**LECTURE 1**

**Introduction to information architecture and design**

**Learning Goals**

1. The world is changing (actually changed), either change or be left behind.
2. Missing the opportunities or going in the wrong direction has prevented us from Growing.
3. What is the right direction?
4. Harnessing the data, in a knowledge driven economy.

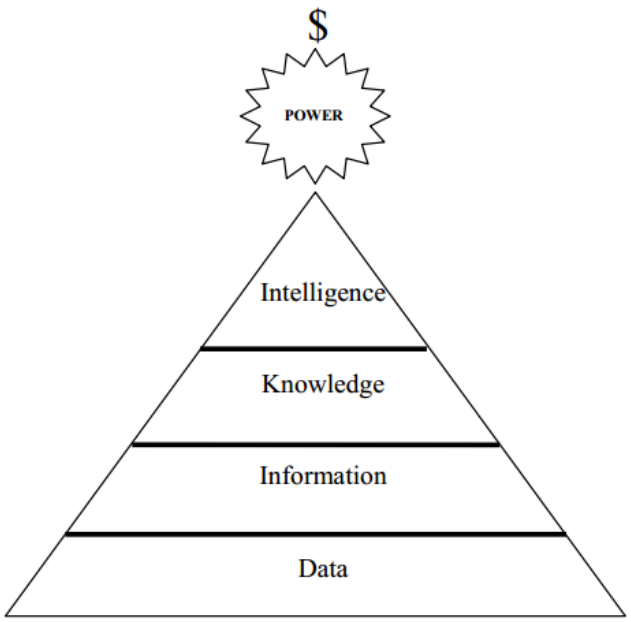
**Why information architecture and design?**

The world economy has moved from the industrial age into information driven knowledge economy. The information age is characterized by the computer technology, modern communication technology and Internet technology; all are popular in the world today. Governments around the globe have realized potential of information, as a "multi- factor" in the development of their economy, which not only creates wealth for the society, but also affects the future of the country. Thus, many countries in the world have placed the modern information technology into their strategic plans. They regard it as the most important strategic resource for the development their society, and are trying their Best to reach and occupy the peak of the modern information driven knowledge economy. What is the right direction? Ever since the IT revolution that happened more than a decade ago every government has been trying and tried to increase our software exports. But have persistently failed to get the desired results. I happened to meet a gentleman who got venture capital of several million US dollars and I asked him why our software export has not gone up? His answer was simple, "we have been investing in outgoing or outdated tools and technologies". We have also been just following India, without thinking for a moment, what India is today, started maybe a decade ago. So my next question was "what should we be doing today?" His answer was "we have captured and stored data for a long time, now it is time to explore and make use of that data". There is a saying that "a fool and his money are soon parted", since that gentleman was rich and is still rich, hence he does qualify to be a wise man, and his words of wisdom to be paid attention to.

**The Need for a information architecture and design**

“Drowning in data and starving for information”

“Knowledge is power, Intelligence is absolute power!”




**Relationship between Data, Information, Knowledge & Intelligence**

Data is defined as numerical or other facts represented or recorded in a form suitable for processing by computers. Data is often the record or result of a transaction or an operation that involves modification of the contents of a database or insertion of rows in tables. Information in its simplest form is processed data that is meaningful. By processing, summarizing or analyzing data, organizations create information. For example the current balance, items sold, money made etc. This information should be designed to increase the knowledge of the individual, therefore, ultimately being tailored to the needs of the recipient. Information is processed data so that it becomes useful and provides answers to questions such as "who", "what", "where", and "when". Knowledge, on the other hand is an application of information and data, and gives an insight by answering the “how” questions. Knowledge is also the understanding gained through experience or study. Intelligence is appreciation of "why", and finally wisdom (not shown in the figure-1.1) is the application of intelligence and experience toward the attainment of common goals, and wise people are powerful. Remember knowledge is power.

**Historical Overview**

It is interesting to note that DSS (Decision Support System) processing as we know it today has reached this point after a long and complex evolution, and yet it continues to evolve. The origin of DSS goes back to the very early days of computers.

Figure-1.2 shows the historical overview or the evolution of data processing from the early 1960s up to 1980s. In the early 1960s, the world of computation consisted of exclusive applications that were executed on master files. The applications featured reports and programs, using languages like COBOL and punched cards i.e. the COBOL an era. The master files were stored on magnetic tapes, which were good for storing a large volume of data cheaply, but had the drawback of needing to be accessed sequentially, and being very unreliable (ask your system administrator even today about tape backup reliability). Experience showed that for a single pass of a magnetic tape that scanned 100% of the records, only 5% of the records, sometimes even less were actually required. In addition, reading an entire tape could take anywhere from 20-30 minutes, depending on the data and the processing required.

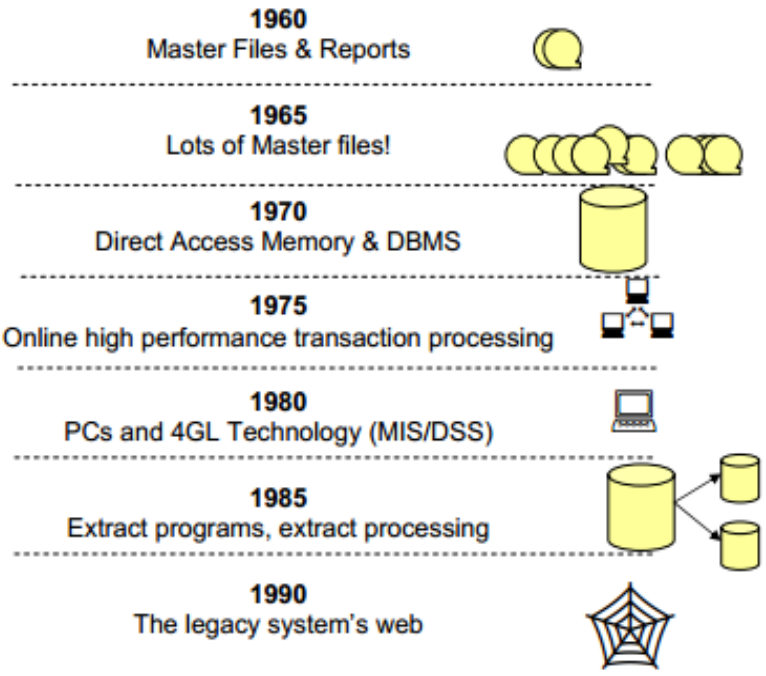



**Figure-1.2: Historical Overview of use of Computers for Data Processing**

Around the mid-1960s, the growth of master files and magnetic tapes exploded. Soon master files were used at every computer installation. This growth in usage of master files, resulted in huge amounts of redundant data. The spreading of master files and massive redundancy of data presented some very serious problems, such as:

1. Data coherency i.e. the need to synchronize data upon update.
2. Program maintenance complexity.
3. Program development complexity.
4. Requirement of additional hardware to support many tapes.

In a nut-shell, the inherent problems of master files because of the limitations of the medium used started to become a bottleneck. If we had continued to use only the magnetic tapes, we may not have had an Information revolution! Consequently, there would have never been large, fast MIS (Management Information Systems) systems, ATM systems, Airline Flight reservation systems, maybe not even Internet as we know it. As one of my teachers very rightly said, "every problem is an opportunity" therefore, the ability to store and manage data on diverse media (other than magnetic tapes) opened up the way for a very different and more powerful type of processing i.e. bringing the IT and the business user together as never before. The advent of DASD By 1970s, a new technology for the storage and access of data had had been introduced. The 1970s saw the advent of disk storage, or DASD (Direct Access Storage Device). Disk storage was fundamentally different from magnetic tape storage in the sense that data could be accessed directly on DASD i.e. non-sequentially. There was no need to go all the way through records 1, 2, 3 . . . k so as to reach the record k + 1. Once the address of record k + 1 was known, it was a simple matter to go to record k + 1 directly. Furthermore, the time required to go to record k + 1 was significantly less than the time required to scan a magnetic tape. Actually it took milliseconds to locate a record on a DASD i.e. orders of magnitude better performance than the magnetic tape.

With DASD came a new type of system software known as a DBMS (Data Base Management System). The purpose of the DBMS was to facilitate the programmer to store and access data on DASD. In addition, the DBMS took care of such tasks as storing data on DASD, indexing data, accessing it etc. With the winning combination of DASD and DBMS came a technological

solution to the problems of magnetic tape based master files. When we look back at the mess that was created by master files and the mountains of redundant data aggregated on them, it is no wonder that database is defined as a single source of data for all processing and a prelude to a data warehouse i.e. “a single source of truth”.

PC & 4GL

By the 1980s, more and new hardware/software, such as PCs and 4GLs (4th Generation Languages) began to come out. The end user began to take up roles previously unimagined i.e. directly controlling data and systems, outside the domain of the classical data center. With PCs and 4GL technology the notion dawned that more could be done with data than just servicing high-performance online transaction processing i.e. MIS (Management Information Systems) could be developed to run individual database applications for managerial decision making i.e. forefathers of today’s DSS. Previously, data and IT were used exclusively to direct detailed operational decisions. The combination of PC and 4GL introduced the notion of a new paradigm i.e. a single database that could serve both operational high performance transaction processing and (limited) DSS, analytical processing, all at the same time.

The extract program

Shortly after the advent of massive online high-performance transactions, an innocent looking program called "extract" processing, began to show up.

The extract program was the simplest of all programs of its time. It scanned a file or database, used some criteria for selection, and, upon finding qualified data, transported the data into another file or database. Soon the extract program became very attractive, and flooded the information processing environment.

The spider web

Figure 1.2 shows that a "spider web" of extract processing programs began to form. First, there were extracts. Then there were extracts of extracts, then extracts of extracts of extracts, and it went on. It was common for large companies to be doing tens of thousands of extracts per day.

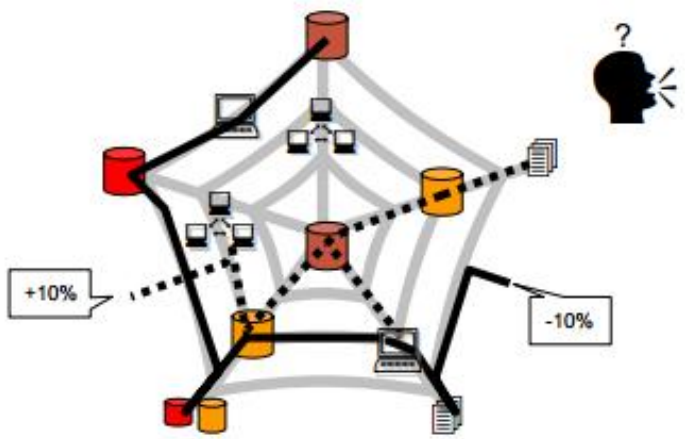
This pattern of extract processing across the organization soon became a routine activity, and even a name was coined for it. Extract processing gone out of control produced what was called

the "naturally evolving architecture". Such architectures occurred when an organization had a relaxed approach to handling the whole process of hardware and software architecture. The larger and more mature the organization; the worse was the problems of the naturally evolving architecture.

Taken jointly, the extract programs or naturally evolving systems formed a spider web, also called "legacy systems" architecture.

**Crisis of Credibility**

**What is the financial health of my company??**




**Crisis of Credibility: Who is right?**

Consider the CEO of an organization who is interested in the financial health of his company. He asks the relevant departments to work on it and present the results. The organization is maintaining different legacy systems, employs different extract programs and uses different external data sources. As a consequence, Department-A which uses a different set of data sources, external reports etc. as compared to Department-B (as shown in Figure-1.3) comes with a different answer (say) sales up by 10%, as compared to the Department-B i.e. sales down by 10%. Because Department-B used another set of operational systems, data bases and external data sources. When CEO receives the two reports, he does not know what to do. CEO is faced with the option of making decisions based on politics and personalities i.e. very subjective and non-scientific. This is a typical example of the crisis in credibility in the naturally evolving architecture. The question is which group is right? Going with either of the findings could spell

disaster, if the finding turns about to be incorrect. Hence the second important question, result of which group is credible? This is very hard to judge, since neither had malicious intensions but both got a different view of the business using different sources.

**Information architecture and design – Part II**

**Learning Goals**

1. Data recording and storage is growing.
2. History is excellent predictor of the future.
3. Gives total view of the organization.
4. Data recording and storage is growing.
5. Intelligent decision-support is required for decision-making.

**Why information architecture and design?**

Moore's law on increase in performance of CPUs and decrease in cost has been surpassed by the increase in storage space and decrease in cost. Meaning, it is true that the cost of CPUs is going down and the performance is going up, but this is applicable at a higher rate to storage space and cost i.e. more and more cheap storage space is becoming available as compared to fast CPUs.

As you would have experienced, when you (or your father's) briefcase seems to be small as compared to the contents carried in it, it seems a good idea to buy a new and larger briefcase. However, after sometime the new briefcase too seems to be small for the contents carried. On the practical side, it has been noted that the amount of data recorded in an organization doubles every year and this is an exponential increase.

**Reason-1: Data Sets are growing**

| **How Much Data is that?** | | |
|---|---|---|
| 1 MB | $2^{20}$ or $10^{6}$ bytes | Small novel – 31/2 Disk |
| 1 GB | $2^{30}$ or $10^{9}$ bytes | Paper rims that could fill the back of a pickup van |
| 1 TB | $2^{40}$ or $10^{12}$ bytes | 50,000 trees chopped and converted into paper and printed |
| 2 PB | 1 PB = $2^{50}$ or $10^{15}$ bytes | Academic research libraries across the U.S. |
| 5 EB | 1 EB = $2^{60}$ or $10^{18}$ bytes | All words ever spoken by human beings |

**Quantifying size of data**

1. Size of Data Sets are going up .
2. Cost of data storage is coming down .
3. Total hardware and software cost to store and manage 1 Mbyte of data
4. 1990: ~ $15
5. 2002: ~ ¢15 (Down 100 times)
6. By 2007: < ¢1 (Down 150 times)

A Few Examples

1. WalMart: 24 TB (Tera Byte)
2. France Telecom: ~ 100 TB
3. CERN: Up to 20 PB by 2006 (Peta Byte)
4. Stanford Linear Accelerator Center (SLAC): 500TB

**A Ware House of Data is NOT a Data Warehouse**

Someone says I have a data set of size 1 GB so I have a DWH can you beat this? Someone else says, I have a data set of size 100 GB, can you beat this?
Someone else says, I have a 1 TB data set, who can beat this?

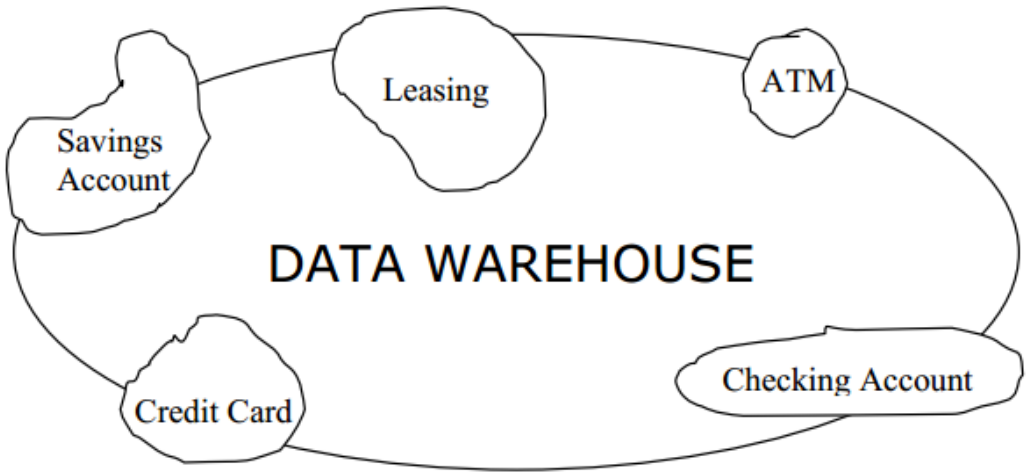
Who has a data warehouse? Not enough information, it is much more than just the size, it is a whole concept, it is NOT a shrink wrapped solution, it evolves. A company may have a TB of data and not have a data warehouse; while on the other hand, a company may have 500 GB of data and have a fully functional data warehouse.

**Size is NOT Everything**

History is excellent predictor of the future
Secondly as I mentioned earlier the data warehouse has the historical data. And one thing that we have learned by using information is that, “past is the best predictor of the future”. You use historical data, because it gives you an insight into how the environment is changing. Also you must have heard that “history repeats itself”, however this repetition of history is not likely to be constant for all businesses or all events. Note that you just can’t use the historical data to predict

the future; you have to have to bring your own insight and experience to interpret how the environment is changing in order to predict the future accurately and meaningfully.

Gives total view of the organization So why would you want data warehouse in your organization? First of all a data warehouse gives a total view of an organization. If you look at the operational system i.e. the databases in most environments, the databases are designed around different lines of business. Consider the case of a Bank; a bank will typically have current accounts and savings accounts, foreign currency account etc. The bank will have an MIS system for leasing, and another system for managing credit cards and another system for every different kind of business they are in. However, nowhere they have the total view of the environment from the customer's perspective. The reason being, transaction processing systems are typically designed around functional areas, within a business environment. For good decision making you should be able to integrate the data across the organization so as to cross the LoB (Line of Business). So the idea here is to give the total view of the organization especially from a customer's perspective within the data warehouse, as shown in Figure-2.1




A Data Warehouse crosses the LoB

Intelligent decision-support is required for decision-making

Consider a bank which is losing customers, for reasons not known. However, one thing is for sure that the bank is losing business because of lost customers. Therefore, it is important, actually critical to understand which customers have left and why they have left. This will give you the ability to predict going forward (in time), to identify which customers will leave you (i.e. the bank). We are going to talk about this in the course using data mining algorithms, like clustering,

classification, regression analysis etc.However, this being another example of using historical data to predict the future. So I can predict today, which customers will leave me in the next 3 months before they even leave. There can be, and there are whole courses on data mining, but we will just have an applied overview of data mining in this course.

**Reason-2: Businesses demand intelligence**

1. Complex questions from integrated data.
2. "Intelligent Enterprise"

| DBMS Approach | Intelligent Enterprise |
|---|---|
| List of all items that were sold last month?<br><br>List of all items purchased by Khizar?<br><br>The total sales of the last month grouped by branch?<br><br>How many sales transactions occurred during the month of January? | Which items sell together? Which items to stock?<br><br>Where and how to place the items? What discounts to offer?<br><br>How best to target customers to increase sales at a branch?<br><br>Which customers are most likely to respond to my next promotional campaign, and why? |

Comparison of queries

Let's take a close look at the typical queries for a DBMS. They are either about listing the

Contents of tables or running aggregates of values i.e. rather simple and straightforward queries and fairly easy to program. The queries follow rather pre-defined paths into the database and are unlikely to come up with something new or abnormal.

**Reason-3: Businesses want much more…**

1. What happened?
2. Why it happened?
3. What will happen?
4. What is happening?
5. What do you want to happen?

These questions primarily point to what is called as the different stages of a Data Warehouse i.e.

starting from the first stage, and going all the way to stage 5. The first stage is not actually a data warehouse, but a pure batch processing system. Note that as the stages evolve the amount of batching processing decreases, this being maximum in the first stage and minimum in the last or 5th stage. At the same time the amount of ad-hoc query processing increases. Finally in the most developed stage there is a high level of event based triggering. As the system moves from stage-1 to stage-5 it becomes what is called as an active data warehouse.

**What is a DWH?**

*A complete repository of historical corporate data extracted from transaction systems that is available for ad-hoc access by knowledge workers*

The other key points in this standard definition that I have also underlined and listed below are:

**Complete repository**

All the data is present from all the branches/outlets of the business.
Even the archived data may be brought online.
Data from arcane and old systems is also brought online.

**Transaction System**

Management Information System (MIS)
Could be typed sheets (NOT transaction system)

**Ad-Hoc access**

Does not have a certain predefined database access pattern.
Queries not known in advance.
Difficult to write SQL in advance.

**Knowledge workers**

Typically NOT IT literate (Executives, Analysts, Managers).
NOT clerical workers.
Decision makers.
The users of data warehouse are knowledge workers in other words they are decision makers in the organization. They are not the clerical people entering the data or overseeing the transactions

etc or doing programming or performing system design/analysis. These are really decision makers in the organization like General Manager Marketing, or Executive Director or CEO (Chief Operating Officer). Typically those decision makers are people in areas like marketing, finance and strategic planning etc.

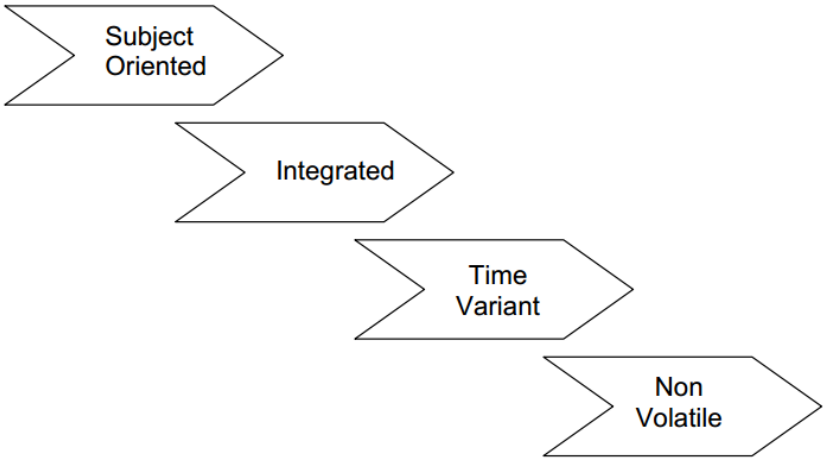
**Completeness:** There is a misnomer here, about completeness. As per the standard definition a data warehouse is a complete repository of corporate data. The reality is that it can never be complete. We will discuss this in detail very shortly.

**Transaction System:** Unlike databases where data is directly entered, the input to the data warehouse can come from OLTP or transactional systems or other third party databases. This is not a rule, the data could come from typed or even hand filled sheets, as was the case for the census data warehouse.

**Ad-Hoc access**: It does not have a certain repeatable pattern and it's not known in Advance. Consider financial transactions like a bank deposit, you know exactly what records will be inserted deleted or updated. That's in OLTP system and in ERP system. But in a data warehouse there are really no fixed patterns. Say the marketing person, just sits down and thinks about what questions he/she has about customers and there behaviors and so on and they are typically using some tool to generate SQL dynamically and then that SQL gets executed and that you don't know in advance.

Although there may be some patterns of queries, but they are really not very predictable and the query patterns may change over time. Hence there are no predefined access paths into the database. That's why relational databases are so important for the data warehouse, because relational databases allow you to navigate the data in any direction that is appropriate using the primary, foreign key structure within the data model. Meaning, using a data warehouse, does not implies that we just forget about databases.

Another view of a DWH




**Another view of a Data Warehouse**

**Subject oriented:** The goal of data in the data warehouse is to improve decision making, planning, and control of the major subjects of enterprises such as customer, products, regions, in contrast to OLTP applications that are organized around the work-flows of the company.

**Integrated:** The data in the data warehouse is loaded from different sources that store the data in different formats and focus on different aspects of the subject. The data has to be checked, cleansed and transformed into a unified format to allow easy and fast access.

**Time variant:** Time variant records are records that are created as of some moment in time. Every record in the data warehouse has some form of time variance associated with it. In an OLTP system, the contents change with time i.e. updated such as bank account balance or mobile phone balance, but in a warehouse as the data is loaded; the moment usually becomes its time stamp.

**Non-volatile:** Unlike OLTP systems after inserting data in the data warehouse it is neither changed nor removed. The only exceptions are when *false* or incorrect data gets inserted erroneously or the capacity of the data warehouse exceeded and archiving becomes necessary.