# MANAGING THE LIFECYCLE OF DATA

## Optimizing performance and lowering the cost of your enterprise data warehouse

By W.H. Inmon

## Table of Contents

# Introduction

Once there were application systems and databases. Soon there were online high performance systems with transaction processing everywhere. And with the explosion of online applications came duplication of data and lack of integration of data. Organizations soon recognized that having duplication of data and un-integrated data led to a false foundation for the purpose of making decisions. In addition, certain communities of the corporation were not having their needs for information being met. In particular the sales, marketing and financial organizations found that systems written for operational personnel did not suit their needs. These functional organizations took it upon themselves to create additional data to meet their reporting needs.

There was a crisis in information systems. There was data everywhere but there was no information. From that scenario of thirst for information that has been played out in corporations everywhere has come about the concept of a data warehouse. And with the explosion of data came a corresponding explosion in the costs of processing and data storage.

In this day and age, data warehousing is a common term in the IT profession. Organizations have spent years integrating many data sources into one common repository for organizations to utilize for reporting purposes. One of the largest problems impacting data warehouse user satisfaction occurs when lots of data is assimilated but very little data is usable or viable information for reporting.

This whitepaper will review the basics of data warehousing followed by a related discussion of Information Lifecycle Management (ILM) for data warehousing.

Information Lifecycle Management (ILM) is the recognition that data passes through its own lifecycle as it enters the corporation then is used within the corporation. At first data is used in an online manner. Then after the data gets to be a little older, data is used actively, but not as often as it once was. Then as data grows a little older and it is used even less often. Placing data warehouse data in different categories; (frequent, in-frequent and dormant) are optimal for the residency of data at different points in the lifecycle of the data.

# The Data Warehouse

The data warehouse is an integrated, historical detailed collection of data that is designed to meet the information needs of any part of the organization. The data warehouse is designed to hold data for a long period of time. The data warehouse is designed to hold data at a low level of detail, so that data can be shaped and reshaped in many ways, meeting the information needs of many different organizations. The data warehouse is designed for integrated data, where data from different sources can be merged together meaningfully, so that a true corporate picture of data emerges into viable information.

For these reasons data warehousing is a concept that has now become conventional wisdom. Data warehousing is found in every industry and is globally recognized as a business critical requirement.

## Large Amounts of Data

However, with data warehousing has come several problems. One of the major problems with data warehousing is the need to manage large volumes of data. Prior to data warehousing, capacity planning for operational systems was done in terms of megabytes and a few gigabytes. With data warehousing, capacity planning is done in terms of hundreds of gigabytes, terabytes and even petabytes of data in many cases. Even the vocabulary used for capacity planning for a data warehouse is at several orders of magnitude greater than the measurement of the volumes of data found in the systems that predated data warehousing. It is the nature of data warehouses to grow in size.

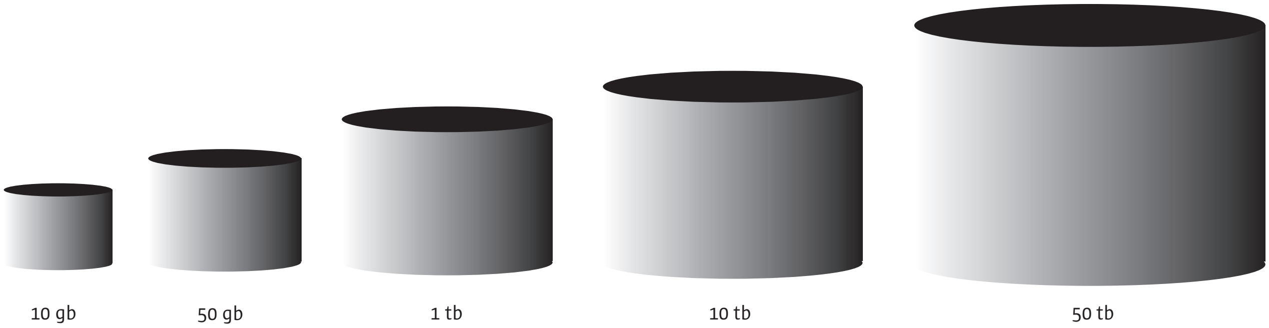
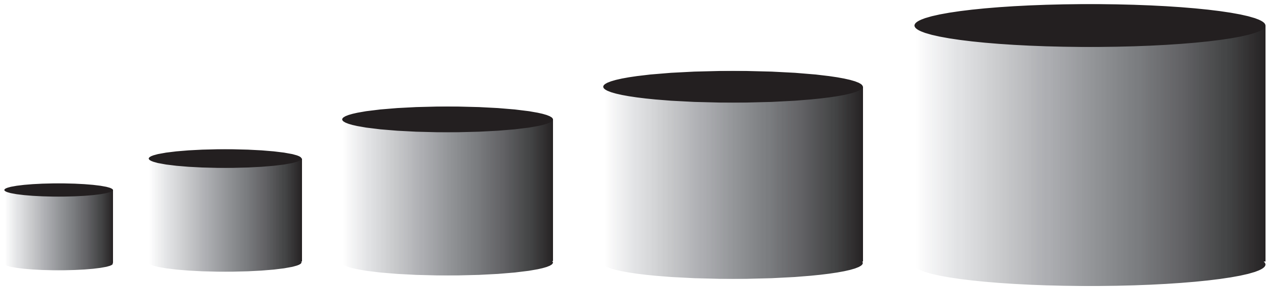


**Figure 1**

Figure 1 shows that it is the nature of data warehouses to grow in size.

The phenomenon of data warehouses growing large is ubiquitous. It occurs everywhere data warehousing exists. This ubiquitous phenomenon drives an interesting question: “Why is it that data warehousing attracts such large amounts of data?”

The reason why data warehousing attracts such large amounts of data is shown in Figure 2.



Detail x history x variety = lots of data

A simple formula explains why data warehouses grow large

**Figure 2**

Figure 2 shows the essence of data warehousing
– Integrated data
– Detailed data
– Historical data

It simply makes sense that when you take a large variety of data and multiply it by history (years and years of data) and then multiply it again by detail that the result is lots of data.

## Managing Historical Data

Consider history. For years the online programmer and systems designer have been throwing away history. Any good systems programmer knows that historical data clogs up the systems. The more a programmer can do to get rid of history, the faster the system will run. Therefore, the essence of good performance in the online environment is to jettison historical data as quickly as possible.

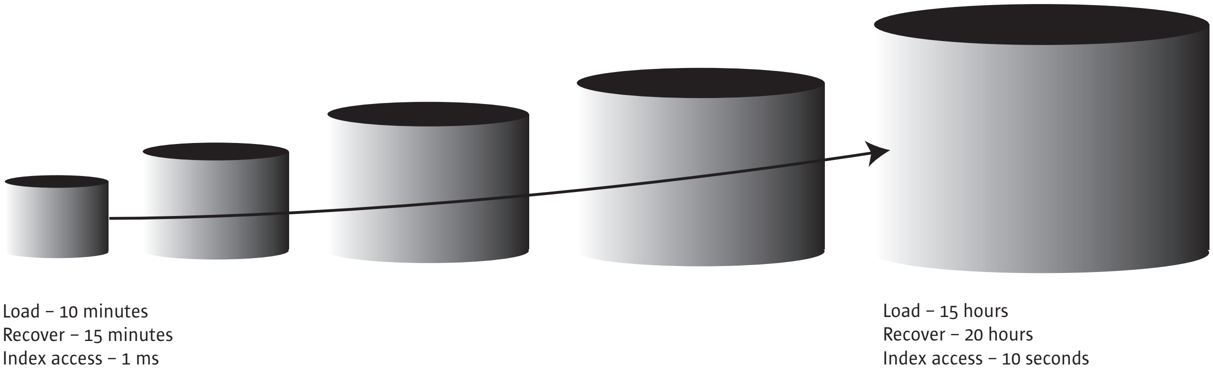
Not so in the data warehouse environment. The data warehouse environment recognizes that historical data has a very important role to play. There are many important reasons for having historical data, especially today with data retention requirements defined in legislation such as Sarbanes Oxley and accounting standards such as Basel II. So the data warehouse becomes the first place where historical data fits comfortably. And of course with historical data comes a large volume of data.

## Consider Detail

Then there is detail. The data warehouse is filled with granular, detailed data. It is this granular data that gives the data warehouse its most important attribute – flexibility. The data warehouse is able to meet all sorts of needs because the granular data can be looked at one way one day and another way another day. In addition, when new information needs arise the data warehouse waits in vigilance to be shaped in a way yet unknown. Therefore, the detailed data found in the data warehouse is one of the most fundamental aspects of the data warehouse. All of these factors combined create a volume of data never before seen in the IT industry.

## Changing the Operational Characteristics

Increasing the volume of a system is not just an exercise in accumulating data. A little recognized fact is that as volumes of data change, so change the operating characteristics of the system. Figure 3 illustrates some of these important changes in the operating characteristics of a system.



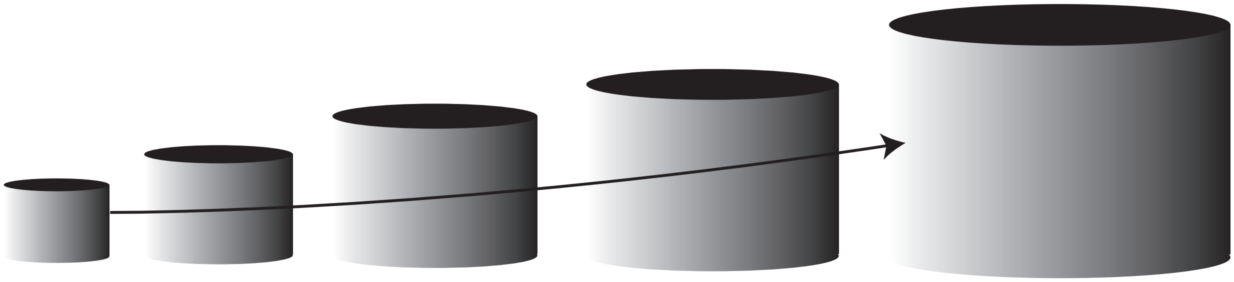
As volume increases, the activities of data management require a whole new level of concern
**Figure 3**

When the volume of data found in a system increases dramatically the time to load the system changes too. For example, a small system may take 10 minutes for the loading of the data into the system. Such a load is so small that it really doesn't require any special planning or special procedures. But when the system size goes to 30 or 40 terabytes of data, a system load may take from ten to fifteen hours. When system loads take that long, there will be a planning and coordination effort that must be carefully orchestrated.

When the system is small, recovery of the system may take 15 minutes or so. As such, not much impact is made by the considerations for recovery. But when the system is large, it may take 20 to 30 hours to recover. When recovery takes that long, careful consideration must be given to the plans for recovery and the impact on ongoing operations.

And then there is the simple indexed access to data. In a small world, much data will be found in the buffer area and can be accessed in nanosecond time. Other data will require disk access. In all, averaging everything out, indexed access to data can be done quickly. But in the case of massive amounts of data, depending on how the data is organized, as much as ten seconds on the average may be required to access a unit of data. When the 10 seconds of access time is multiplied over the number of records to be accessed, the result is a real drag on everyday system performance.

So it is seen that the mere acquisition of volumes of data greatly change system characteristics even though nothing else may have changed.


Relational database design
Partitioned parallelized database design

Database design practices change as well as operations practices
**Figure 4**

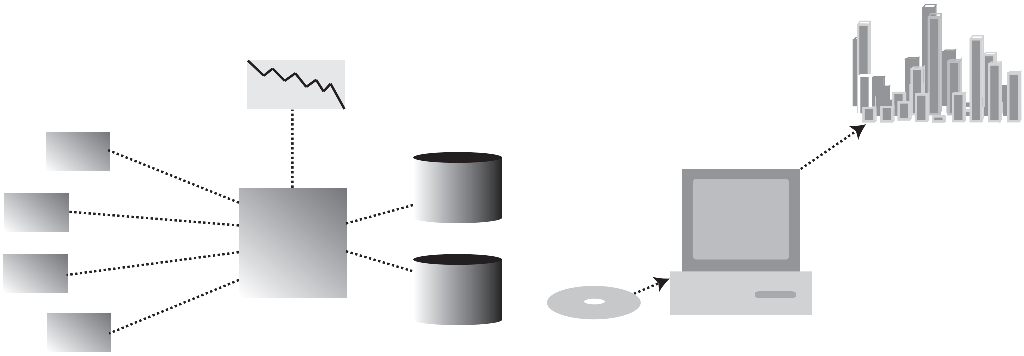
## Database Design For Large Volumes of Data

Operational processes are not the only things that are affected by a significant increase in the volume of data. Database design is affected as well. Figure 4 shows the impact of large volumes of data on database design.

Figure 4 shows that simple relational design may well be optimal for small systems. But as systems grow very large, other database design issues come into play. For example, when there are very large volumes of data, data must be partitioned or otherwise prepared for parallel access. The techniques of database design that were adequate for the small database are hardly sufficient for the large database.

## Operational and DSS

There are other factors in the treatment of large databases that must be taken into account as well. One of those factors is that of the general usage of the environment. Figure 5 shows this consideration.



Operational processing

- transaction oriented
- high performance
- high availability
- clerical support
- update processing
- limited amounts of history
- application oriented

DSS - decision support systems processing

- analytical processing
- no update
- new snapshots
- in support of management
- lots of history
- relaxed response time
- integrated data

The different kinds of processing
**Figure 5**

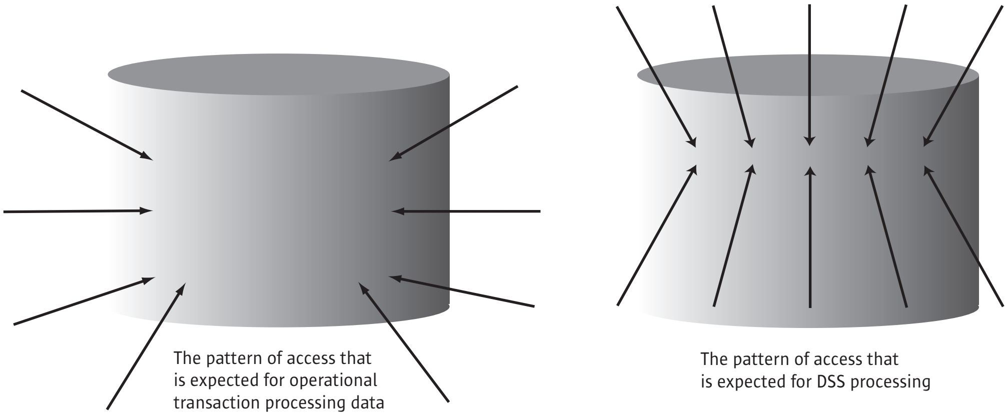
Figure 5 shows that operational processing is quite different from decision support systems (DSS) processing. Operational processing entails a lot of transactions, with high performance and high availability. Operational processing is where updates occur, typically serving the clerical community. There are limited amounts of history in the operational environment and the operational environment is typically application oriented.

The DSS environment on the other hand does not involve much, if any, transaction processing and in some cases, may have relaxed response times. The DSS environment does not support updates, it supports snapshots of data and a great deal of history. The DSS environment is integrated, rather than application oriented.

Another big difference between the two environments is that the operational environment operates on relatively small amounts of data while the DSS environment operates on very large amounts of data.

## The Basic Way That Data is Accessed on Storage

There is another very important difference between the two environments and that difference is that the operational environment operates on a base of data in a random manner while the DSS environment operates on data in anything but a random pattern of data. Figure 6 shows this important difference between the two environments.


The pattern of access that
is expected for operational
transaction processing data
The pattern of access that
is expected for DSS processing

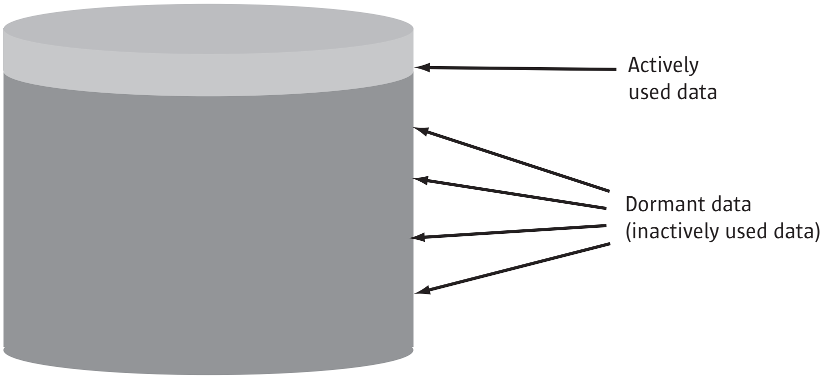
There is a fundamentally different pattern of access of storage for the different modes of data access
**Figure 6**

In Figure 6 it is seen that in the operational environment there is a random pattern of access when it comes to accessing data. One transaction comes in and wants to access part 123RTY. The next transaction comes in and wants to access part YUI998. The next transaction comes in and needs to access part HYG667. In short there is no rhyme or reason to the order in which data is accessed in the operational, transactional processing environment. Stated differently, in the operational environment there is an even probability of access for any given unit of data.

But the DSS informational environment is entirely different when it comes to patterns of access of data. The first request for data comes in for the months from March 2005 to the month of February 2006 for a customer. The next request comes in for activity about a vendor for April 2005. The next query comes in and needs to see data for a set of customers for January 2005 to December 2005. In general the location of the data being requested for analysis is very close to other data needed for analysis. Rarely is there a request for data from 1995. And rarely is there a request for data about manufacturing, for example. In fact whole sections of data are almost never accessed.

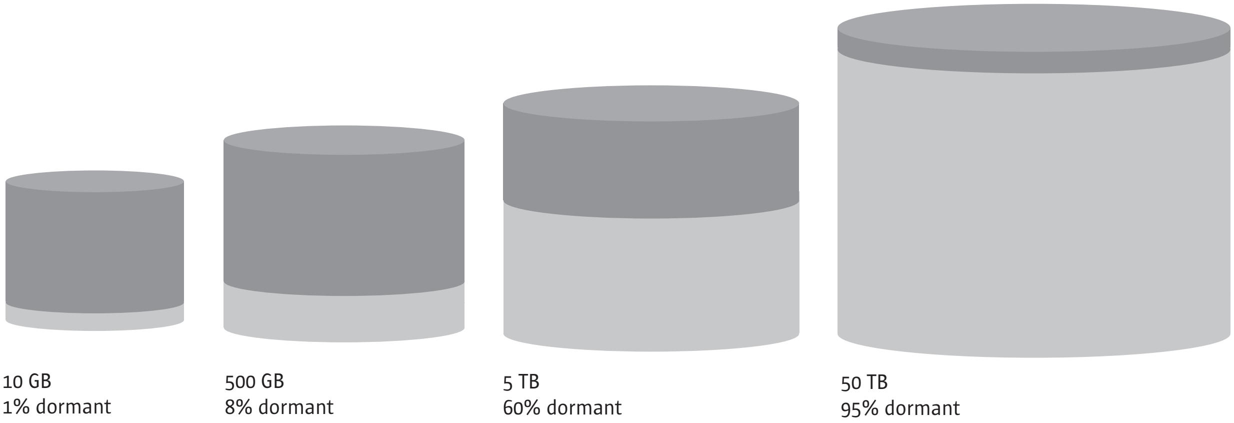
## Dormant Data

There is a phenomenon that can be called the settling of data into one of two patterns for DSS data warehouse data. Some data is very actively used and other data is very infrequently used. Figure 7 shows this phenomenon.



As volumes of data grow, the data divides itself into one of two categories
**Figure 7**

The data that is actively used is called – not surprisingly – actively used data. The data that is inactively used is called "dormant data". It is absolutely normal for dormant data to grow inside a system as the volume of data grows. Figure 8 shows this pattern.



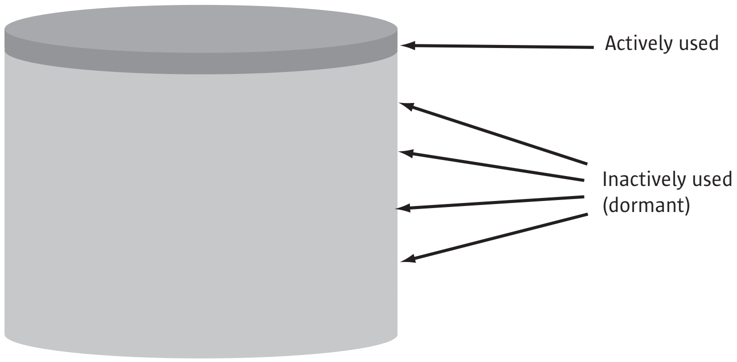
As the volume of data increases, the percentage of the data that is dormant also increases
**Figure 8**

It is seen that when the data warehouse is small that there is little if any dormant data. When an end user makes a query against this data, it is not a problem if all data is included in the query whether needed or not. Then the data warehouse grows in size. As the data warehouse grows there starts to be some noticeable amount of dormant data, say 8% to 10%. Even at this stage the dormant data is not troublesome. Then the data warehouse grows even more. At this point the data warehouse has considerable dormant data. The data warehouse continues to grow, and as it grows the percentage of the dormant data continues to climb.

Once a very large organization measured the amount of dormant data they had in their warehouse. They had over 99.5% dormant data in their data warehouse. This means that 199 out of 200 records were never accessed over a year's time. And the irony was that this large corporation continued to buy disk storage from their vendor every year. Even though they were using only a small fraction of the data they had, they continued to buy more storage rather than manage their data in a rational manner.

## The Issues of Dormant Data

There are several very important issues associated with dormant data in a data warehouse. Those issues are seen in Figure 9.



The issues of dormant data:
- high cost to keep on high performance disk storage
- negatively impacts performance

**Figure 9**

In Figure 9 it is seen that the two main issues associated with dormant data are the issues of the cost of disk storage that is not being used, and the fact that a massive amount of dormant data greatly hurts the performance of a system.
The cost of unused disk storage should be obvious. Consider two Corporations – A and B. Corporation A has 40 terabytes of data which have cost about $300 million dollars (at $750,000 per terabyte). Corporation B has 10 terabytes of high performance disk storage, 10 terabytes of lower cost, slower disk storage, and 20 terabytes of nearline storage. Corporation B has paid $7.5 million dollars for their high performance disk storage, $12 million dollars for their lower cost alternate storage, and $10 million dollars for their nearline storage. Overall Corporation B has paid approximately $30 million dollars while Corporation A has paid $300 million dollars. Each company spent this money for exactly the same volume of data.

So the cost implications are obvious. If you want to save massive amounts of money, create your data warehouse where a significant portion of the data warehouse is housed on a combination of lower cost, slower storage and nearline storage.

But there is an irony here. Where is there better performance? Corporation A (where there is nothing but disk storage) or in corporation B (where there is a combination of disk storage and nearline storage?) The irony is that there is much better performance in Corporation B, where there is a combination of storage types.

## Information Lifecycle Management for Data Warehousing

Information Lifecycle Management (ILM) is a set of business practices implemented to align the business value of information over time to the most appropriate and cost effective IT resource while ensuring secure and ready access based on user requirements.

ILM is not just about storage classes, disk and tape, or any other hardware at all; it's about cost-effective, secure, automated data management, policies and practices.

ILM solutions should provide an automated integrated storage portfolio that:

- Understands relations between data stores and their associated information sets
- Applies (predetermined) policies regarding placement, protection, access and retention of data
- Automatically migrates, protects, retains and eventually discards data according to business requirements

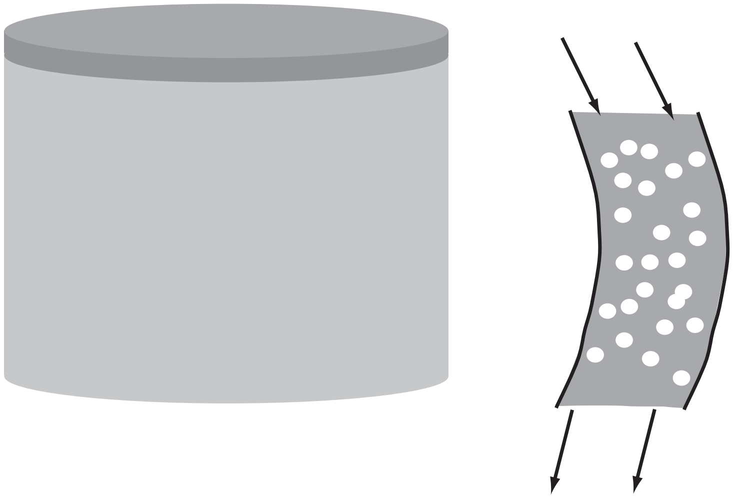
Data then passes through its own lifecycle, and as it passes through its own lifecycle, the probability of access of the data varies considerably. Very current data needs high performance and is accessed frequently. Data that is a little older has a somewhat less high probability of access. And finally data that is old has a very low probability of access. This information lifecycle matches very nicely with different forms of storage. For very current data, use high performance disk storage sometimes referred to as Tier I storage. For less current data, use less expensive, less redundant forms of storage sometimes referred to Tier II storage. And for older dormant data use even less expensive storage such as tape storage referred to as Tier III storage or nearline storage. By matching the age of data to storage tiers through corporate and regulatory data policies, some really nice things happen.

- The cost of storage goes down – dramatically
- Performance of the system goes up and gets faster as older data is pulled out of the way of younger data

ILM is the matching of data according to the age of data and the changing probability of access, to appropriate forms of storage.

## Dormant Data and Performance

In order to understand why there is better performance in Corporation B, consider Corporation A, where there are 40 terabytes of disk storage. Consider the scenario seen in Figure 10.



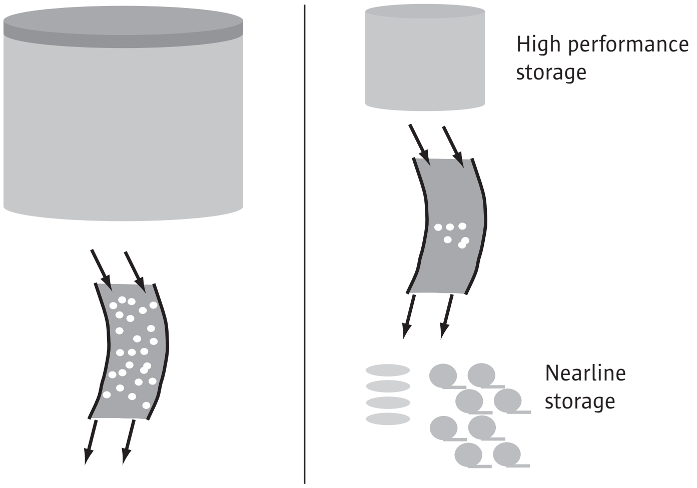
The flow of blood through an artery
– the more cholesterol we have, the harder the heart has to pump

**Figure 10**

In Figure 10 it is seen that there is active data and dormant data all mixed together. When it comes time to access almost anything, the performance is abysmal. Why? Performance is abysmal because the dormant data clogs up the arteries of the system. To use an analogy, consider an aorta of the body. In this aorta is a flow of blood. But in this aorta there is a massive amount of cholesterol. The heart has to pump really hard in order to get the blood through the artery through the cholesterol. The cholesterol is a real drag on performance.

In Company A where in disk storage unused data is freely mixed with actively used data, the system has to work really hard just to satisfy a single query because the system has to move around lots and lots of data that isn't being used. The system is like the heart. Both have a heavy and unnecessary load because of all the unused data and all of the cholesterol in the systems.

Now consider an environment where there is a separation between the two types of data – i. e., a separation between active data and dormant data. Dormant data has been placed in nearline storage. Actively used data is placed on high performance storage. Figure 11 shows this placement of data.



When the arteries are cleared of cholesterol, the flow of blood runs unimpeded and total system performance is very good
**Figure 11**

In Figure 11 it is seen that blood is passing through an artery. But there is very little cholesterol in the artery. The blood flows through unimpeded and efficiently. The heart has to pump the minimum amount of pressure in order for the system to function.

And why is this scenario so different from the previous scenarios? The answer is that there is no cholesterol – no dormant data – to impede the flow of data throughout the system.

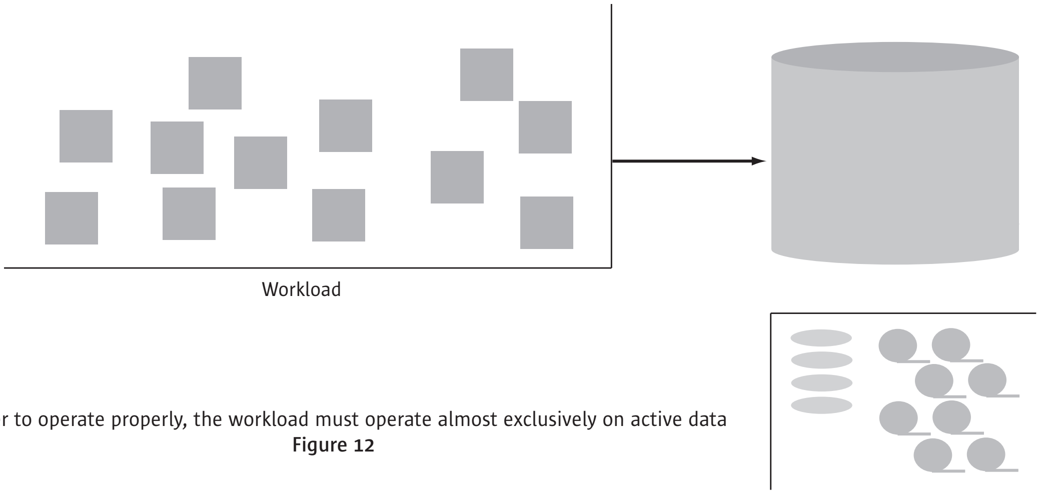
## What The Disk Vendors Say

These are the reasons why performance improves when inactive data is placed on nearline storage and across a hierarchy of storage appropriate to the lifecycle of the data. This line of reasoning is completely contrary to the disk vendors who are taught to say that performance automatically gets worse when you put data on nearline storage or slower storage. The drop in performance that the disk storage vendors are referring to is an environment where the probability of access is even – the operational environment, not the DSS data warehouse environment. Indeed, if nearline storage is used in the operational environment then performance does get worse. But in the data warehouse DSS environment there is a markedly different pattern of access of data. As long as data has been properly placed according to an ILM strategy, there is no performance degradation when dormant data is placed in either Tier I, Tier II or Tier III storage. In fact, there is a performance enhancement when dormant data is placed in a higher tier of the storage environment according to an ILM strategy.

The question then becomes – how should dormant data be placed into Tier II and Tier III storage? The simplest method used in many places is the movement of data into Tier II and Tier III storage based on date. For example, data older than 25 months is placed in Tier II and Tier III storage, as a general strategy. This crude approximation works well for the purposes of assessing when data becomes dormant. But it is a crude measurement at best.

## A Data Warehouse Monitor

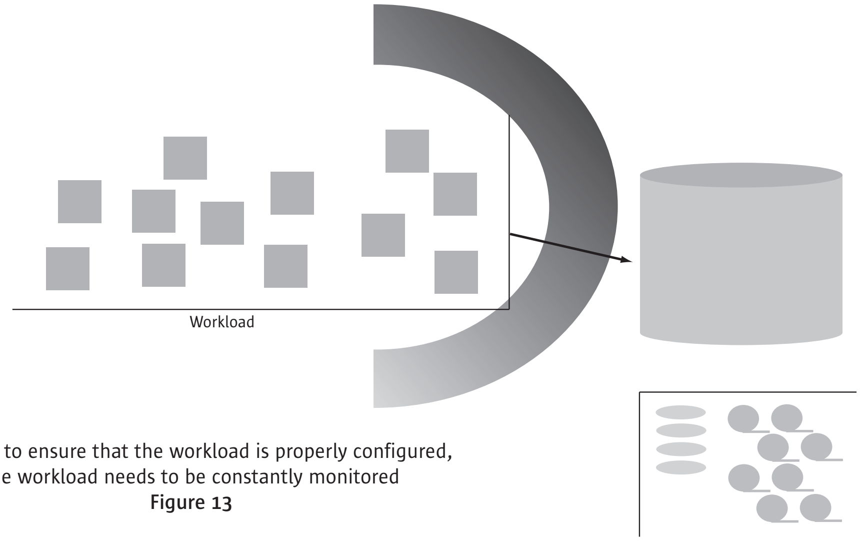
A much more sophisticated way to determine what data should and should not be placed in Tier II and Tier III storage is the use of a data warehouse monitor. A data warehouse monitor logs what data has and has not been accessed. Figure 12 shows that the data warehouse monitor looks at the workload that is being passed into and out of the data warehouse.



In order to operate properly, the workload must operate almost exclusively on active data
**Figure 12**

The data warehouse monitor uses the WHERE clauses and the results set to determine what data is being accessed. As a rule third party data warehouse monitors are preferred over the dbms vendor's monitors. The dbms vendor's monitors take up an excessive amount of overhead. The third party software vendors take up only a miniscule amount of systems resources. And the third party vendor's data warehouse monitors are more focused than the dbms vendors monitors.

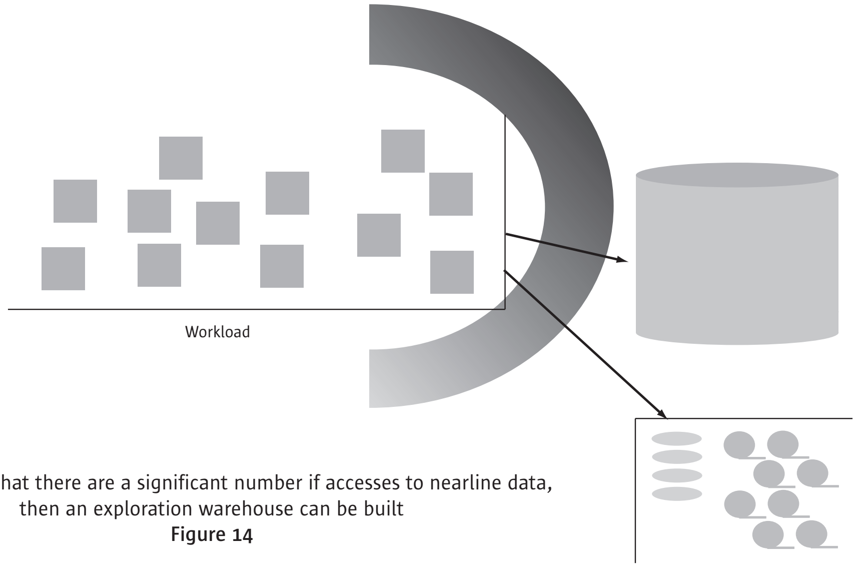
One of the advantages of a third party vendor's data warehouse monitor is that it can be turned on during peak periods of processing, as seen in Figure 13.



In order to ensure that the workload is properly configured,
the workload needs to be constantly monitored
**Figure 13**

In Figure 13 it is seen that the most important time for the data warehouse monitor to be turned on is during peak period processing, when the most critical processing is occurring.

But what if it turns out that there is quite a bit of processing going on against nearline data? Figure 14 shows this circumstance.



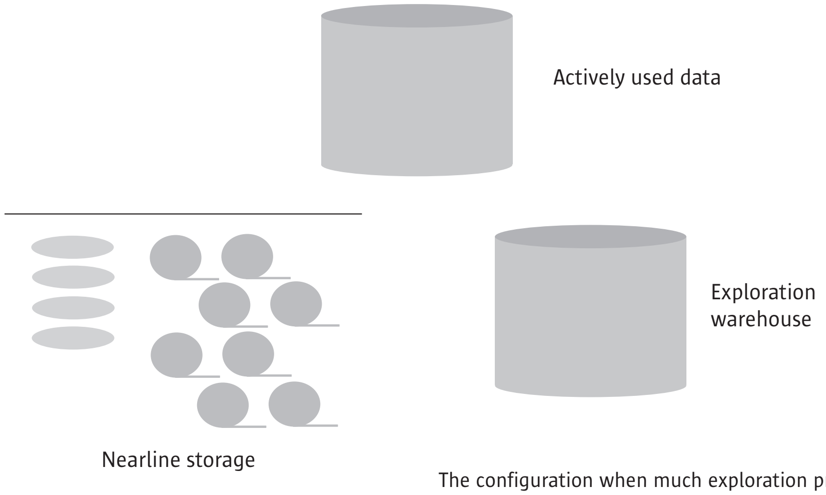
If it turns out that there are a significant number if accesses to nearline data, then an exploration warehouse can be built
**Figure 14**

Figure 14 shows that there is more than the occasional access to the nearline environment. When there is more than the occasional access to the nearline environment, performance suffers. The monitoring of the activity going through the data warehouse environment shows this occurrence.

## Exploration/Statistical Processing

Usually when there is more than infrequent access to nearline data that is a sign of other kinds of processing (other than the standard analytical processing.) Usually more than infrequent access to the nearline environment indicates that exploration processing and heavy statistical analysis is occurring. When people do heavy statistical analysis, they usually do it randomly, looking at one set of data on one occasion and looking at another set of data on another occasion. Furthermore the data looked at statistically is usually not banded by time, the way other queries are banded. Therefore when there is a lot of exploration occurring, the statistical processing is random.

In order to accommodate regular statistical processing, it is often useful to create what is termed an exploration warehouse. Figure 15 shows an exploration warehouse.

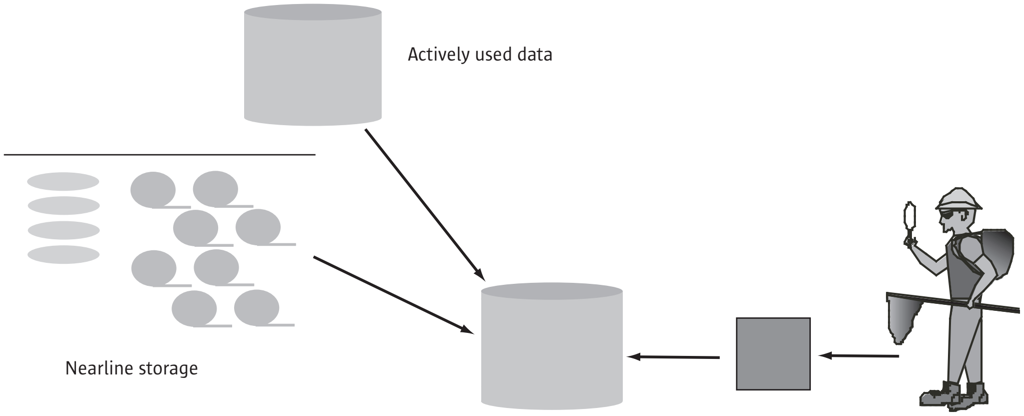


The configuration when much exploration processing is occurring
**Figure 15**

The exploration warehouse is used for statistical analysis. The exploration warehouse is usually created on a project basis. After the analysis is finished, the exploration warehouse is usually torn down. The statistician can analyze data to the finest degree without having a performance impact on the data warehouse.

In truth there are many differences between the data warehouse and the exploration warehouse. The book BUILDING THE EXPLORATION WAREHOUSE, John Wiley and Sons is recommended for further explanation.

Figure 16 shows that the exploration process draws on data found in the data warehouse and the nearline storage component.

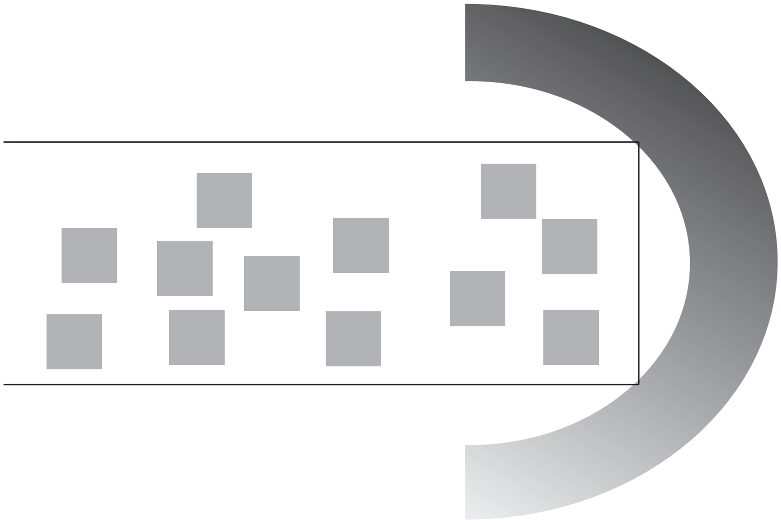


Heavy statistical analysis by explorers is done on a separate exploration warehouse facility
**Figure 16**

By drawing on data from both the data warehouse and the nearline environment, the systems programmer mitigates much of the need to go down to the nearline environment. When the analyst needs to see nearline data, the analyst sees that data as part of the exploration warehouse, rather than accessing the data directly from the nearline environment.

## Monitoring at The Row and Column Level

Monitoring then leads the organization to an understanding of what processing is occurring in the data warehouse workload. In order to be most effective, the monitoring needs to be done at the row level and the column level. Figure 17 shows this need.



Monitoring needs to be done at the row and the column level.
**Figure 17**

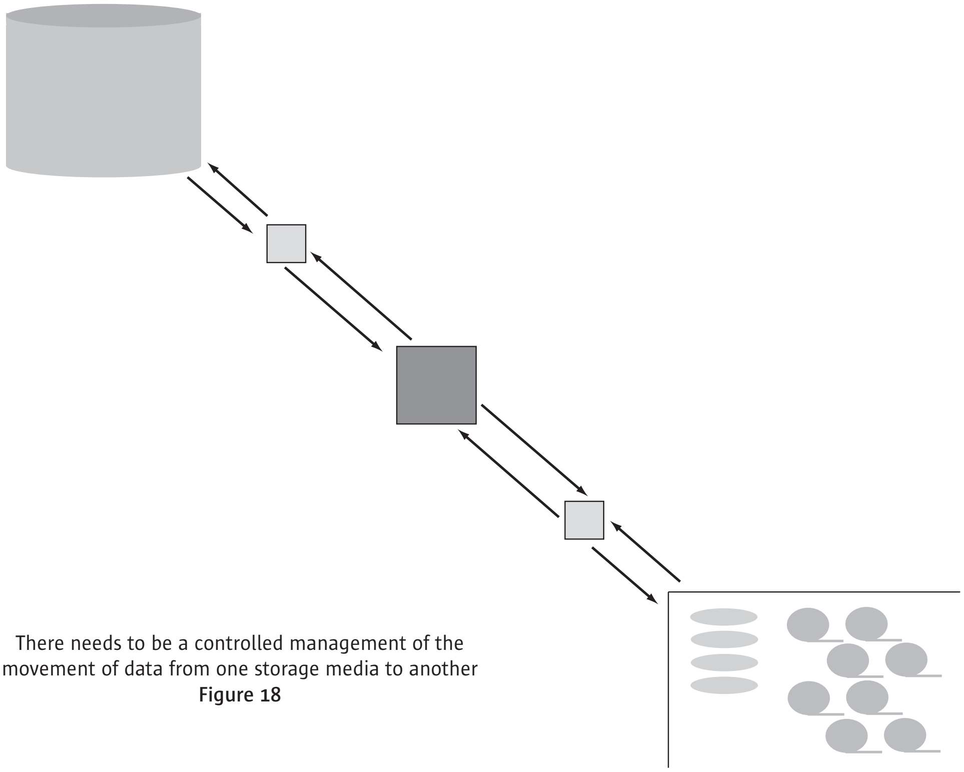
Monitoring data at the row level is obvious. Monitoring data at the row level tells what rows are being used and what rows are not being used. But monitoring at the column level may not be obvious at all.

Suppose that there are rows with the column UNIT OF MEASURE as part of the row. Now suppose that all calls to the data base do not reference UNIT OF MEASURE either as a search argument or as data to be placed in the results table. In this case UNIT OF MEASURE is extraneous and could be removed from the database with no ill effects. In other words, UNIT OF MEASURE is simply taking up space. If and when the database goes through a re-organization it might be useful to remove UNIT OF MEASURE. Without the data warehouse monitor such information would never go noticed.

## Moving Data to and from Various Tiers of Storage

One of the most important aspects of the nearline environment is the movement of data from one tier of storage to the next. In other words, how data gets to be moved from disk storage to nearline storage and back is of great importance.

Figure 18 shows the movement of data to and from the different storage media.



There needs to be a controlled management of the movement of data from one storage media to another
**Figure 18**

In Figure 18 it is seen that data is moved from disk storage to nearline storage and back the other direction on occasion.

There are many ways to accomplish this movement. One way to accomplish this movement is to manage the movement manually, as seen in Figure 19.

One way movement can be accomplished is by managing the movement of data manually
**Figure 19**

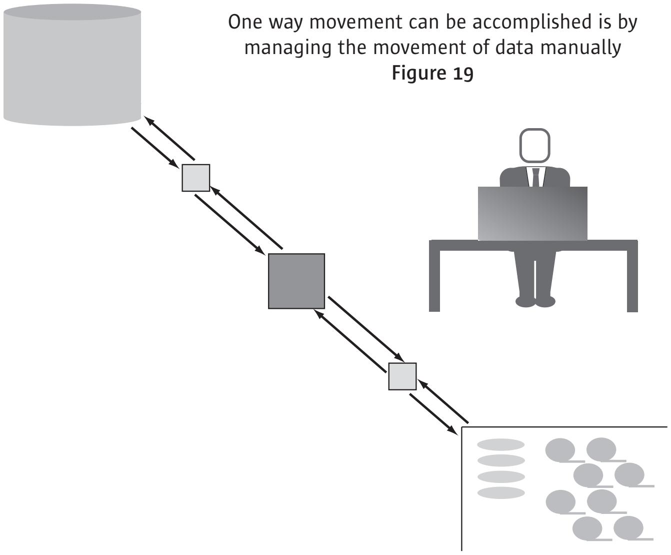
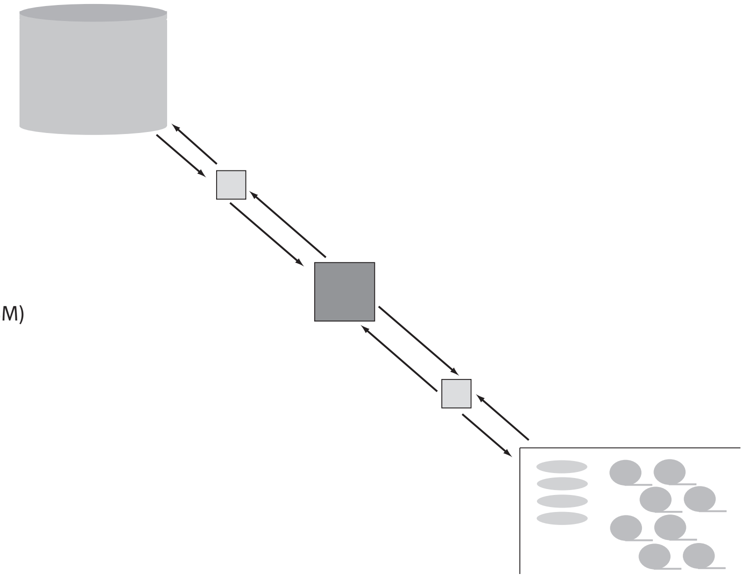


Figure 19 shows that a database administrator sits between the three levels of storage. When data ages the database administrator moves data from disk storage to nearline storage. When data is requested the database administrator moves data from nearline storage to disk storage.

While this approach may seem clumsy, there are actually a number of shops that operate in this manner quite successfully. As a long term solution, the approach leaves a lot to be desired. But on a short term basis this approach is quite workable.

A second approach to the movement of data to and from disk storage and nearline storage is to automate the movement of the data. Figure 20 shows the use of automation as a basis for the movement and management of data.

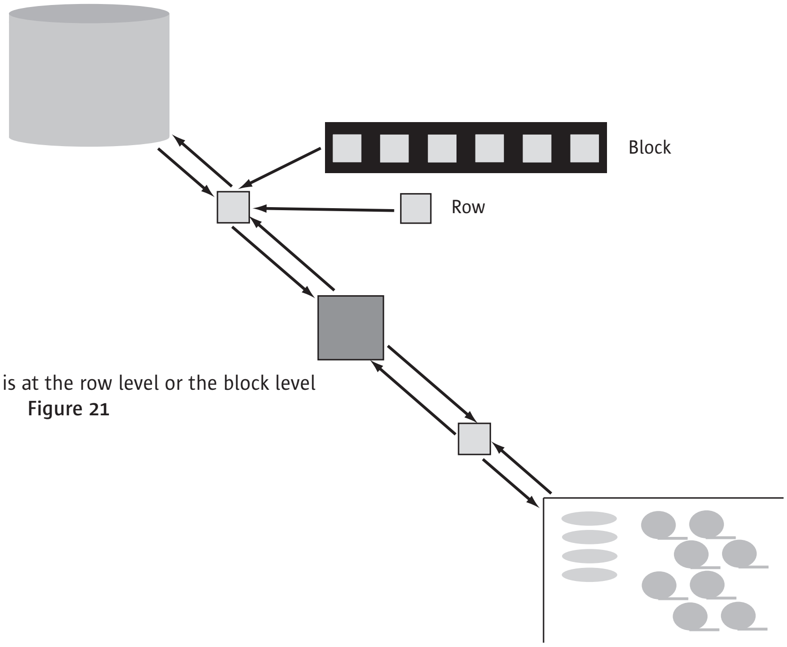


Cross media storage management (CMSM)
**Figure 20**

## Cross Media Storage Management

The technology that moves data to and from disk storage and nearline storage is called Cross Media Storage Management (CMSM). CMSM technology is software that determines when it is time to move data to and from one location to another. One example of a company that provides CMSM software is Princeton Softech.

It is interesting to note that the data that is moved from disk storage to nearline storage and back is data that is formatted at the block level or the row level. Figure 21 shows this movement of data.

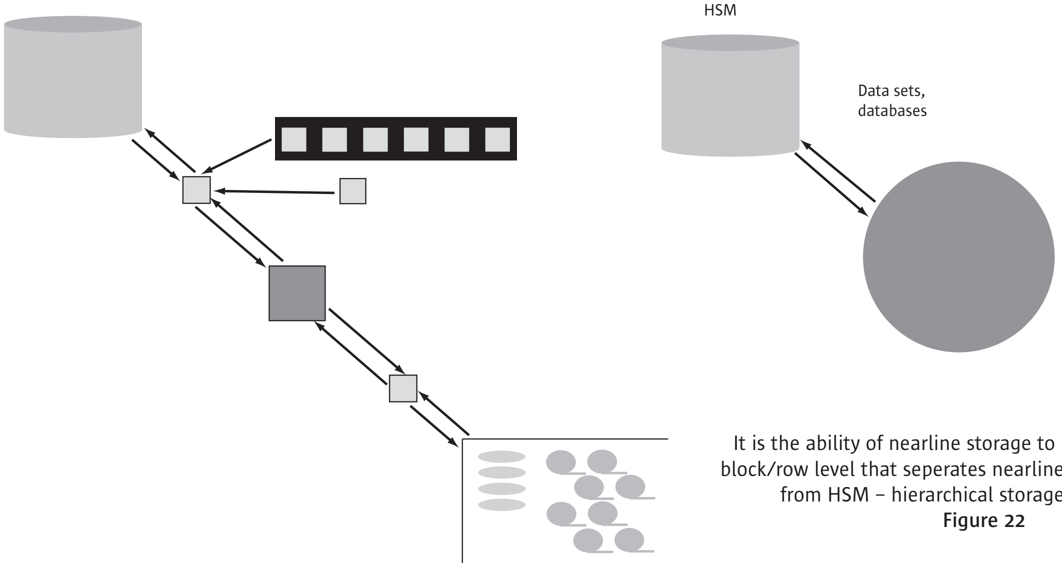


The passage of data is at the row level or the block level
**Figure 21**

The movement and management of data at the row or block level is at a very fine level of granularity. This means that entire tables and databases do not have to be moved. The low level of granularity ensures the highest degree of flexibility when it comes to the interchange of data between different levels.

## Navigating The Tiers of Storage; Hierarchical Storage Management

It is this low level of granularity of data that is transferred by the cmsm to and from different storage platforms. This separates the processing of data between tiers of storage, from what is known as Hierarchical Storage Management (HSM) processing. Figure 22 shows the difference between nearline processing and HSM.



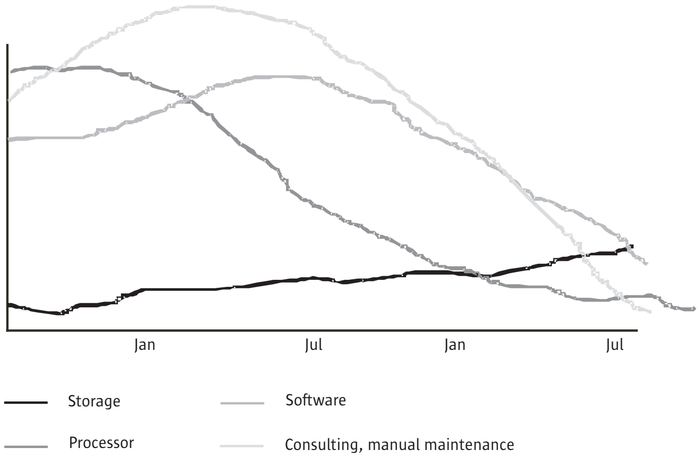
It is the ability of nearline storage to manage data at the block/row level that seperates nearline storage managment from HSM – hierarchical storage managemant
**Figure 22**

In HSM processing, entire tables of data and entire databases are moved from one level of storage to another. There are several problems related to HSM processing when it comes to managing databases and data warehouses. Some of those problems are:

- The amount of storage and the amount of resources involved with the moving of a database can be enormous
- The databases inside a data warehouse are not entirely used or are not entirely unused. Instead whole parts of a database or a table are used while other parts of the table or database are unused. For these reasons databases have to be managed at the block or row level when it comes to effective data management

## Nearline Storage and Costs

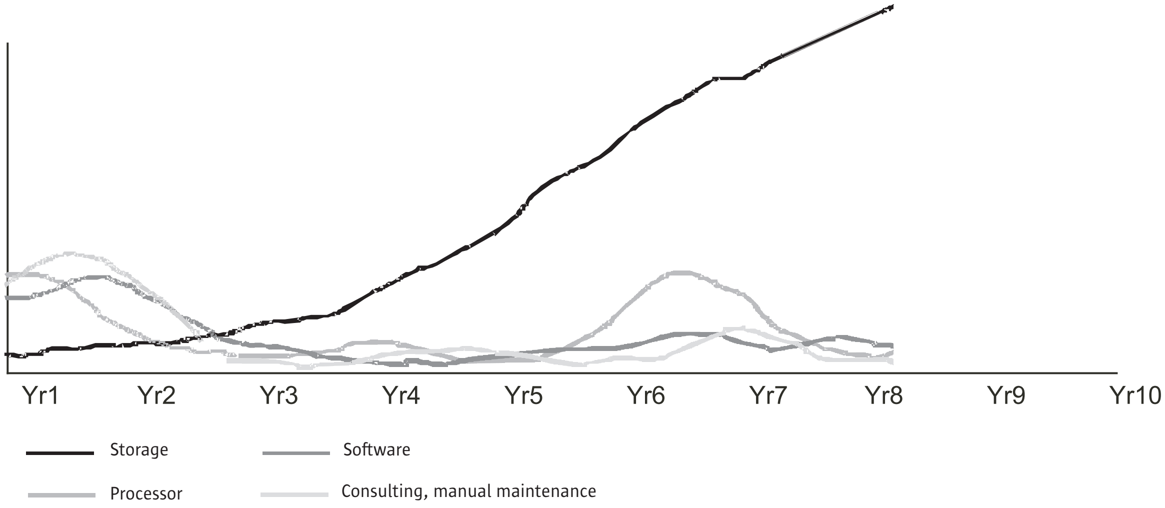
One of the arguments used against the implementation of nearline storage is that of the relative cost of storage. Disk storage vendors love to point out that with a data warehouse that storage costs are only a small part of a much larger cost. Disk storage vendors love to use the chart shown in Figure 23.



Costs over a two year time horizon
**Figure 23**

Figure 23 shows that storage costs are indeed only part of other costs for the data warehouse. Other major costs include processor costs, software costs, and consulting costs.

But there is something misleading about the graph shown in Figure 23. What is misleading is that the graphic is only for two years – the first years of life of the data warehouse. Were the graphic extended over a longer period of time, an entirely different story would be told. Figure 24 tells a different story.

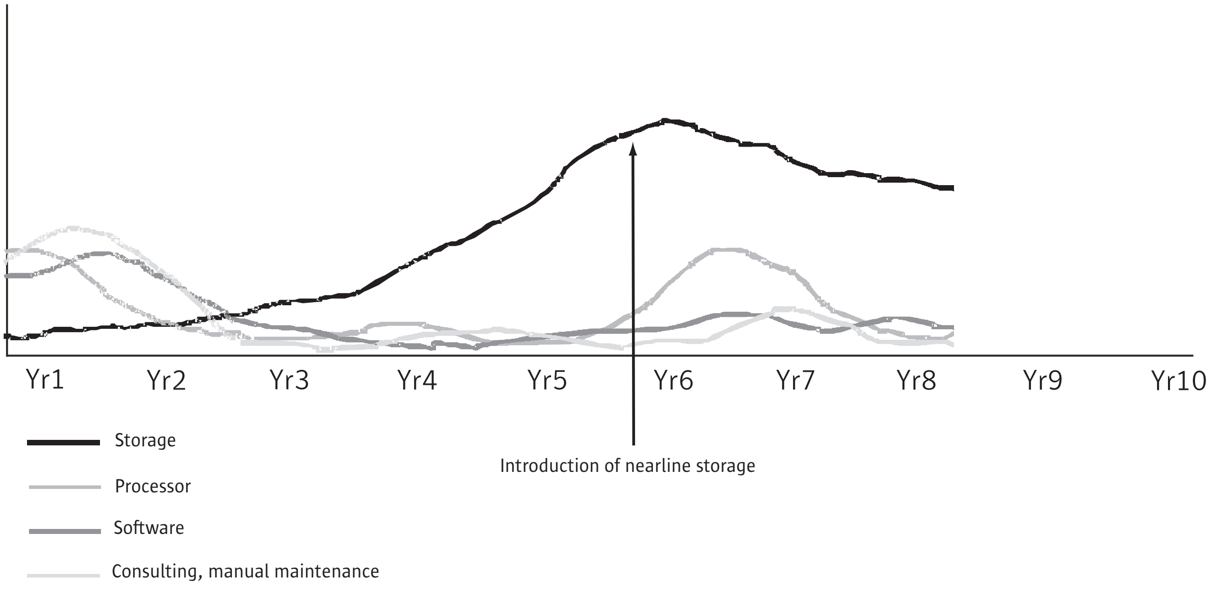


Total data warehouse costs over a ten year horizon
**Figure 24**

Figure 24 shows that storage costs over a lengthy period of time (in this case – ten year's time) make up the major costs of a data warehouse. Over time the other costs of a data warehouse subside. Occasionally there are upgrades, and occasionally there is new development that needs to be done. But over time the major costs of a data warehouse are maintenance costs. But the cost of storage keeps going up and up.

Furthermore, as storage costs go up, the unit price of storage costs also go up. For these reasons then, when looking at a longer time horizon, a very different picture is painted.

There is yet another picture to be painted and that picture is shown by Figure 25.

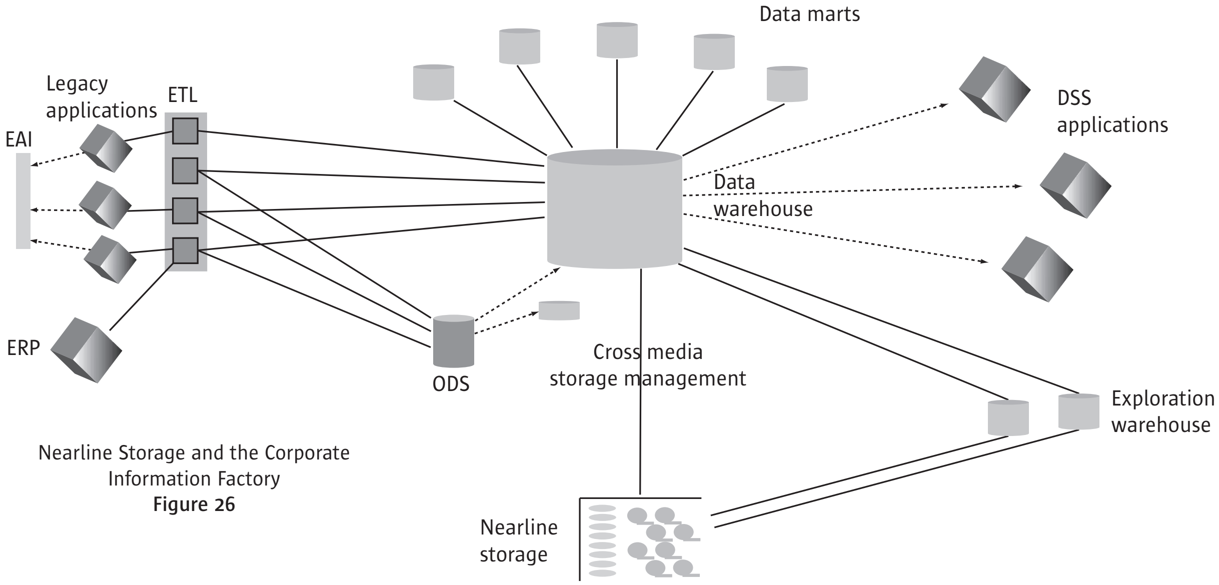


Over time nearline storage mitigates the
major costs of a data warehouse
**Figure 25**

Figure 25 shows that when nearline storage is brought into the picture that the costs of storage start to drop, not accelerate. Once nearline storage is introduced into the picture the total costs of the data warehouse start to become quite reasonable.

## Nearline Storage and the Corporate Information Factory

One final perspective of nearline storage is that nearline storage is part of the larger architecture called the Corporate Information Factory (CIF). Figure 26 shows the fit between nearline storage and the remaining parts of the CIF.



Nearline Storage and the Corporate Information Factory
**Figure 26**

Nearline storage adds a great deal to the CIF. It is because of nearline storage that the CIF can hold an infinite amount of data. It is because of nearline storage that the CIF can hold data over a very long historical perspective of data. It is because of nearline storage that the costs of the CIF environment are reasonable. It is because of nearline storage that exploration processing can be done to a great depth. For these reasons then, nearline storage is an important and necessary part of the CIF.

# In Summary

Data warehouses grow large for a variety of reasons. As data warehouses grow large, a percent of the data found in the data warehouse turns into what can be called dormant data. By moving dormant data to different tiers of storage, including Tier III or nearline storage, the total cost of the data warehouse drops and the performance of the data warehouse is greatly increased (i.e., the data warehouse becomes much faster.)

In order to determine what data to place in each tier of storage, many shops use the crude approach of loading data by date. In this case all data older than two years is loaded into the data warehouse. Another approach is to use a data warehouse monitor. As a rule, the dbms vendor's monitor is not used because of overhead. Instead a third party monitor is used. A more accurate way to determine and monitor data usage patterns of data within a data warehouse is by using a third party tool. An example of a company that provides tools to monitor the warehouse is Teleran Technologies.

If the dbms monitor shows that there are a significant number of accesses to the nearline sector, then the odds are good that a certain amount of exploration/statistical processing is occurring. If that is the case then a separate facility for exploration/statistical warehouse can be built. The effect of building an exploration/statistical warehouse is to make sure that a minimal number of calls to nearline storage are necessary.

The movement of data to and from disk storage and nearline storage is either done manually by a database administrator or is done by software called cross media storage manager (cmsm.) Because cmsm operates at the block or row level, nearline storage can accommodate large data warehouses where some rows are used and other rows are dormant.

Nearline storage is incorporated into the corporate information factory as a normal and necessary component.

## Sun – Bringing Real Value to Information Lifecycle Management

Customers need solutions that enable Information Lifecycle Management (ILM) and demonstrate measurable business value. Sun's experience as a systems company enables it to provide the seamless integration of its systems and software to make Information Lifecycle Management a reality.

Sun understands that ILM isn't about a box and because of this only a systems company like Sun can deliver a true ILM solution. With the alignment of Sun and StorageTek, all of the building blocks are now in place to bring the full potential of information lifecycle management into a reality, including:

1) Broad infrastructure leadership. A broad, value-based portfolio of infrastructure offerings including storage, servers, software and services, provides the right solution at the right cost to meet customer's business requirements.

2) Security and compliance platforms. Sun solutions help you comply more efficiently with industry regulations, control access to information assets and manage operational risk with systemically secure and identity-centric software and systems infrastructure.

3) Systems and application integration. No single vendor has all the pieces needed for a customized ILM strategy. Sun's open systems approach assembles best-in-class components and services to implement the best possible solution for your organization.

Sun provides a broad portfolio of software solutions to address each aspect of the ILM process. Sun is dedicated to providing open, standards-based software solutions supporting heterogeneous environments. Sun's strategy is to provide an integrated, application- and information-aware portfolio of solutions designed to fully automate the end-to-end ILM process, thus optimizing the capabilities inherent in both Sun's storage hardware and existing heterogeneous systems that help customers reduce risk in their environments.

Sun's solutions leverage a deep understanding of the interplay between data and applications used to manage and move data in networked, high-performance computing environments. Leading innovations from Sun have included Network File System (NFS) for managing data in network computing environments, Sun StorEdge™ SAM-FS and QFS software, a scalable, high-capacity file system coupled with policy-based archiving services, and virtualization devices allowing the flexibility for customers to easily allocate and dedicate resources to specific applications.

## Sun's Security and Identity Management Tools

Security is absolutely critical to ILM. As recent newspaper headlines have illustrated, when companies lose or mishandle information such as customer credit card numbers or other personal data, it is not only embarrassing–it can cause irreparable harm or damage to the people whose information was stolen as well as to the company itself. Not only is knowing who has access to what data and the access history important, it is extremely proactive measures are in place as well, when it comes to protecting your data.

Three requirements for a secure infrastructure are authentication, access, and containment. A person's identity must be authenticated and a system put in place for verification of identity, there must be an efficient, automated way to provide access and a unified view of your organization's information for those who need it, such as customers, operations, management, HR, or former employees, and information must be controlled and contained appropriately, using preemptive and defensive techniques, to ensure it is kept out of the wrong hands.

Sun can provide the critical security and identity management tools needed to develop an infrastructure for a secure business intelligence / data warehouse solution. Sun's identity management products provide a unified portfolio for using, sharing, and managing identity information. Solaris™ 10 is open source, unified, and secure – meeting the needs identified as critical to a comprehensive ILM strategy.

## Get Started Today

Sun provides the comprehensive ILM solutions needed to meet today's strategic business needs. Sun offers an Information Lifecycle Management solution that enables businesses, governments, and service providers to accelerate the deployment of ILM solutions to reduce risk and manage business intelligence/data warehouse complexity in the enterprise. Only a systems company like Sun can deliver true ILM for data warehouses.

Contact your local Sun Sales or Partner Representative for more information on how to get started.

- Sun Data Warehouse Health Check, a review to identify dormant data and performance deficiencies in ETL and user query performance against your data warehouse and overall infrastructure.

- Sun Business Intelligence/Data Warehouse (BIDW) Workshop, a one to two day workshop to gather information about your existing BIDW environment and your implementation options followed by a go forward recommendation.

- Sun Data Warehouse ILM Proof-of-Concept (POC), an evaluation to assist in building and testing prototype solutions before deployment.

**Sun Microsystems, Inc.** 4150 Network Circle, Santa Clara, CA 95054 USA **Phone** 1-650-960-1300 or 1-800-555-9SUN **Web** sun.com