Praise for "Super Charge Your Data Warehouse"
and
The Data Vault Model and Methodology

Most Important Design Technique

"Data Vault represents the most important and most basic design technique for integrity of data and audit-ability of data in the data warehouse environment. Built on a DW 2.0 foundation, Data Vault advances the state of the art for data base design for those organizations where auditability of data is an issue.

In addition data vault allows the organization to manage data whose semantics and structure change over time. Classical data base design approaches have required a redefinition and a reorganization of data every time there is a need for a structural change in data.

But with data vault you can change your data structures gracefully over time, thus saving data base administration time and machine resources required for data unload/reload …"

Bill Inmon, Father Of Data Warehousing

Truly Groundbreaking Innovation

"Dan has devised one of the only truly groundbreaking innovations in information architecture over the past twenty years

The Data Vault should be considered as a potential standard for RDBMS-based analytic data management by organizations looking to achieve a high degree of flexibility, performance and openness …"

Doug Laney, Deloitte Analytics Institute

"This enables organizations to take control of their data warehousing destiny, supporting better and more relevant data warehouses in less time than before."

Howard Dresner, Dresner Advisory Services

I Keep A Copy On My iPad

"This book captures a practical body of knowledge for data warehouse development which both agile and traditional practitioners will benefit from. I keep a copy on my iPad so that I have ready access to it when working with clients. 'Nuff said…."

Scott Ambler, Author Of Agile Modeling,
Agile Database Technologies and Several Other Books

"The Data Vault is a foundationally strong and an exceptionally scalable architecture …"

Stephen Brobst, CTO, Teradata

A strong and enduring foundation

"Data Vault Modeling is a strong basis for the majority of the data warehouse implementations. The conceptual simplicity of the data structure makes it easy to populate, extract, extend and explain…"

Mark Zwijsen, The Netherlands

"Tearing down and building up … By tearing down your data stream into hubs, links and satellites you gain a lot of flexibility:

- Flexibility to recreate you source data to a certain point in time. (Great for compliance)

- Flexibility to easily adapt to ever changing reporting requirements.

- Flexibility to automate the data warehousing process and thus save a lot of time and money.

Great reasons to adopt Data Vault Modeling and Methodology! …"

Marco Schreuder, Certified Data Vault Modeler

Highly Scalable Modeling Method

"Relational and dimensional-based integration layers have served the business intelligence discipline well…until now. The future is the Data Vault, an innovative, hybrid approach that draws upon the strengths of relational, dimensional, and highly-normalized modeling architectures resulting in a fresh, process-based, rule-guided, and highly scalable modeling method. There is no going back now …"

Randy Benzel

It Is A Must Read

"Data Vault modeling combines the best of 3NF and Dimensional modeling to allow building fast, flexible, reliable and modular Data Warehouses.

This book answers the what, how & why questions of Data Warehousing.

It is a must read for everyone building or planning to build a Data Warehouse. …"

Gabor Gollnhofer

Flexibility and Simplicity

"I've been dealing with issues related to data integration and constructing physical models of databases many years and I am absolutely convinced that if the architecture and data model is designed correctly, then most problems can be avoided.

Familiarity with the methodology of data Vault allowed me to see old problems in new light.

I found two reasons to adopt Data Vault Methodology

First – flexibility and simplicity modelling data structure; Second – technical issues such as: building more optimal storage of the date, parallel data overloading and partitioning."

Dmitry Pushkashu

Chapter 2
ARCHITECTURAL DEFINITIONS

The Data Vault approach (project/methodology) has common architectural components defined. The components are referred to throughout this book. The purpose of this book is to define the Data Vault data model structures. Context for those structures is a necessary foundational component of understanding the Data Vault. The common architectural components utilized in the Data Vault approach are defined in Figure 2-1:

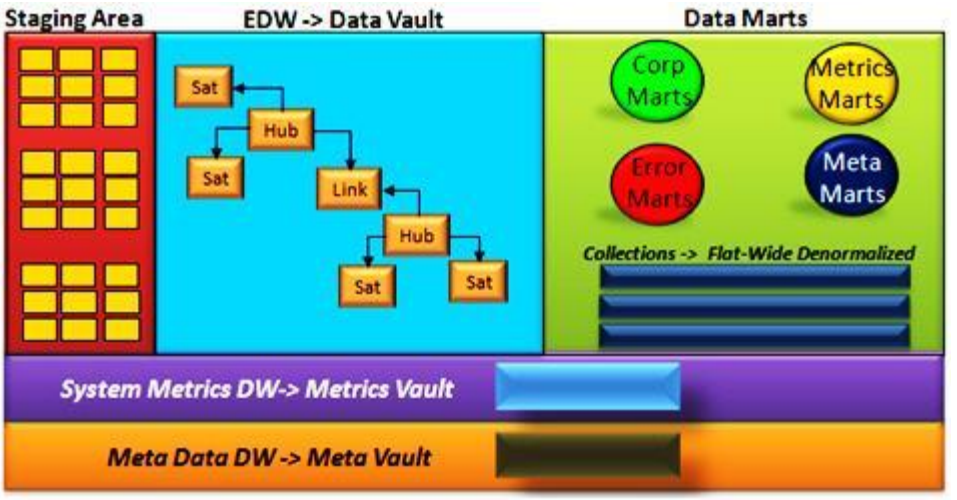

Figure 2-1: Enterprise BI Architectural Components

The Data Vault methodology includes each of these components. The architectural components discussed in this book (in detail) include the Staging area and the Data Vault. This section briefly introduces the other sections as part of the architecture for you to consider.

2.1 Staging Area

The staging area consists of tables in the database to house incoming data 1:1 with the source system (with some additional system driven elements). The staging area is refreshed (purged) prior to each batch load cycle, in other words, they should not ever house history of loads. This is often called a transient staging area. Staging tables house no referential integrity and no foreign keys. They house a sequence number which is reset and cycled for each table with each batch cycle. They house a load date stamp and a record source for each table. These components are described in Chapter 3.0 Common Data Elements.

These tables do not carry any foreign keys, or original primary key definitions. Exceptions: Loading a de-normalized COBOL based file, and executing normalization (splitting into multiple tables), the staging tables will carry parent ID references. Loading a denormalized XML based file and executing normalization, the staging tables will carry parent ID references.

The staging area may be partitioned in any manner desired. The format is owned and maintained by the data warehousing team. The staging area tables may also contain any indexes needed (post-load) in order to provide the data warehouse/Data Vault loads with the proper performance downstream. Staging area data should be backed up at regular intervals (if the data arrives in real-time), otherwise it will be backed up at scheduled intervals.

The future need for a staging area is in question. In fact, within the operational Data Vault and 100% real-time feeds there appear to be no real needs to have a staging area. There are already a few Operational Data Vaults built using the principles of by-passing the staging area, and loading data directly (from the real-time feeds/web-services) to the Data Vault. The only reasons for staging areas to continue to exist (as of 2010) include the following:

- Data Synchronization – with other static lookup data
- Hot-Data Backup – continuous backup in case the queuing engine dies (the transactional feed engine)
- Batch data Delivery – Reformatting and consolidation
- File Format adjustments / alignment

2.2 EDW – Data Vault

The EDW (enterprise data warehouse), or core historical data repository, consists of the Data Vault modeled tables. The EDW holds data over time at a granular level (raw data sets). The Data Vault is comprised of Hubs, Links, and Satellites (defined in section 1.6 and further defined throughout this book). The Enterprise Data Warehousing Layer is comprised of a Data Vault Model where all raw granular history is stored. Unlike many existing data warehouses today, referential integrity is complete across the model and is enforced at all times. The Data Vault model is a highly normalized architecture. Some Satellites in the Data Vault may be denormalized to a degree under specific circumstances.

The Data Vault model follows all definitions of the Data Warehouse (as defined by Bill Inmon) except one: the Data Vault is functionally based, not subject oriented – meaning that the business keys are horizontal in nature and provide visibility *across* lines of business.

The Data Vault modeling architecture has been likened to 3½ normal form. The business keys in the Hub appear to be 6th normal form, while the load date and record source are 3rd normal

form. The Data Vault model should represent the lowest possible grain. The Hubs and Links in the Data Vault model provide the back-bone structure to which context (the Satellites) are applied.

2.3 Metrics Vault

A component for capturing technical metrics about the: load process, loading time-lines, completion rates, amount of data moved, growth of tables, files, and indexes. This Data Vault captures the technical metadata for the processes and the database. By capturing growth rate actuals along with run-times, insert numbers, update numbers, and row counts – projections of future storage requirements can be created and managed. This allows the business to monitor their needs, and budget 6 months to 1 year in advance for future hardware.

The Metrics Vault can also be crafted to include information about CPU utilization, RAM access; I/O throughput and I/O wait times. The additional information in the Metrics Vault begins to provide a consistent and concise view of the utilization of the system in *conjunction with* the growth of the data sets and the *hot spots* on disk. From all of these metrics, a nearly complete technical ***management*** dashboard can be presented to monitor the EDW effort.

2.4 Meta Vault

The Meta Vault contains business metadata (ontologies/taxonomies/definitions) and physical data model attribute names, functions (for translation) and technically implemented business rules that ETL / ELT follows to interpret the data. The Meta Vault allows business to produce, maintain, and deliver metadata across the board from within their EDW/BI solution set. The Meta Vault is in fact one form of Operational Data Warehouse.

The Meta Vault contains metadata for the staging area, EDW Data Vault, Report Collections, Data Marts, and Metrics Vault areas. The metadata is defined through IT, business, and process technologies.

2.5 Report Collections

Report collections are defined as flat-wide denormalized structures, used for high-speed reporting or flat file output access; they may also be used by data mining tools. They are a form of data mart where end-user access is direct. Report collections provide the business users with pre-computed totals at the end of each row. These pre-computed totals allow high speed filtering against patterns of rows that are "out of the normal zone" (in other words, breaking business requirements).

2.6 Data Marts

Data marts are defined as: any point at which generic users directly access the structures and the data for ad-hoc reporting, or drill-down analysis. This may or may not be a Star Schema. It may also include normalized and denormalized tables. Data Marts may be virtualized; for example: in-RAM cubes, and dynamically altered information sets. A form of a data mart is an Excel spreadsheet that communicates directly with the Data Vault through an interactive metadata layer (possibly something like Microsoft SharePoint direct to the Data Vault back-end). Direct communication between the user, the metadata management, and the Data Vault is the beginnings of an Operational Data Warehouse.

For purposes of auditability and accountability the data is separated into two physical layers: corporate marts, and error marts. Corporate marts serve as the standard data marts, where data

that meets soft business rules is contained. Error marts serve as the landing zone for "bad data", that is: data that does not meet soft business rules. The definition of hard and soft business rules is covered in the book: "The Next Business Supermodel, the Business of Data Vault Modeling."

2.7 Business Data Vault

There is a new component in the architecture (not shown in Figure 2-1). The component is called the "Business Data Vault." Business users and IT alike are seeing the benefits of the flexibility, scalability, and adaptability of the Data Vault model. They want the benefits, but with the business data embedded. Downstream of the raw Data Vault, (between the Data Vault and the Data Marts in the Figure 2-1) they are building a new store called the Business Data Vault.

The Business Data Vault (BDV) is a concept, a grouping of specific tables in fashioned using Data Vault modeling concepts, but not necessarily following all the Raw Data Vault modeling rules. A Business Data Vault (also known as EDW+) can be a group of tables inside the raw Data Vault (where the record source has changed), or can be a completely separate data store. Either way, the data that exists in the BDV has been altered, cleansed and changed to meet the rules of the business and is downstream of the raw Data Vault. You may be able to dual-purpose the BDV and apply master data rules as well, thus making the BDV a starting point for a Master Data System.

The Business Data Vault contains all business data, all altered data, aggregated, and cleansed information. IT staff are executing the business transformations once, assigning more metadata (including master data definitions), and then releasing (through simple copy) the data needed in the marts. The Business Data Vault is considered an extra copy of the information; however it is paired with the business metadata and all of the transformations needed to make virtual cubes and high speed delivery possible. The argument received from the business is that the data (post-transformation) is used on the financial reports, and as such, must also be accountable and auditable. Therefore a second copy of the data (post-transformation) is necessary as another system of record.

The technical argument provided is that the IT staff only wishes to do the transformation once, or that they have a standing order to provide "virtual marts"; which in this case translates to RAM based cubes, and views that look like dimensions and facts.

2.8 Operational Data Vault

The nature of the Raw Data Vault (EDW as depicted in Figure 2-1) is changing to include operational data. The need to combine/consolidate operational data with the raw Data Vault is being driven by Master Data Initiatives, and business needs. The business wants more historical data mixed with current transactions at their finger-tips.

In order to meet this demand the Data Warehousing teams are loading operational data (real-time loading) directly in to the Raw Data Vault, thus creating an Operational Data Vault. The entire discussion of Operational Data Vaults is outside the scope of this text, and will be defined elsewhere in articles and discussion forums.

> **WARNING:** AN ODV INHERITS ALL THE ISSUES, PROBLEMS, AND RELIABILITY CONCERNS OF AN OPERATIONAL SYSTEM. ITEMS SUCH AS GOVERNANCE, UP-TIME (6X9'S), 24X7X365 SUPPORT, ALL COME TO BEAR WITH AN OPERATIONAL DATA VAULT. THE DECISION TO BUILD ONE SHOULD NOT BE TAKEN LIGHTLY.

What is an Operational Data Vault? The Operational Data Vault is part data warehouse, and part on-line transactional data store (operational data store). The Operational Data Vault stores all changes to data as inserts (as does a traditional data warehouse), however at the same time it also offers "update/edit" access to the ***operational applications sitting directly on top of the data warehouse***.

In case you are wondering: "Has this ever been done successfully?" The answer is yes, it has – several times already. A company called Cendant Timeshare Resource Group (Cendant TRG) rebuilt their entire operational layer in Java directly on top of the Data Vault, consolidating data warehousing directly with operational applications. There were no separate systems for reporting, no separate systems for "operational data" or OLTP applications, simply the Data Vault and the Java OLTP application. This is one example which has been in use since 2001.

Another example is a drug manufacturing traceability warehouse that was built in 2008 for the US Congress. This Data Vault had operational applications that were driven by drug packaging machines which assigned unique ID's to every drug package from every manufacturer around the world. These machines fed the data over remote web-services connections directly to the Data Vault every 10 minutes, where the data was encrypted, secured, and stored – only to be accessed every time the drug was scanned at different points in the supply chain. At which time the warehouse would provide different web-service access points to retrieve audit trails of all points where the drug was scanned. In this manner, you (the consumer) could log in to a web-site after purchasing a drug, type in its bar-coded number, and check its authenticity. It was called: Drug Track And Trace anti-counterfeit operation.

2.9 Dynamic Data Vault

The Dynamic Data Vault is an operational Data Vault with dynamic adaptation to the structure. In other words, the tables, columns, indexes, and keys are all subject to change – automatically. Of course to achieve this state requires a constant vigilant watch on the metadata, including but not limited to incoming structures. The incoming structures may include XSD, XML, staging tables, or other metadata (including queue based or process metadata) that describe the structure of the incoming data set.

The dynamic nature of the Data Vault means: new attributes may be added to Satellites, new Links and new Hubs may be formed on the fly. ETL /ELT loading code will be adjusted automatically, and BI Query views will also inherit certain changes. At the end of all the automatic model changes, emails of the changes are sent to the IT staff for review in the morning.