organizational success and failure.

Appendix: Additional Cost Factors To Consider

The WinterCorp TCOD framework takes into consideration the largest and most easily estimated costs in implementing a big data analytic solution. While the following costs are not reflected in the TCOD, they could influence the choice of a solution depending on an organization's specific requirements. Some of these costs may be addressed in later versions of the framework as additional data becomes available:

- Cost of maintaining data integrity – Maintaining data integrity is a key requirement in many data warehouse applications. In Hadoop, the system does not manage data according to a data definition or constraints (e.g., business rules). One can specify a schema for a file, but there is no guarantee that this schema accurately describes the data as stored. Identifying and correcting problems caused by inconsistencies between the schema and the actual data could require additional time and effort in Hadoop. In addition, any business rules concerning the data would have to be enforced by the applications, not by Hadoop. Data warehouse platforms will automatically enforce the data definitions and business rules built into the schema, relieving application programmers of this burden.
- Cost of preparing simple queries – For simplicity, the effort required to express simple queries in HiveQL is assumed to be similar to the effort required to express them in SQL. In addition, many graphical query and data visualization tools will generate either HiveQL or SQL as needed. Thus, for simple queries, we make the assumption that there is not a major cost differential in the difficulty of developing the query, and do not include the cost of preparing simple queries in the TCOD calculations. Note that in Hive 2.0, a subset of SQL is supported in addition to HiveQL, thus expanding the set of tools that can be used for query with Hive as well as the set of queries that can be submitted to Hive.

- System cost of executing queries – We have ignored the cost of executing queries, although there may be significant performance differences between the cost and time to execute queries on Hadoop vs. a data warehouse platform.
- Apache Pig – Apache Pig is a software system available with Hadoop that enables the development of applications in a higher-level language than Java/MapReduce. Analyses and analytic applications that can be written in Pig Latin (the language supported by Pig) should incur a lower development cost than those written in Java/MapReduce.
- Costs resulting from other key capabilities not available on Hadoop, or not available with the same level of function and automation compared to data warehouse platforms. Examples here include:
    - Workload management – Workload management enables the system to process multiple classes of work at different service levels.
    - Business continuity costs, including the cost of system and data recovery when component or system failures result in an interruption of service, loss or work, or loss of data integrity.
    - Data compression – Data compression can result in a large savings in system capacity and cost. In our current big data examples, we assume that data compression is always two (2). In fact, data compression ratios vary widely with the platform and the data. In addition, the ease of using data compression varies considerably with the platform.
    - The cost of security and the ability to control access to data at the column and row level in Hadoop.
    - The cost of making schema changes (data warehouse technology) vs. programming changes (Hadoop) to reflect changing business requirements, data sources and data structures.

There are several other aspects of the TCOD assumptions that could affect the overall five-year cost of data for a specific organization:

- System prices – We use average list prices by category and our assumed illustrative discounts; organizations will use actual prices and discounts offered by specific vendors on specific products.
- Cost and timing implications of upgrades to the analytic system platform – EDW customers typically view the EDW as a production platform. Therefore, the stability of the platform, including high availability and consistent performance, is a key operating requirement. We assume annual upgrades, but many organizations will choose to upgrade less frequently and buy extra capacity initially and at each upgrade point. In addition, DW customers typically do not pay for maintenance and support on an upgrade on the DW platform for the first year after the upgrade.
- Salary costs – We are using national averages. Personnel costs may be higher or lower in some areas.
- Staffing for system and data administration is not linked to system capacity, data volume or the pace of change in data requirements in the TCOD framework. In reality, an organization may need additional administrative staff as the system grows in size or as the pace of new or changing requirements increases.
- Cost of Hadoop development – We assume that the skill level and staffing cost for development in the Hadoop environment is that of a Hadoop Developer. Some organizations may also choose to utilize one or more Hadoop Subject Matter Experts, at a higher annual cost, in addition to Hadoop Developers. On www.indeed.com a Hadoop Subject Matter Expert has an average salary 60% higher than a Hadoop Developer.
- System capacity – The number of TB of user data alone does not always determine system capacity. If an analytic environment requires extensive processing capability, the organization may need additional processing power (that is, additional nodes in the data warehouse platform or Hadoop cluster). Our

assumptions make no explicit assumption about workload. We assume that the workload is the average workload built into the vendor's standard system price.

New Products. In addition, a new class of commercial products is emerging which proposes to simplify query, analysis or analytic application development with Hadoop. These products provide some degree of database function and SQL support for data stored in Hadoop. This is accomplished via a separate software layer between the user and Hadoop. Examples of these products are:

- Cloudera Impala
- EMC HAWQ Technology
- IBM BigSQL
- Teradata SQL-H

Note, however, that these products involve license fees and may consume additional hardware resources. In addition, the methods of query execution on Hadoop and on a data warehouse platform will be different. There are likely to be large differences in query performance between the data warehouse platform and the Hadoop platform, even for the same query on the same data. Once sufficient information is available, it would be desirable to extend the TCOD framework to represent such products, each of which may have a significantly different cost and performance profile.

Data License or Subscription Costs. A final point to keep in mind is the license or subscription cost for the data itself – It is increasingly common for analytic data applications to use at least some data that is acquired at a price from third-party data suppliers. In general, the acquisition cost of this data will be the same, whether it is stored on Hadoop or on a data warehouse platform. For that reason, we did not build this cost into our framework. But if you are planning an initiative in which data is acquired from a third party, you will want to build this cost into your own estimate of TCOD.

WinterCorp is an independent consulting firm expert in the architecture and scalability of big data and analytic database solutions.

Since our founding in 1992, we have architected solutions to some of the largest scale and most demanding big data and data warehouse requirements, worldwide.

We help technology users define their requirements; architect their solutions; select their platforms; and, engineer their implementations to optimize business value. We create and conduct benchmarks, proofs-of-concept, pilot programs and system engineering studies that help our clients manage technical risk, control cost and reach business goals.

Our seminars and structured workshops help client teams establish a shared foundation of knowledge and move forward to meet their challenges in big data and analytic database scalability, performance and availability.

We're expert with SQL, MapReduce and Hadoop—with structured data, unstructured data, and semi-structured data—with the products, tools and technologies of data analytics in all its major forms.

With our in-depth knowledge and experience, we deliver unmatched insight into the issues that impede scalability and into the technologies and practice that enable business success.

WinterCorp

245 FIRST STREET, SUITE 1800
CAMBRIDGE MA 02145
617-695-1800

visit us at www.wintercorp.com

V1.08 - 8/5/13