Harmonizing Data Through Data Vaults (DV 2.0)

A White paper

November 2018

Presented by:

ANAND RAGHOTHAMA RAO

Contents

1. What are the prerequisites required to understand this White Paper?

In the recent years, the database community has witnessed the emergence of a new Tech stack to manage data followed by advanced modeling techniques which are most likely revolves around data and information management space. Data management frameworks is usually a global repository that stores pre-processed queries on data which resides in multiple, possibly heterogeneous, operational or legacy sources. The information stored here can be easily and efficiently accessed for making effective decisions. Predominantly, the analytics front will help user community to get deep dive into complex multi-dimensional analysis of underlying data. However, there are still a number of problems that need to be solved for making data management effective. Some of the key pain areas witnessed are:

Multiple data stores which are not relevant to business which can eventually leads to compliance issues.
This can be addressed quickly using the technique 'Data Vault' by dividing the information into chunks of info thereby automatically supporting compliance issues. Similarly, there are 'Subject Area' relevant issues which requires special attention, otherwise it produces data chaos. Good example would be, let's say CDK has acquired some xyz organization, and it has its own data mart which needs to be integrated with CDK portfolio. It is the best use case to illustrate data vault concept.

In this paper, we discuss recent developments in advanced modeling techniques to address data anomalies. A number of technical issues for exploratory research within our product portfolio are presented and possible solutions are discussed.

2. Background

In any enterprise it is a legacy or tradition to maintain de-centralized pieces of information in multiple data stores with an aim to ease of maintenance and reduce complexity during designing systems. Eventually over time, when organizations demand for integrated business information on a global model, it is cumbersome to put together pieces of related information across various systems to meet the objective of business goals.

With the existing Enterprise Data warehouse (EDW) or Centralized or Corporate Data Warehouse (CDW) we are getting the metrics we need for our reports, and management seems content with what it is receiving. But what happens when the executive management decides to get rid of getting multiple reports from different areas of the business, all of which are completely un related to each other? Suddenly there is a business need to '**conform**' information from across the organization and turn our humble creation into an Enterprise Data Warehouse (EDW). Something tells me the executive management won't be happy to hear an estimate of months or years to deliver value from such a project…

Best use case would be, build centralized data store /repository to integrate multiple data stores which are scattered across organization. Which provides the flexibility to define a business goal, relating data within or across business domains.

Agile BI would be another use case.

This is where Data Vault comes in handy.

We still have source systems in their various forms and we still need those data marts, typically in the form of a dimensional model, for the business reporting needs. What we might be missing is the middle bit in

between that allows us to continually add new sources to data warehouse without breaking it, while capturing all of our previous data's history.

That's right, Data Vault evolves as your organization evolves! Data Vault is designed to quickly adapt to change within the business and to allow new areas of the business to be easily integrated into the EDW.

It turns what can be a very painful process of trying to fit other areas of the business and their entirely different terminology and data into our data warehouse, into a relatively simple one.

The beauty of Data Vault is it allows us to bring together numerous and unrelated sources and conform them into logical groups (hubs and satellites to get technical) that suit the organization as a whole, removing the tangled spaghetti mess that often arises when combining sources. The advantage of taking this smaller step to logical groups first using Data Vault, before driving all the way to fully conformed dimensions, is adaptability to change.

Data Vault is the real success story behind a truly integrated and agile Enterprise Data Warehouse.

With the above context, harmonization ensures combining datasets collected at different times into a single, consistent data series. It can be re-organizing the data, metadata management, dissemination and so on. So, I admit the fact that, harmonization and data vaults are interdependent each other.

3. Introduction

'Data Vault' (DV) is one of the flexible data modeling techniques built for data management especially when implemented on Enterprise wide platforms. More precisely, it is a hybrid approach encompassing the best of breed between relational (3NF) and dimensional models. The design is flexible, scalable, consistent and adaptable to needs of the enterprise. It removes any need for multiple data storages as it stores information 'as it is' delivered to the data warehouse, thereby automatically supporting compliance/audit issues (Basically we divide the Information into chunks of information regarding a specific business entity or more precisely a business key).

Architected specifically to meet the needs of today's enterprise data management systems that accommodate **Hybrid** platforms.

4. Perspective: What are we trying to solve through Data Vaults (DV 2.0)?

To address the limitations with the currently known Enterprise Data Management options such as:

- Relational model: Complex primary keys (PK's) with cascading snapshot dates
- Dimensional: Difficult to reengineer fact tables for granularity changes

In addition:

- Difficult to get it right the first time
- Not adaptable to rapid business change
- Missing pieces from data harmonization

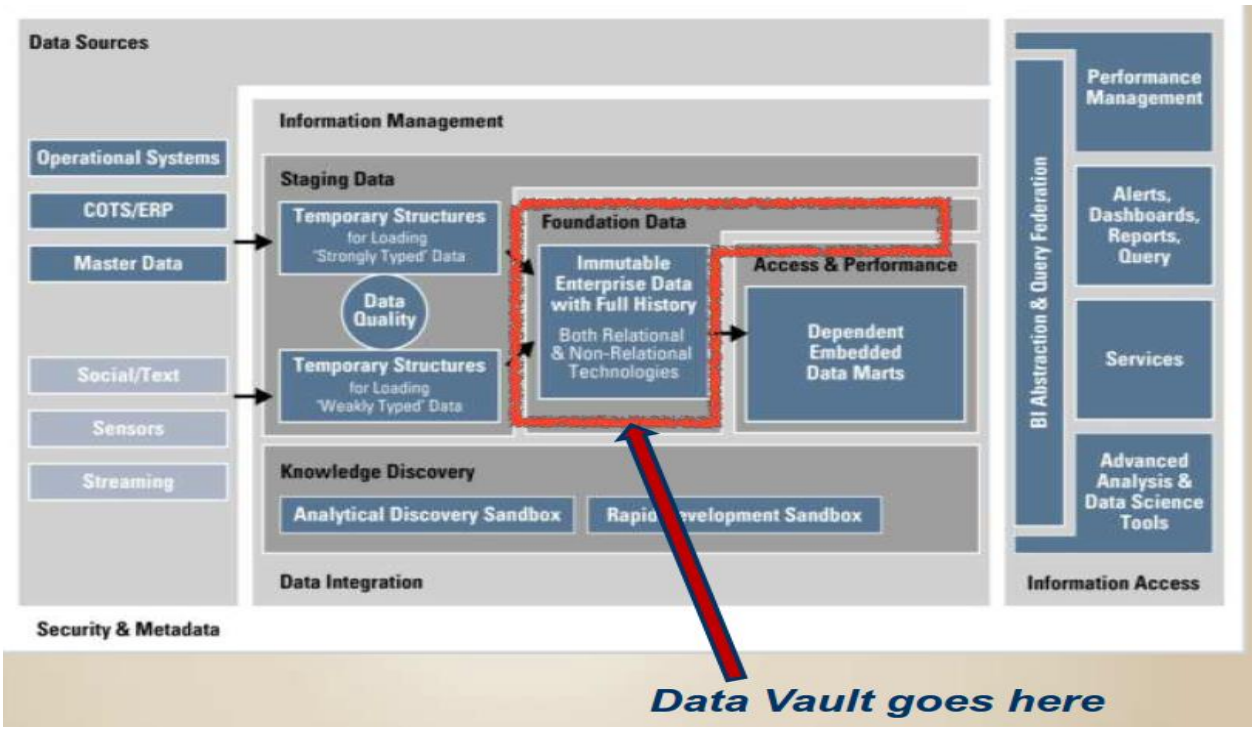
Business perspective:

- Using a DV forces examination of source data processes, and source business processes.
- Businesses believe their existing operational reports are "right", the DV architecture proves this is not always the case.
- Business Users from different units MUST agree on the elements (scope) they need in the Data Vault before parts of it can be built.

Technical perspective:

- Data Vault model is based on **MPP computing, not SMP (symmetric multiprocessing)** computing, and is not necessarily a clustered architecture.
- Data Vault contains all deltas, only houses deletes and updates as status flags on the data itself.
- Data must be made into information BEFORE delivering to the business.
- Stand-alone tables for calendar, geography, and sometimes codes and descriptions are acceptable
- Businesses must define the metadata on a column based level in order to make sense of the Data Vault storage paradigm

5. Where does DV fit into Information Management Architecture?



What is Foundation Layer all about?

- It is basis for long term enterprise scale data warehouse
- Must be atomic level data
  - A historical source of facts
- It is NOT based on any one data source or system
- Single point of integration
- Flexible
- Extensible
- Provides data to the access/reporting layer

6. Data Vault Core Architecture Components

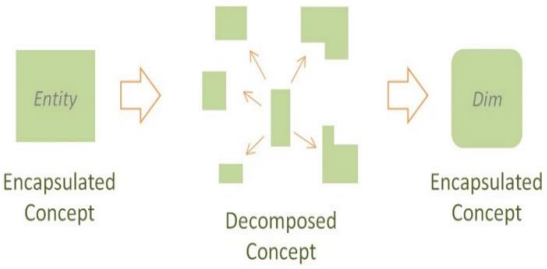
DV works on the principle of **'Unified Decomposition'** technique.

Unifying

If you ever worked with object oriented design, you are probably accustomed the idea of encapsulation. The idea of encapsulation is to bring together methods and data into the same object so that everything that deals with that object is contained within it. One of the advantages of this kind of design is the ability to take an object class from one area and place it in another area knowing that everything it needs to exist (keys and descriptive context) and to perform (behaviors) moved along with it. The object is self-contained.

Breaking into Parts - decomposition

The other part of unified decomposition is the idea of breaking things into component parts. The decomposition is in some ways the opposite of unifying. If we strive to keep things together, why then would we want to break them apart? One major reason in data warehousing is that things change. In fact things change all of the time. If there is one constant it is that things change. But not everything about a concept changes at the same time.

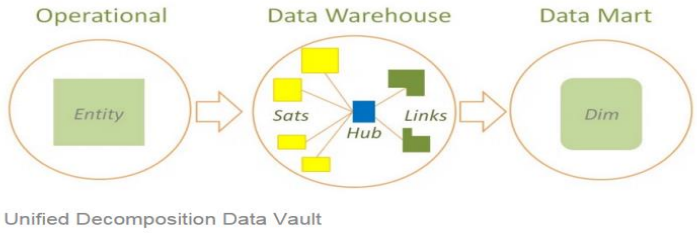


Unified Decomposition

If the concept parts are all kept together (in the same table for example) then that would mean any change to any one component part would have an impact on the whole. If we want to limit the impact of the changes we need to isolate the part that is changing. In data modeling (especially for data warehousing) this theory is being deployed in many different forms. If we are designing a database that needs to integrate data and also needs to maintain history then the benefits of decomposing the core concepts is very compelling. This happens in Dimensional modeling with mini-dimensions and factless facts, it happens in Data Vault with hubs, links and satellites, but it also happens with other approaches such as Anchor

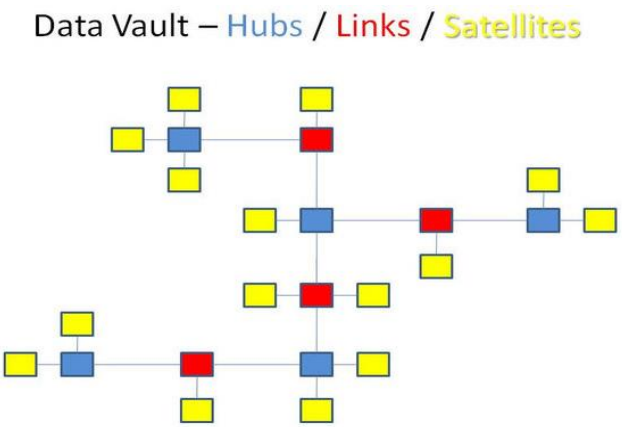
Modeling, 2G and Focal Point. The common theme is data warehousing and the common thread is decomposition.

Putting it all together



Unified Decomposition Data Vault

- Hubs = Unique List of Business Keys
- Links = Unique List of Relationships across keys
- Satellites = Descriptive Data (Slowly and Rapidly Changing attributes)
- Satellites have one and only one parent table
- Satellites cannot be "Parents" to other tables
- Hubs cannot be child tables



a. What makes a Hub Key?

- ✓ A Hub is based on an identifiable business key.
- ✓ An identifiable business key is an attribute that is used in the source systems to locate data.
- ✓ The business key has a very low propensity to change, and usually is not editable on the source systems.
- ✓ The business key has the same semantic meaning, and the same granularity across the company, but not necessarily the same format.

Attributes and Ordering

- All attributes are mandatory.

- Sequence ID 1st, Busn. Key 2nd , Load Date 3rd ,Record Source Last (4th).
- All attributes in the Business Key form a UNIQUE Index.

b. What Makes a Link?

- A Link is based on identifiable business element relationships.
- Otherwise known as a foreign key,
- AKA a business event or transaction between business keys,
- The relationship shouldn't change over time
- It is established as a fact that occurred at a specific point in time and will remain that way forever.
- The link table may also represent a hierarchy.

Attributes

- All attributes are mandatory

c. Modeling Links – 1:1 or 1:M?

Today:

- Relationship is a 1:1 so why model a Link?

Tomorrow:

- The business rule can change to a 1:M
- You discover new data later.

With a Link in the Data Vault:

- No need to change the EDW structure.
- Existing data is fine.
- New data is added.

d. What Makes a Satellite?

- A Satellite is based on an non-identifying business elements.
- The Satellite data changes, sometimes rapidly, sometimes slowly.
- The Satellite is dependent on the Hub or Link key as a parent,
- Satellites are never dependent on more than one parent table.
- The Satellite is never a parent table to any other table (no snow flaking).

Attributes and Ordering

- All attributes are mandatory – EXCEPT END DATE
- Parent ID 1st, Load Date 2nd, Load End Date 3rd,Record Source Last

e. Satellite Entity - Details

- A Satellite has only 1 foreign key; it is dependent on the parent table (Hub or Link)
- A Satellite may or may not have an "Item Numbering" attribute
- A Satellite's Load Date represents the date the EDW saw the data (must be a delta set)
  - This is not Effective Date from the Source!
- A Satellite's Record Source represents the actual source of the row (unit of work)

- To avoid Outer Joins, you must ensure that every satellite has at least 1 entry for every Hub Key

7. How we can correlate Hub, Satellite and Link to CDK product portfolio?

HUB:

For example, invoice number, employee number, customer number, part number and VIN (Vehicle Identification Number). If the business were to lose the key they would lose the reference to the context, or surrounding information.

Another example : The requirement is to capture customer number across the company. Accounting may have a customer number (12345) represented in a numeric style and contracts may have the same customer

number prefixed with an alpha (AC12345). In this case, the representation of the customer number in the Hub would be alphanumeric and set to the maximum length to hold all of the customer numbers from both functional areas of business. The Hub would have two entries: 12345 and AC12345, each would have their own record source – one from accounting and one from contracts. The obvious preference is to perform cleansing and matching on these numbers to integrate them together. However that topic is out of scope for this paper. The Hubs' primary key always migrates outward from the Hub. Once the business is correctly identified through keys (say customer and account) the Link Entities can be constructed.

LINK:

For example, it is not enough to know what a VIN number is for a vehicle, or that there is a driver number 5 out there somewhere. The customer is looking to know what the VIN represents (i.e. a blue Toyota pickup, 4WD, etc.) and that driver number 5 represents the name Jane and then they may want to know that Jane is the driver of this particular VIN.

SATELLITE:

For example, the fact that VIN 1234567 represents a blue Toyota truck today and a red Toyota truck tomorrow. Color may be a Satellite for automobile. Its design relies on the mathematical principles surrounding reduction of data redundancy and rate of change. For instance, if the automobile is a rental, the dates of availability / rented might change daily which is much faster than the rate of change for

color, tires or owner.

8. How to be Agile using DV?

- Model iteratively
  - Use Data Vault data modeling technique
  - Create basic components, then add over time
- Virtualize the Access Layer
  - don't waste time building facts and dimensions up front
  - ETL and testing takes too long
- Users see real reports with real data

9. How does harmonization fit with DV

I would argue the fact that, harmonization and DV are co-related each other. Harmonization ensures combining datasets collected at different times into a single, consistent data series. It can be re-organizing the data, metadata management, dissemination and break up's are as below ..

1. Data :
   - Reorganize data structure
   - Recode variables so codes are consistent
2. Metadata :
   - Systematize documentation
   - Develop new metadata describing comparability
3. Dissemination
   - A system to deliver the data and metadata

With proper marriage between 'Harmonization' and 'DV' we can address the challenges pertaining to Metadata management where metadata is more challenging than data. Assuming, this will go well we can treat Harmonization as a data research activity.

a. After all, what is the need of harmonizing the data?

- It will have great impact in Investment in infrastructure
- It supports pooling of data
- Needless to say, it will increases efficiency/decreases errors
- On top of above all , more feasible to replicate results for data research activities

Although, mentioning the harmonization types (such as., Standardization, Integration, Dissemination) is out of scope here, I would think it will serve as extension for future study.

10. Measuring Data Vaults (DV)

a. Measuring flexibility:

- Goes beyond standard 3NF
  - Hyper normalized
  - Hubs and Links only hold keys and meta data
  - Satellites split by rate of change and/or source
    - Enables Agile data modeling
  - Easy to add to model without having to change existing structures and load routines
    - Relationships (links) can be dropped and created on-demand.
  - No more reloading history because of a missed requirement
- Based on natural business keys
  - Not system surrogate keys
  - Allows for integrating data across functions and source systems more easily
- All data relationships are key driven

b. Measuring extensibility:

- Adding new components to the EDW has NEAR ZERO impact to:
  - Existing Loading Processes
  - Existing Data Model
  - Existing Reporting & BI Functions
  - Existing Source Systems
  - Existing Star Schemas and Data Marts

c. Measuring Productivity:

- Standardized modeling rules
  - Highly repeatable and learnable modeling technique
  - Can standardize load routines
- Delta Driven process
- Re-startable, consistent loading patterns.
  - Can standardize extract routines
- Rapid build of new or revised Data Marts
  - Can be automated
  - Can use a BI-meta layer to virtualize the reporting structures
  - Can put views on the DV structures as well
- Simulate ODS/3NF or Star Schemas

d. Measuring adaptability:

- The Data Vault holds granular historical relationships.
- Holds all history for all time, allowing any source system feeds to be reconstructed on-demand
- Easy generation of Audit Trails for data lineage and compliance.
- Data Mining can discover new relationships between elements
- Patterns of change emerge from the historical pictures and linkages.
- The Data Vault can be accessed by power-users Data Vault Adaptability
- Dynamic Model Adaptation – self healing
- Terabytes to Petabytes of information (Big Data)
- Seamless integration of unstructured data
- Business rule changes (with Ease)

7. Other Benefits of a Data Vault

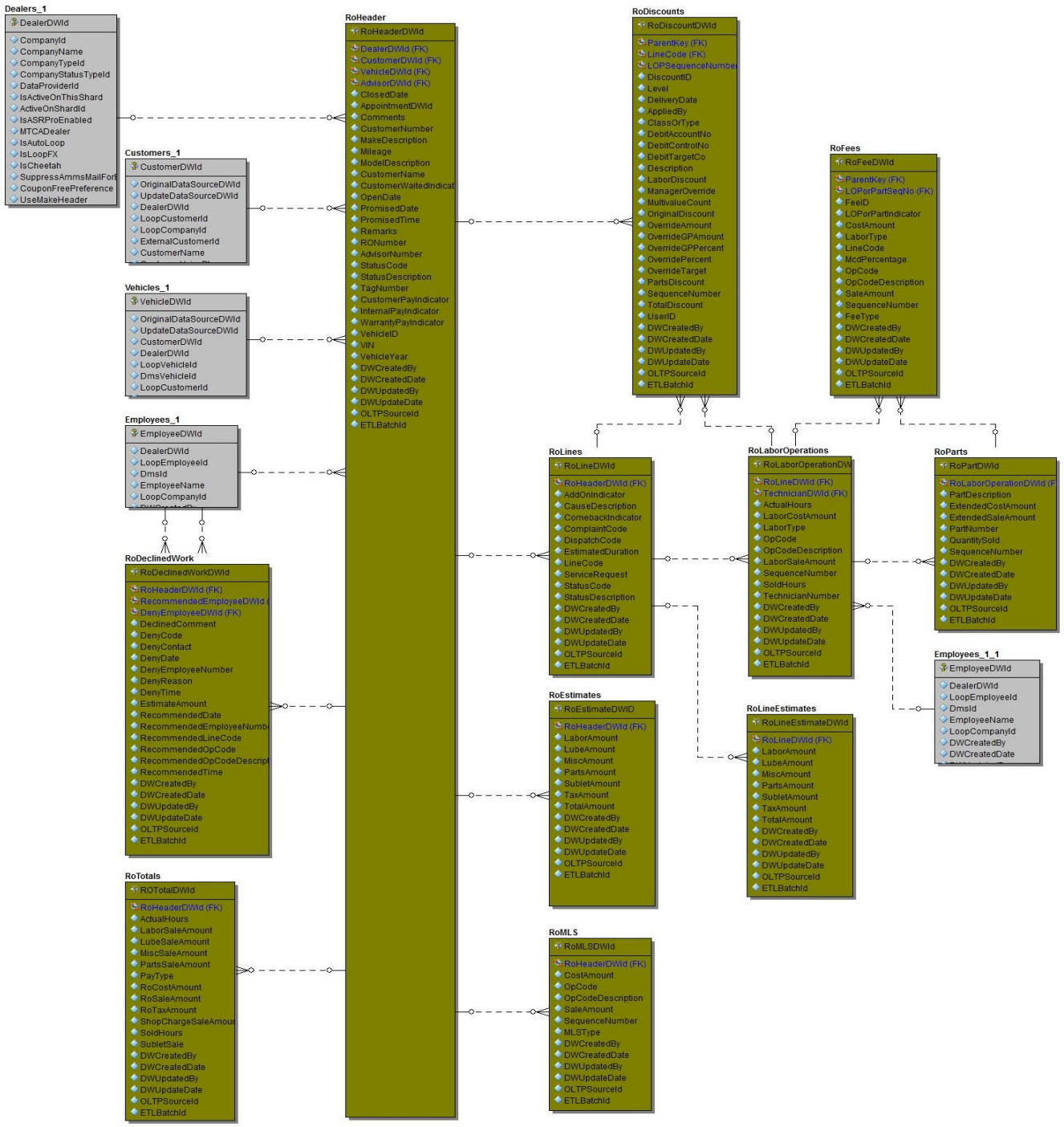
- Modeling it as a DV forces integration of the Business Keys upfront.
  - Good for organizational alignment especially during acquisitions
- An integrated data set with raw data extends it's value beyond BI:
  - Source for data quality projects
  - Source for master data
  - Source for data mining
  - Source for Data as a Service (DaaS) in an SOA (Service Oriented Architecture).
- Upfront Hub integration simplifies the data integration routines required to load data marts.
  - Helps divide the work a bit.
- It is much easier to implement security on these granular pieces.
- Granular, re-startable processes enable pin-point failure correction.
- It is designed and optimized for real-time loading in its core architecture (without any tweaks or mods).

8. Application of DV within CDK – Practical Example

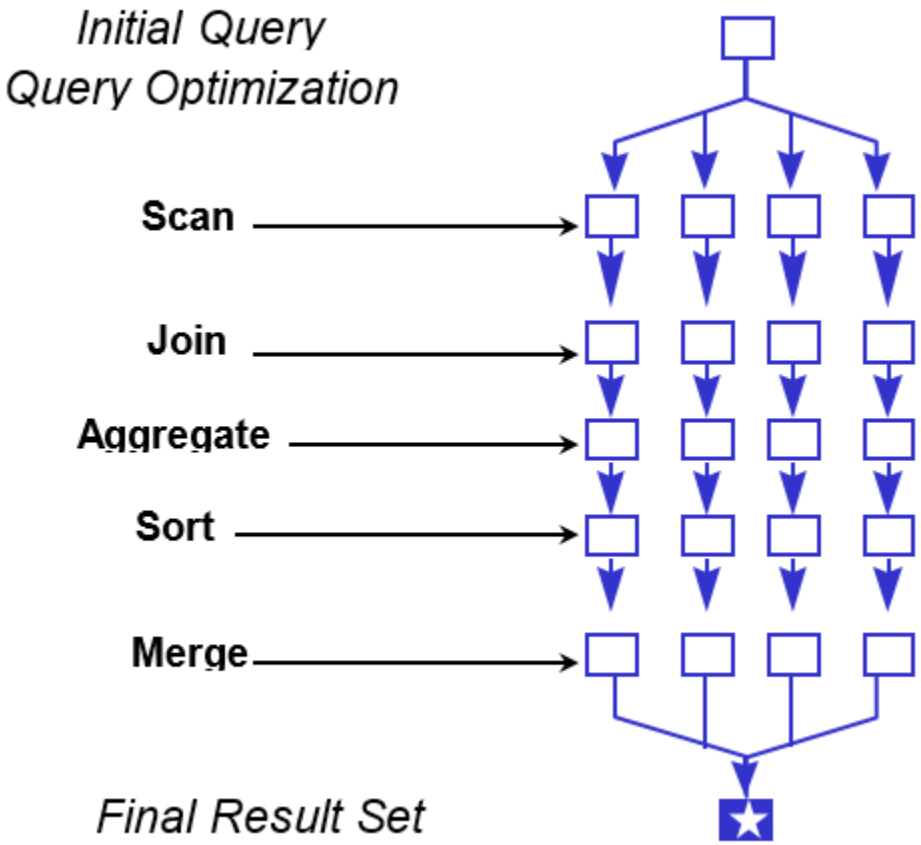
The below model from Repair Orders module of CDK Service product demonstrates the conversion techniques from Normalized model into Data Vaults.

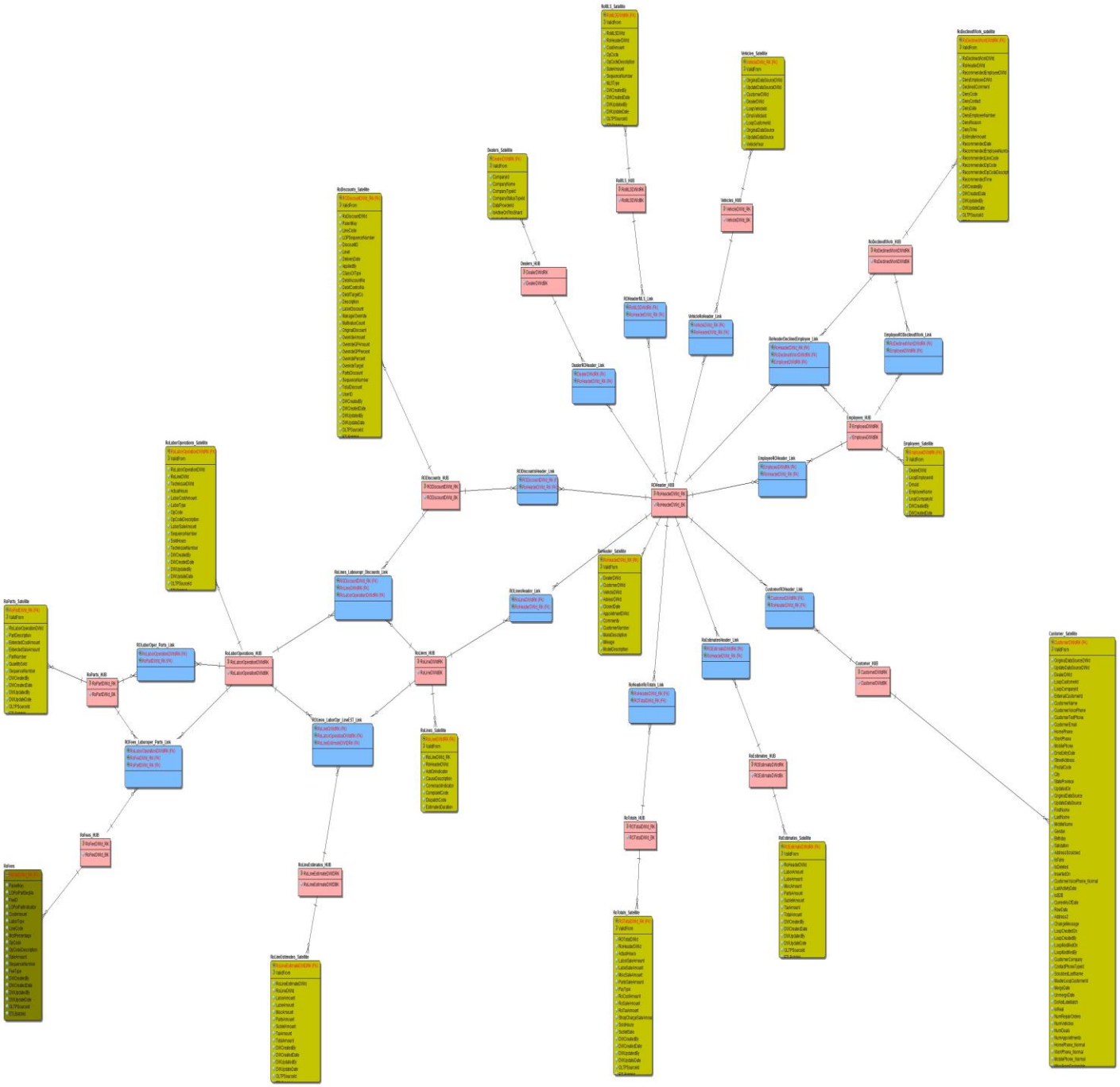
Please refer the before and after sections of implementation. While Hubs make it business driven and allow for semantic integration across systems, Links give us the flexibility to absorb structural and business rule changes without re-engineering (and therefore without reloading any data). Similarly, Satellites ensures the adaptability to record history at any interval you want plus unquestionable auditability and traceability to your source systems.

Before conversion:



After conversion:





Appendices

Appendix A – Organizations using Data Vault

- Independent Purchasing Cooperative (Subway)
- WebMD Health Services
- Blue-Cross Blue Shield
- Microsoft Corporation and many more ….

Appendix B – References

http://www.danlinstedt.com/

https://en.wikipedia.org/wiki/Data_Vault_Modeling

http://www.linkedin.com/groups?gid=44926

www.youtube.com/LearnDataVault