IDERA® | Data Architecture Whitepaper

MODERN DATA ARCHITECTURE

BY W H INMON

DATA WAREHOUSE

Data warehouse is an established concept and discipline that is discussed in books, conferences and seminars. Indeed data warehouses are a standard feature of modern corporations. Corporations use data warehouses to make business decisions every day. In a word, the data warehouse represents “conventional wisdom” and is a standard part of the corporate infrastructure.

ENTER BIG DATA

Into this world comes a new technology – Big Data. In some ways Big Data competes (or thinks that it competes) with data warehousing. Indeed there are some similarities between Big Data and a data warehouse. Big Data and data warehouses both hold data electronically. Big Data and data warehouses both hold lots of data. Big Data and data warehouses both hold data that can be used for decision making. So it is natural for vendors of Big Data to proclaim that with Big Data you don’t need a data warehouse. At least that is the impression that many Big Data vendors seem to give.

DO YOU NEED A DATA WAREHOUSE WHEN YOU HAVE BIG DATA?

But is that impression correct? Is it true that with Big Data you don’t need a data warehouse? This paper will explore this issue. First off, what is a data warehouse? From the beginning, the accepted definition of a data warehouse is a collection of data that is:

1. Subject oriented
2. Integrated
3. Time variant
4. Non-volatile

This definition of data warehouse is widely quoted as the definition of what is a data warehouse. (See BUILDING THE DATA WAREHOUSE, John Wiley, originally published 1991.)

The definition of Big Data is not quite as clear. Indeed there are different interpretations as to what is meant by “Big Data”. But for the purposes of this paper the following definition of Big Data will be used.

Big data:

1. Encompasses very large volumes of data
2. Is stored on affordable storage
3. Is stored in an unstructured manner
4. Is managed using the “Roman census” technique.

(For an in depth discussion of this definition, refer to the book BIG DATA – A PRIMER FOR THE DATA SCIENTIST, Elsevier Publishing 2014.)

AN ARCHITECTURE

Data warehouse is an architecture. Data warehouse requires a discipline to build and store. A data warehouse can be stored on a variety of media. The essence of a data warehouse is integrity of data. Another way of thinking of a data warehouse is that a data warehouse is a single version of the truth. The data that enters a data warehouse is carefully crafted and vetted. The data found in a data warehouse is data that is used for the most basic decisions the corporation makes.

Traditionally the data entering a data warehouse is integrated by means of technology called "ETL" (extract/transform/load). Data typically starts off in an application and is recast into a singular, integrated corporate format when it is placed inside a data warehouse.

The essence of a data warehouse is an architecture of integrity of data.

A TECHNOLOGY

Big Data is a technology. Big Data is capable of storing a large amount of data. Big Data is a physical media. In Big Data, there are storage mechanisms that cause data to be written and then sought when desired.

There is a fundamental difference between a technology and an architecture. Analogically speaking, an architecture is like the time of day and a technology is like the clock that keeps the time of day. While the time of day is certainly related to the clock that keeps the time of day, there is nevertheless a fundamental difference between an architecture and a technology. The fact that it is 2:36 pm in Orange County, California is quite different than the fact that the time is kept on a Rolex or a Seiko watch. A Rolex in Orange County can show 4:13 pm when it is actually 2:36 pm. Just because a Rolex is a fine timepiece does not mean it has the right time. And it is still 2:36 pm in Orange County regardless of whether a wristwatch agrees or not. So it is seen that there is a difference between the actual time and the clocks that keep the time.

The time of day is the time of day regardless of what a Rolex says, and one Rolex may show one time and another Rolex may have another time.

There is the same difference between an architecture and a technology. You can put a data warehouse on Big Data or on standard storage technology. It is still a data warehouse wherever it is located. Or you can put any data that is not a data warehouse in Big Data or on storage technology.

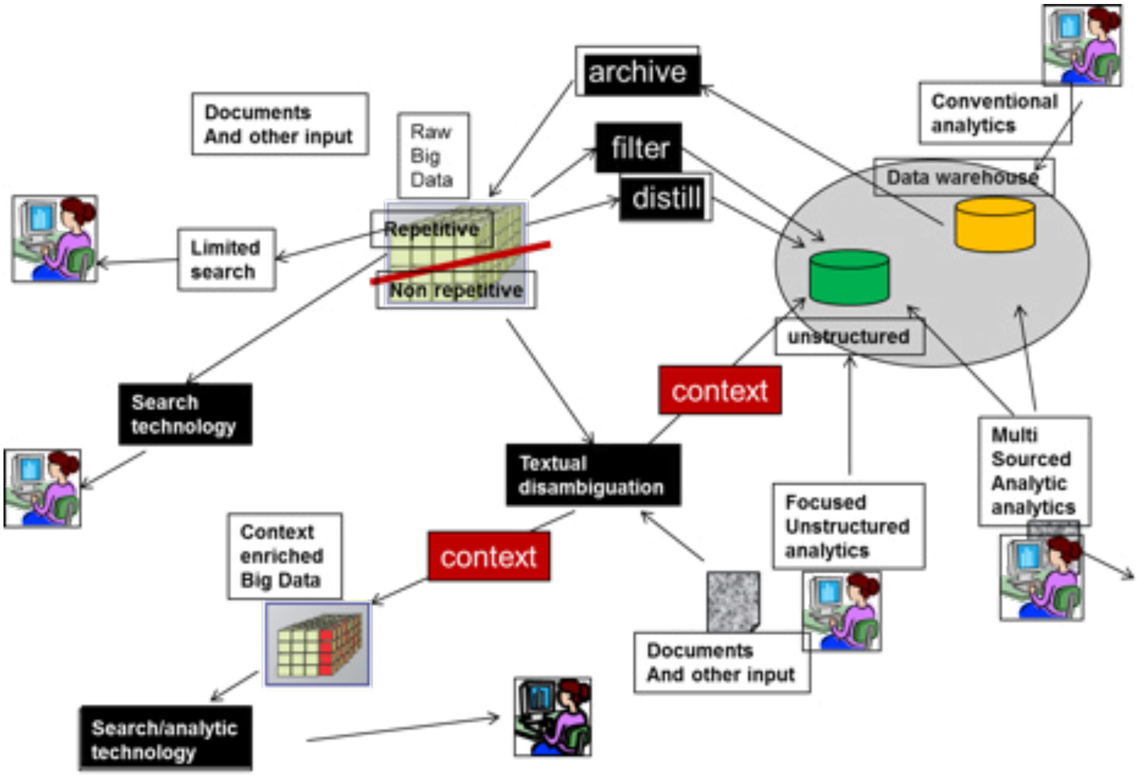
There is no competition between Big Data and a data warehouse. They are entirely different things.

BIG DATA **AND** A DATA WAREHOUSE
CAN COEXIST AND WORK IN HARMONY

HARMONIOUS COEXISTENCE

Despite the confusion that is sown by the vendors of Big Data, there is a need to understand how Big Data and a data warehouse can coexist. There is a need from an architectural standpoint to have a "big picture" that outlines how Big Data and a data warehouse can coexist and work in harmony and in a constructive manner.

That architectural rendition is seen in this figure:



REPETITIVE/NON-REPETITIVE DATA

The figure – general architecture – shows lots of major architectural features. The first major architectural feature shown is that Big Data is divided into two major subdivisions – repetitive occurrences of data and non-repetitive occurrences of data.

Repetitive occurrences of data consist of data where the same structure of data is repeated many times. There are many different examples of repetitive data. Typical repetitive data consists of log tape records, telephone call record detail records, click stream data, metering data, meteorological data, and so forth. In repetitive data, the same structure of data occurs over and over again. In many cases repetitive data is machine-written data or is produced by analog processing.

Non-repetitive data also has many examples. Some examples of non-repetitive data include email, call center conversations, survey comments, help desk conversations, warranty claim data, and so forth. In non-repetitive data it is only an accident if the same data or the same structure of data ever occurs twice. In almost every case, non-repetitive data is textual-based data that was generated by the written or the spoken word.

THE “GREAT DIVIDE”

The difference between repetitive data and non-repetitive data in Big Data has been called the “Great Divide”. There is a TREMENDOUS amount of difference between the ways that these two types of data need to be handled. The storage of the data, the writing of the data, and the reading of the data all require a very different approach and very different technology.

DATA MODELING

One of the interesting differences between repetitive data and non-repetitive data is in terms of how the data is modeled. Repetitive data is typically modeled by an ERD (entity relationship diagram) data model. Non-repetitive data is modeled in an entirely different manner by the usage of taxonomies and ontologies.

With an ERD the designer is free to change the data to fit the model. But with taxonomies and ontologies, the base data NEVER changes. As a consequence, if there is a need to make changes, it is the taxonomy or ontology that changes, not the base data.

Both types of data models can be (and usually should be) built generically. There is very little difference between the models built for an industry. As a consequence, generic models – at least as a starting point – are strongly advised.

TEXTUAL DISAMBIGUATION

The typical path for non-repetitive data to be handled and managed is through the passage of the non-repetitive data through technology known as “textual disambiguation”. Non-repetitive data is read and reformatted and – more importantly – contextualized. In order to make any sense out of the non-repetitive data, it must have the context of the data established. The job of textual disambiguation is to derive and identify the context of non-repetitive data. In many cases the context of the non-repetitive data is MORE important than the data itself. In any case, non-repetitive data cannot be used for decision making until the context has been established.

CONTEXT ENRICHED BIG DATA

Once the context of non-repetitive data has been established, the data can be transported to one of two places. The contextualized data can be sent to the analytical environment and combined with other data warehouse data, or the contextualized data can be sent back to the Big Data environment. If the contextualized data is sent back to the Big Data environment it is sent back as “context enriched” data.

It is possible that there is so much contextualized data that it cannot be sent to the data warehouse environment because of the sheer volume of data. If, however, the contextualized data is sent to the classic data warehouse, the processing that takes place on it can be done with standard analytical tools such as Tableau, Qlik, Business Objects, SAS, Excel, and so forth.

TWO KINDS OF DATA IN THE DATA WAREHOUSE

Note that if contextualized data is sent to the data warehouse, it is stored in a special place in the data warehouse. The data warehouse ends up having two kinds of data inside of it – data whose source is traditional transaction-based structured data, and data whose source is unstructured data. Even though the data warehouse is housed in a classic relational structure, the sources of data in the data warehouse are drastically different. For that reason then, the classic data warehouse ends up having two distinct types of data – transaction-based, structured data and unstructured, contextualized data.

One of the really nice things about the two types of data in the data warehouse is that because all the data arrives in a structured relational format, the data can be freely mixed and matched, and joins and analysis across the different data types can be done.

A NEW TYPE OF ANALYTICAL PROCESSING

The ability to join the different types of data gives rise to analytical processing that heretofore could not be done. Previously structured relational data could not be analytically mixed and matched with unstructured textual data. But with the advent of contextualization, these types of analysis can be done and are natural and easy to do.

REPETITIVE DATA/ DATA WAREHOUSE INTERFACE

There is another type of data found in the Big Data environment and that data is the repetitive type of data. Repetitive data does not need to be passed through textual disambiguation because repetitive data is not textual-based. But repetitive data can be placed in a data warehouse if so desired. There are two basic ways that repetitive data is passed into a data warehouse. One way is through filtering. In filtering, repetitive data is read and then after the data has been selected, the data is sent to the data warehouse. For example, the analyst may wish to find all telephone call detail records for St Louis, MO for Sept 22, 2015 and have those records sent to the data warehouse. Once the records are stored in the data warehouse, they are subject to further analysis and scrutiny.

Filtering then is merely the reading and selection of records that are then sent to the data warehouse.

The second kind of processing is distillation. Distillation is similar to filtering except distillation requires that further processing be done before the records are sent to the data warehouse. A simple example of distillation might be the counting of records that have been selected. For example, the distillation process may simply count the number of sales of items greater than $10.00 for each Wal-Mart store for the month of September 2015.

The result of both the distillation and filtering of Big Data is placed in the data warehouse. Usually the results are placed in a separate part of the data warehouse since the basis of the data found in the data warehouse is not structured, transaction-based data.

It is noted that the process of filtering and distillation of repetitive data can become quite involved. Usually the complications come in the form of handling the volume of data that is needed for analysis. In some cases, there is an enormous amount of data that must be processed. In other cases, the characteristics of the data being sought are not clearly defined and are ambiguous.

ARCHIVAL DATA TO BIG DATA

There is another flow of data that occurs between the data warehouse and Big Data. That flow occurs when it is time to archive data off of the data warehouse. Big Data – repetitive data – is used to house the archival of data found in the data warehouse that has aged.

DOING ANALYTICS

Analytics can be done all over the landscape. Classic analytical processing of transaction-based data is done in the data warehouse as it has always been done. Nothing has changed there.

But now analytics on contextualized data can be done, and that form of analytics is new and novel. Most organizations have not been able to base decision making on unstructured textual data before. And there is a new form of analytics that is possible in the data warehouse, which is the possibility of blended analytics. Blended analytics is analytics done using a blend of structured transactional data and unstructured contextualized data.

But there are many other forms of analytics that are possible as well. There is the possibility of doing analytics inside the repetitive data Big Data environment. This is where NoSQL analytical processing is a possibility. And another form of analytics is analytics of the context-enriched Big Data. A certain portion of the Big Data environment is context-enriched data which can produce its own analytical results as well.

Each of these different forms of analytical processing produces is own unique results.

DATA MARTS AND THE DIMENSIONAL MODEL

One of the interesting questions in the data architecture that has been presented is what has happened to data marts? Data marts are still part of the architectural landscape. End users still need their own individual renditions of their own data. There still is a need for dimensional technology. Data marts are still fed and fueled by the raw data that resides in the data warehouse.

WHAT ABOUT MODELING?

The basis of design for the data warehouse remains the data model and the relational structure. The basis for the design of the data mart environment is still the dimensional model. The basis of understanding non-repetitive data is the taxonomy and the ontology. And the basis of design and management of repetitive data is the schema for the occurrences of repetitive records.

Interestingly, the data model, the dimensional model, the taxonomy and the ontology are all very related but still different. They are like multiple blood-related siblings in a family. If you take a look at a group of siblings, you see that they are all either boys or girls, they all have similar skin color, you see that they have similar noses and mouths and eyes. And at the same time there are individual differences that each sibling has. They are all clearly from the same family and at the same time they are all still unique individuals.

THE SYSTEM OF RECORD

A related and interesting question is whether a system of record can be defined across the architecture that has been described. The answer is yes – it is absolutely possible to define and administer a system of record across the architecture. However, the tools that are needed and the administrative techniques are still to be discovered.

But then the more important question arises – can we achieve integrity of data across the architectural landscape?
The answer is a resounding yes. By using a consistent modeling strategy across all types of data, you can establish the foundation for data integrity.

THE REMAINING ISSUES

The net result of the architecture that has been described is that the architecture can handle an unlimited amount of data. Big Data and the data warehouse work together in a complementary, cooperative manner. Different kinds of analysis can be done in different places. There still remain the issues that have faced the data administrators for ages:

1. How can I understand my data?
2. How can I manage my data?
3. How can I ensure the integrity of my data?
4. How can I manage the budget needed for my data?
5. What technology do I need for doing all of this?

YOU CAN ACHIEVE DATA INTEGRITY
ACROSS THE ARCHITECTURAL LANDSCAPE BY
USING A CONSISTENT MODELING STRATEGY
ACROSS ALL TYPES OF DATA

ABOUT THE AUTHOR

William Inmon of Castle Rock, Colorado, is the father of data warehouse and the developer of textual disambiguation at Forest Rim Technology. Bill has written 56 books translated into 9 languages. Bill was named as one of the ten most influential people in the history of computing by ComputerWorld in 2007.

REFERENCES

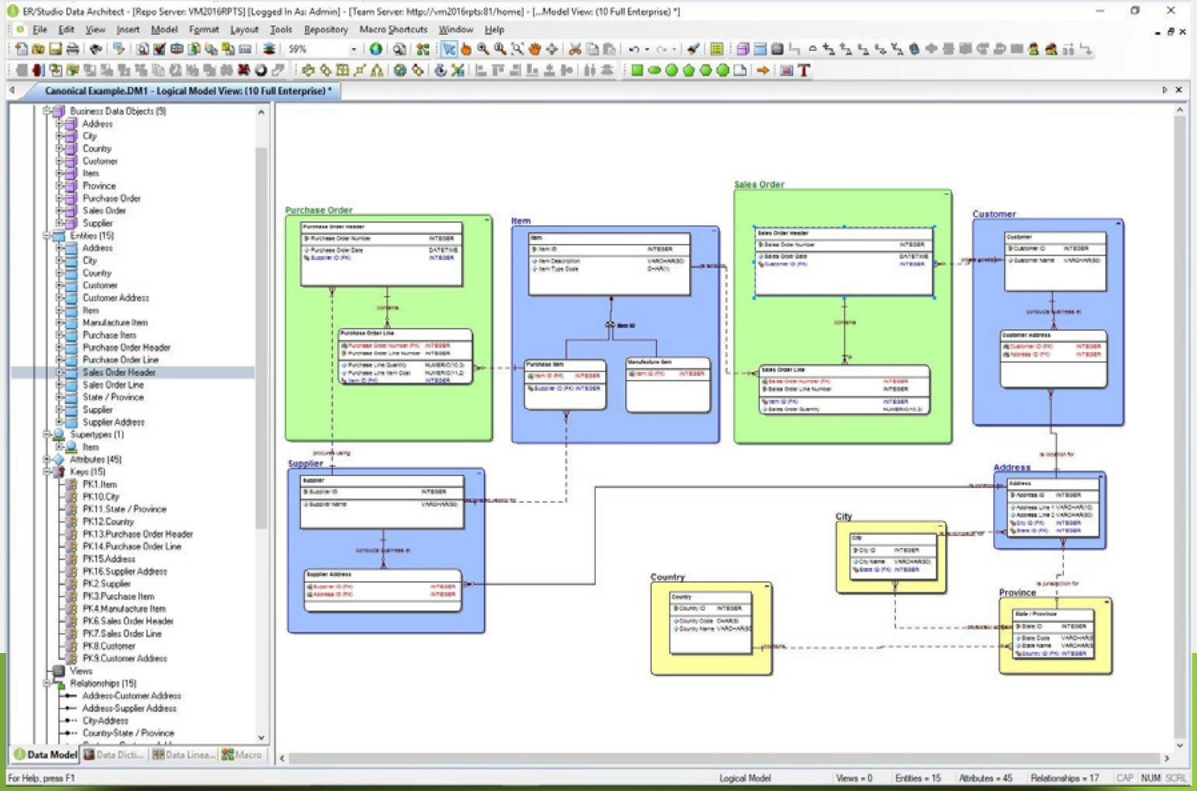
BUILDING THE DATA WAREHOUSE, John Wiley, NY, NY – the original book on data warehousing
DATA ARCHITECTURE – A PRIMER FOR THE DATA SCIENTIST, Elsevier Press, 2014 – a complete description of data architecture
THE DATA WAREHOUSE TOOLKIT, John Wiley – a guide to dimensional modeling and the building of data marts

ER/STUDIO DATA ARCHITECT

Model, Analyze and Optimize Enterprise Data

- Create effective models to build a business-driven data architecture
- Document and enhance existing databases to reduce redundancy
- Implement naming standards to improve data consistency and quality
- Effectively share and communicate models across the enterprise
- Map data sources and trace origins to enhance data lineage

Download Here


ER/Studio Data Architect - [Repo Server: VM2016RPTS] [Logged In As: Admin] - [Team Server: http://vm2016rpts:81/home] - [...Model View: (10 Full Enterprise) *]
File Edit View Insert Model Format Layout Tools Repository Macro Shortcuts Window Help
Canonical Example.DM1 - Logical Model View: (10 Full Enterprise) *
Business Data Objects (9)
Entities (15)
Supertypes (1)
Attributes (45)
Keys (15)
Views
Relationships (15)
Purchase Order
Item
Sales Order
Customer
Supplier
Address
City
Country
Province
Data Model
Data Dicti...
Data Linea...
Macro
For Help, press F1
Logical Model
Views = 0
Entities = 15
Attributes = 45
Relationships = 17
CAP NUM SCRL

IDERA.com