COMP440 – Mushkatblat Michael Phillips

CSUN SP2017 00984-1886

**Kimball vs. Inmon**

**Two Opposing Design Philosophies of Data Warehouse Architectures**

**Abstract**

Since the development and evolution of data warehouses through the past decades, two major philosophies have emerged as major design models. These two design approaches are the Inmon top-down architecture vs. the Kimball bottom-up architecture. Both structures have advantages and disadvantages which should be considered during design planning. These are not necessarily overall characteristic pros and cons, but advantages in the context of each particular database design.

**Executive Summary**

Bottom Line Up Front – There is no generalized conclusion that one database structure is better than the other. The two methods described here differ in architecture by how the data storage is structured and modeled. They are simply opposing concepts which both are advantageous in certain conditions. The top-down approach may be better suited for businesses that require historical data that aid in corporate decisions at the enterprise level. If however, data storage is primarily used by independent teams or departments, then the bottom-up model would be more appropriate. The choice falls on how a business would prefer to deal with data that support their style of analysis and reporting.

**Introduction**

Two ideological branches have emerged with regards to large data storage. Companies can store a large repository of insightful corporate data that can be mainly used for strategic business decisions, or they can store their data in department level data marts which are primarily used for team level operations. There is no right or wrong decision – with the top-down approach, enterprise level data storage can still be queried down to the team level; with the bottom-up approach, department level data marts can ultimately be aggregated to the corporate level. The differences in data warehousing lie in business requirements, or which areas demand the most optimization. The same data is stored either way, but packaged differently based on needs.

Data warehouses are central repositories for storing enormous amounts of consolidated data. Big data is a storage trend that has evolved through the past couple of decades as businesses retain client information for aggregated statistics, and keep data to analyze internal trends. Client information can be as simple as contact information from past transactions, or may include descriptive identifiable information. With respect to the ever-increasing dimensions of the internet, online businesses should be focused on an optimized understanding of their customers. Online retailers, for instance, should keep customer data that reflect geographical location, demographics, transaction and referral data, clickstream data (web pages recently visited or linked to), and social media cookies. While these specifics of ordinary transactions may not seem that useful on their own, they can be grouped with thousands of other transactions where trends emerge. This type of analysis and modeling is termed as data mining. Data mining is invaluable to businesses that are looking to grow through maximizing income techniques.

**Data Warehousing Practices**

Warehouses are typically designed to suit the specific needs of an organization. A business may be mostly concerned with data pulls at the enterprise level which may not suit the needs of those organizations that require reporting at the department level. Enterprise-wide data (archived or current) provide helpful insight to an organization’s business intelligence. Warehouses are therefore designed with the concept of streamlined data retrieval.

Bill Inmon, author of Building the Data Warehouse defined a data warehouse as a “subject-oriented, integrated, time-variant, and nonvolatile collection of data in support of management’s decision-making process” (Ricardo 18). These four characteristics are detailed in table 1. The result is a massive database with a complete organizational view from all business operations. With this architecture, team level data marts can be built after the complete database has been developed.

|  |  |
| --- | --- |
| **Subject-oriented** | The data items that describe a subject area (e.g., an entity) are linked together. Examples of subjects might be customers, products, and sales transactions. |
| **Non-volatile** | Data warehouse data is static – not updated or deleted once it is stored. |
| **Integrated** | Data that may be from disparate sources is made coherent and consistent. |
| **Time-variant** | All data stored in the warehouse is identified as belonging to a specific time period. |

Table 1. Data warehouse characteristics

Source: Ricardo, *Databases Illuminated*, Jones & Bartlett Learning, 2017, p. 479

With the Inmon approach, all departments feed their data to a centralized area which is then treated as application independent. One large database is retained as a warehouse that spans across all databases from all departments. Each record can be of any subject from any team. These data are typically denormalized to query corporate level trends from aggregated historical business information. Since warehoused data is kept in one centralized location, they can be obtained through standard queries with common keys. Inmon describes a centralized uniform warehouse with integrated historical data “a single version of the truth” (2).

Ralph Kimball, author of The Data Warehouse Toolkit, developed a different data storage model by which departments keep their own isolated data mart repositories which can then be combined into a complete storage area. This union of data marts describes the bottom-up dimensional model and is connected through bus architecture. Data marts typically arise from segregated departments with independent applications. Therefore, the data from each department are project-oriented facts and are not typically shared between decentralized groups. These data are not necessarily normalized as they are mainly reserved for specific tailored operations. Instead they are organized using a star schema, where multiple dimensional attribute tables are linked to fact tables. Officially they are “a simple form of a data warehouse that is focused on a single subject (or functional area), such as Sales, Finance, or Marketing. Data marts are often built and controlled by a single department within an organization.” (Oracle).

This concept arose from the emergence of business optimization. As corporate leaders demanded efficiency from each subdivision of their business, it became necessary to keep track of team goals and KPI’s (key performance indicators, which describe how well the business is operating from a specific subject area). It is natural for each unit to focus on efficiency in their particular area, and therefore keep application-specific data in their own databases. Retrieving their local data specific to their own day-to-day operations is much faster than requesting data be pulled from an already aggregated massive data warehouse. There is a tradeoff, however, that many independent databases could create unnecessarily higher operational and maintenance costs. Redundant records can be inherently stored amongst multiple data marts. Additionally, star data records would need to be restructured to 3NF format when aggregated into a centralized database.

It has been stated that both a massive relational database can be queried down to application-specific data marts, and dimensional data marts can be aggregated into a union of data marts. As data marts are tailored to the operations of each division, they typically follow the star structure for a dimensional view. This is in contrast to keeping a massive normalized database, which is usually structured for a relational view. Since these methods are defined by different data structures, it is almost certain that a transformation in data structures will be encountered when drilling up or drilling down. The typical storage and retrieval of data is summarized in table 2. “Daily”, “Monthly”, “Quarterly” are not definitive terms; they are simply illustrating the relative frequency or rarity of uploading or downloading information. The main summarized differences are that Inmon supports immediately storing the data as 3NF until it is needed by the user, while Kimball’s view is that many more attributes can be linked to an entity which complicates storage in a single normalized database. Kimball states that even sales order data which can easily fit in star schema would “turn into a complex spider web diagram as a 3NF model” (8).

|  |  |  |
| --- | --- | --- |
|  | Inmon  Top-Down Enterprise Level Storage | Kimball  Bottom-Up Department Level Storage |
| Daily | At the end of business day, transaction data from all departments can be archived to a single relational database. Data would typically be immediately 3NF and uniform. | At the end of business day, each department stores facts and dimensions in their own data mart. Records may not be uniform between departments. |
| Monthly | Departments can query a data pull to a data mart that will be used during monthly reporting to show team goals. A data mart sourced from a centralized warehouse will probably require restructuring. | Data marts can be accessed at the enterprise level through bus structures. They upload their star tables to the centralized data warehouse which requires transformation to normalized, standardized records. |
| Quarterly | Leadership can query across all departments during quarterly reviews to show business trends. Data would typically not be restructured. | Leadership source enterprise info in the same manner, once the warehouse is current with the normalized records. |

Table 2. Typical business-related data storage and retrieval.

**Compromise**

While many businesses find their optimization with one of these two methodologies, some may still view the inefficiencies as a drawback. This extends towards the idea of customization, whereby a combination of both approaches can be created and utilized. With this premise, businesses can choose a primarily top-down architecture, while still using local star schema data marts when applicable (and vice versa). Inmon even suggested that there is a hybrid model that makes sense, shown in figure 1. It contains a “single version of the truth” warehouse surrounded by star schema data marts which are sourced from the 3NF data. This optimized compromise utilizes the best from both architectures and satisfies both ideological branches.

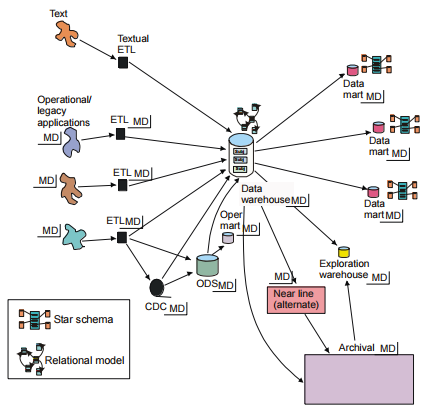


Figure 1. Inmon’s Combined Data Warehouse

Source: Inmon, *A Tale Of Two Architectures*, http://www.inmoncif.com/products/ A%20TALE%20OF%20TWO%20ARCHITECTURES.pdf, p.12

**Conclusion**

In order to create data reports that are meaningful to a business, data must be stored in a logical way that suits the best needs of the application. If data pulls are required that will be used for corporate or enterprise decisions, the Inmon top-down model for data storage is best suited. If however, data storage is primarily used by independent teams or departments, then the Kimball bottom-up model would be more appropriate. A hybrid model is also recommended for new database warehouses. In *Data Warehousing* Concepts, Oracle summarizes that “Data warehouses are designed to help you analyze data” (1).

It has been demonstrated that the Kimball and Inmon approaches to data warehousing are both successful at storing and delivering data. Both models have advantages and disadvantages which need to be considered by the warehouse designers when creating new methods for big data in different circumstances.

On a personal level, I have had the chance to use methods that are comparable to both techniques, although not in the traditional SQL database manner. In private industry, I have had to store client data in one massive storage area. This information was still able to be subdivided into data marts, but was mainly used in whole for top-level business decisions. In the public sector, I currently work in a federated organization with de-centrally segregated divisions working semi-autonomously. Analogous to a data mart structure, there is never information sharing between independent working units, but it is commonly aggregated with other units for top-level decisions. Both methods arose out of tailored business needs and work well in their own circumstances. My industry experience supports the conclusion that both data warehousing methods work suitably dependant on their applications.

References

"Data Mart Concepts". Oracle. 2007. http://docs.oracle.com/html/E10312\_01/dm\_concepts.htm

“Data Warehousing Concepts”. Oracle. 2005. https://web.stanford.edu/dept/itss/docs/oracle/ 10gR2/server.102/b14223/concept.htm

Inmon, Bill. *A Tale of Two Architectures*. http://www.inmoncif.com/products/ A%20TALE%20OF%20TWO%20ARCHITECTURES.pdf

Kimball, Ralph. *The Data Warehouse Toolkit*. Third Edition. Wiley, Indianapolis, IN, 2013. pp.8

Ricardo, Catherine M. *Databases Illuminated*. Third Edition. Burlington, MA. Jones & Bartlett Learning, 2017. pp. 18,479.