Data Warehouse & Business Intelligence Fundamentals

Todor Kichukov

todor.kichukov@bipartner.biz

https://www.facebook.com/groups/SUDWBI2022/

Faculty of Mathematics and Informatics

Sofia University

2022

Data Warehouse & Business Intelligence Fundamentals

Course Scope

- DW Concept
- DW Architecture
- DW Data Modeling
- Data Integration
- Gathering and Analyzing Requirements
- Business Intelligence
- Deployment, Support and Maintenance

Data Warehouse Architecture

Part II

- Data Vault Evolution and Pillars
- Data Vault 2.0 Architecture
- Data Vault Integration with NoSQL (BigData)
- Data Vault vs 3NF vs Star-Schema
- Data Vault 2.0 Modeling
- Why Data Vault?
- Architecture Implementation Specifics
- Terminology

Dan Linstedt’s Challenges

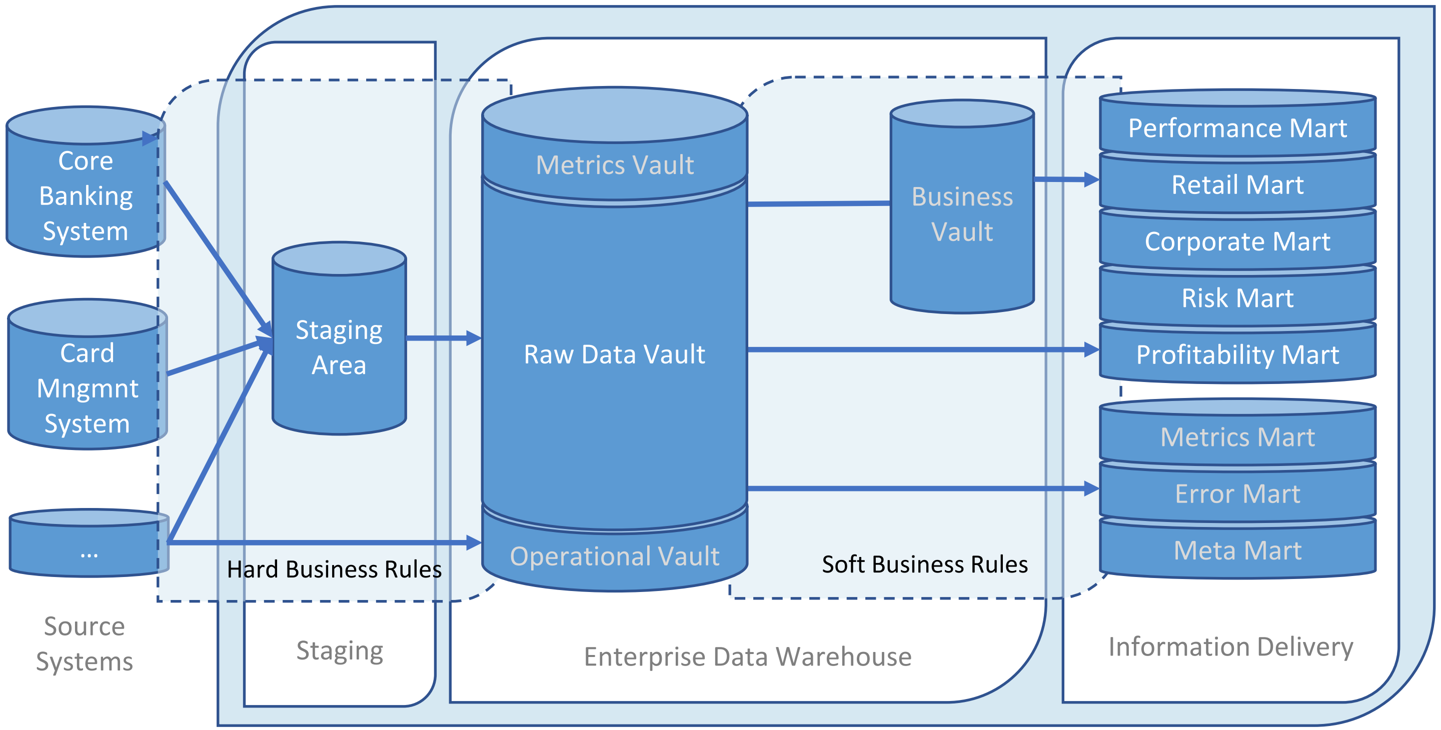
- To consolidate data from/on different data stores and platforms
- To integrate quickly and efficiently disparate data
- Auditability of the Data Warehouses
- Process Repeatability, Cycle Time Reduction
- The ability to build DW in a standardized way, over and over again, regardless of the business requirements

Data Vault Evolution and Pillars

- **2000** Data Vault 1.0;
- **2013** Data Vault 2.0;
- **2019-07-19** Data Vault 2.0.2 Modeling Specification Update

- Data Vault 2.0 Pillars
  - **Data Vault 2.0 Modeling** – built for load performance and scalability, interacts seamlessly with (or live on) NoSQL and Big Data systems.
  - **Data Vault 2.0 Architecture** – including NoSQL systems, real-time feeds, and big data systems for unstructured data handling and big data integration.
  - **Data Vault 2.0 Methodology** – following Scrum and Agile best practices, focuses on 2 to 3-week sprint cycles with adaptations and optimizations for repeatable data warehousing tasks.
  - **Data Vault 2.0 Implementation** – focuses on automation and generation patterns for time savings, error reduction, and rapid productivity of the data warehousing team.

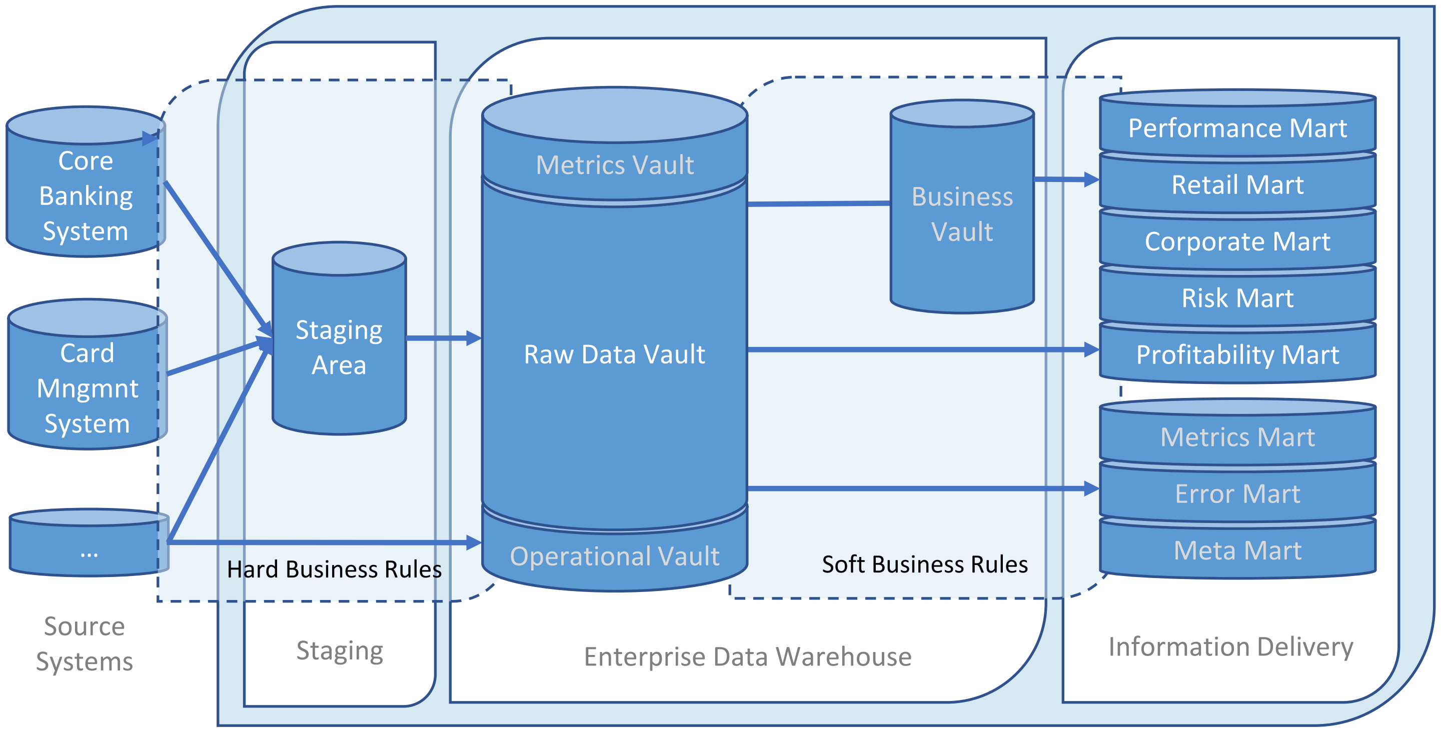
Data Vault 2.0 Architecture



Specifics

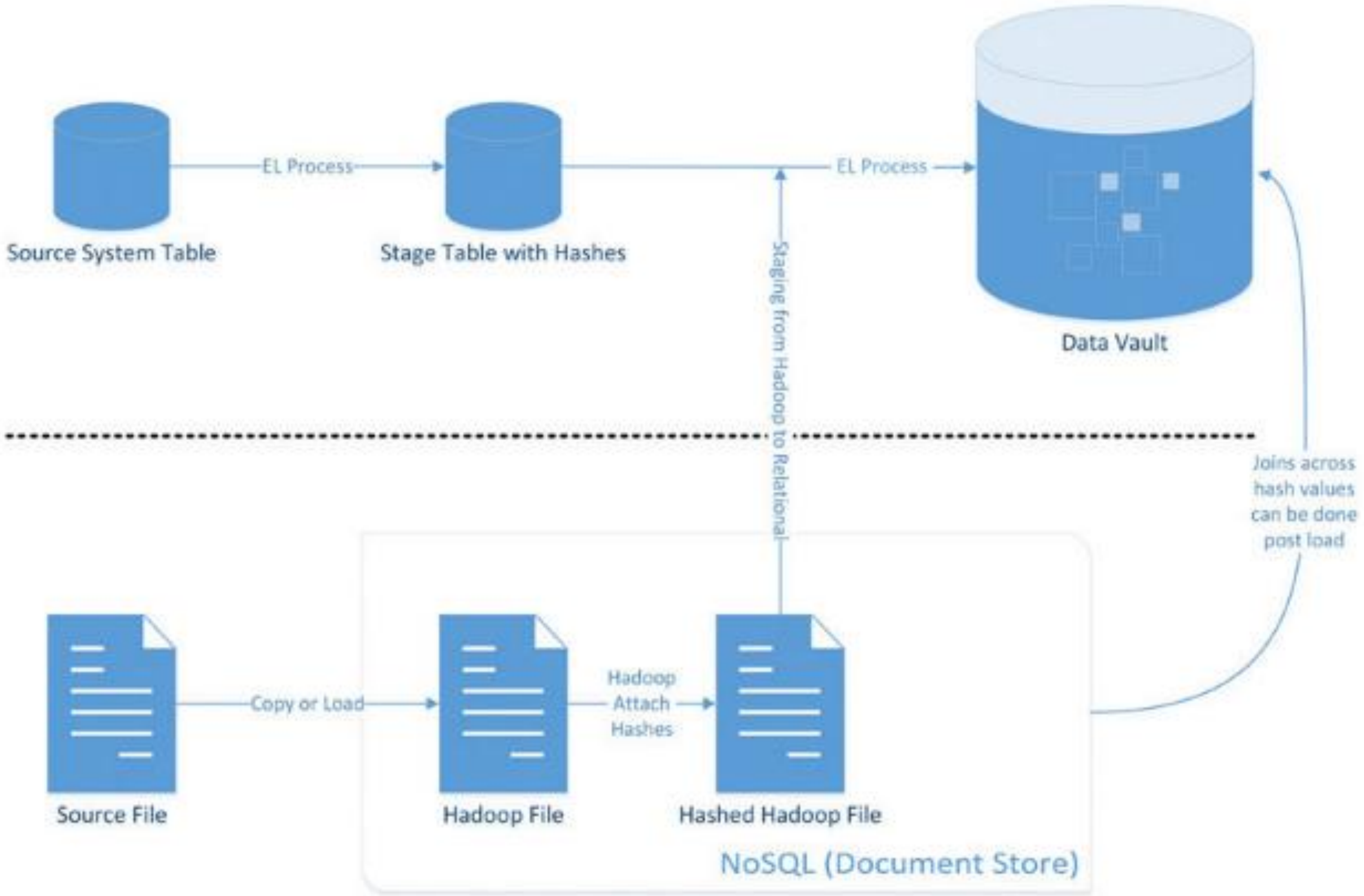
- Business processing moved downstream
- Single Source of Facts (not Single Source of Truth)
- All the data, all the time
- Auditability and Traceability
- Suitable for Operational Data Store (ODS)
- Suitable for sourcing an already built Inmon / Kimball Data Warehouse

Data Vault 2.0 Architecture



- An optional **Business Vault** that is used to store information where common business rules have been applied. Acts as intermediate layer to the DMs. Dependent on Raw Data Vault.
- An optional **Operational Vault** that stores data fed into the EDW directly from operational systems. Read/Write access. Acts as **Operational Data Store (ODS)**.
- An optional **Metrics Vault** that is used to capture and record runtime information, including the run history, process metrics, and technical metrics (CPU loads, RAM usage, disk I/O metrics, network throughput). On top of it, the **Metrics Mart** provides the performance metrics information to the user.
- An optional **Error Mart** for errors in the EDW load.
- An optional **Meta Mart** for EDW metadata.

Data Vault Integration with NoSQL (BigData)



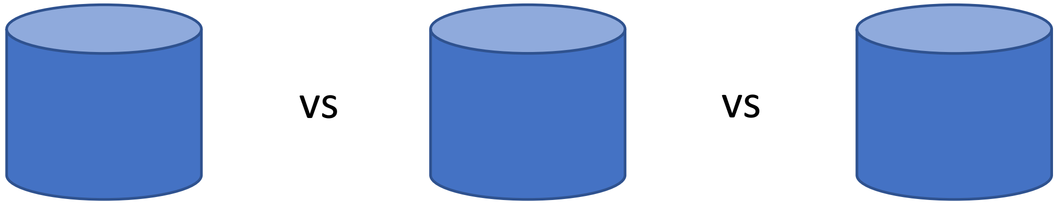
Data Vault uses **hash keys** as they improve the interoperability between different platforms, such as the relational database and NoSQL environments. By using hash keys, it is possible to integrate data on various platforms, structured in the relational database and unstructured data in NoSQL environments such as Hadoop. However a relational database is the best choice if the incoming data is coming from relational sources (if the source data is extracted directly from its operational database).

Data Vault vs 3NF vs Star-Schema

Bill Inmon’s
3NF, Normalized
Data Model

**Dan Linstedt’s
Hybrid
Data Model**

Ralph Kimball’s
Star-schema, Denormalized
Data Model



Source Systems,
Enterprise Data
Warehouse

Data Vault

Enterprise Data
Warehouse, Data Marts

Issues with: cascading change impacts, difficulties in near real time loading, troublesome query access, problematic drill-down analysis, scalability, flexibility

Issues with: synchronization in near real time loading, relationship change, consistent grains, to some extent flexibility

Data Vault 2.0 Components:

- Modeling – Hub, Link, Satellite
- Methodology – Scrum/Agile/CMMI/Six Sigma/etc.
- Architecture – including NoSQL, BigData
- Implementation – pattern based, automation

Data Vault Advantages:

- Flexible, scalable, consistent and adaptable to the needs of the enterprise.
- Quicker data loading compared to 3NF and star-schema
- Very suitable for real-time loading
- Easily adding of new data sources
- Process Repeatability, Cycle Time Reduction

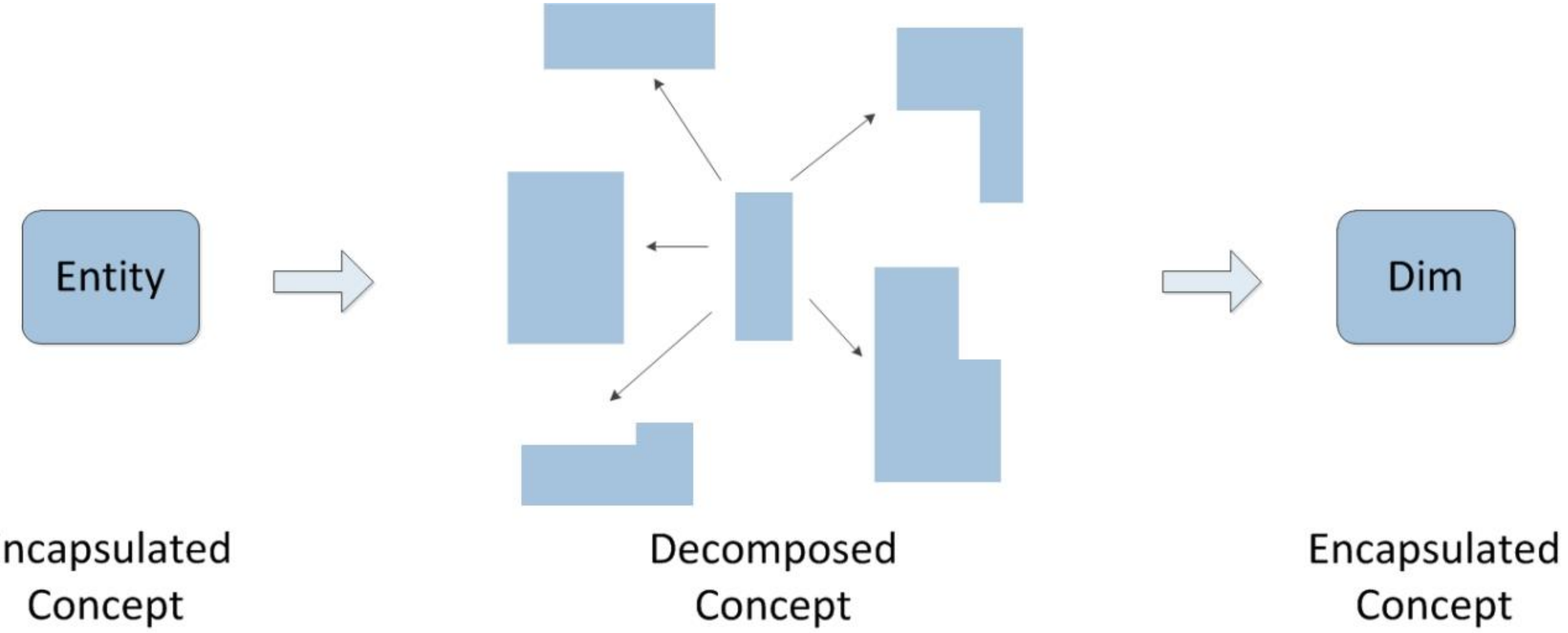
Data Vault Disadvantages:

- Needs additional integration downstream
- Difficult data loading from Data Vault

Data Vault 2.0 Modeling
Unified Decomposition™, Hans Hultgren

Separate:

- Things that change from things that don't change
- Things that change independently from each other
- Things that are interpreted in different ways

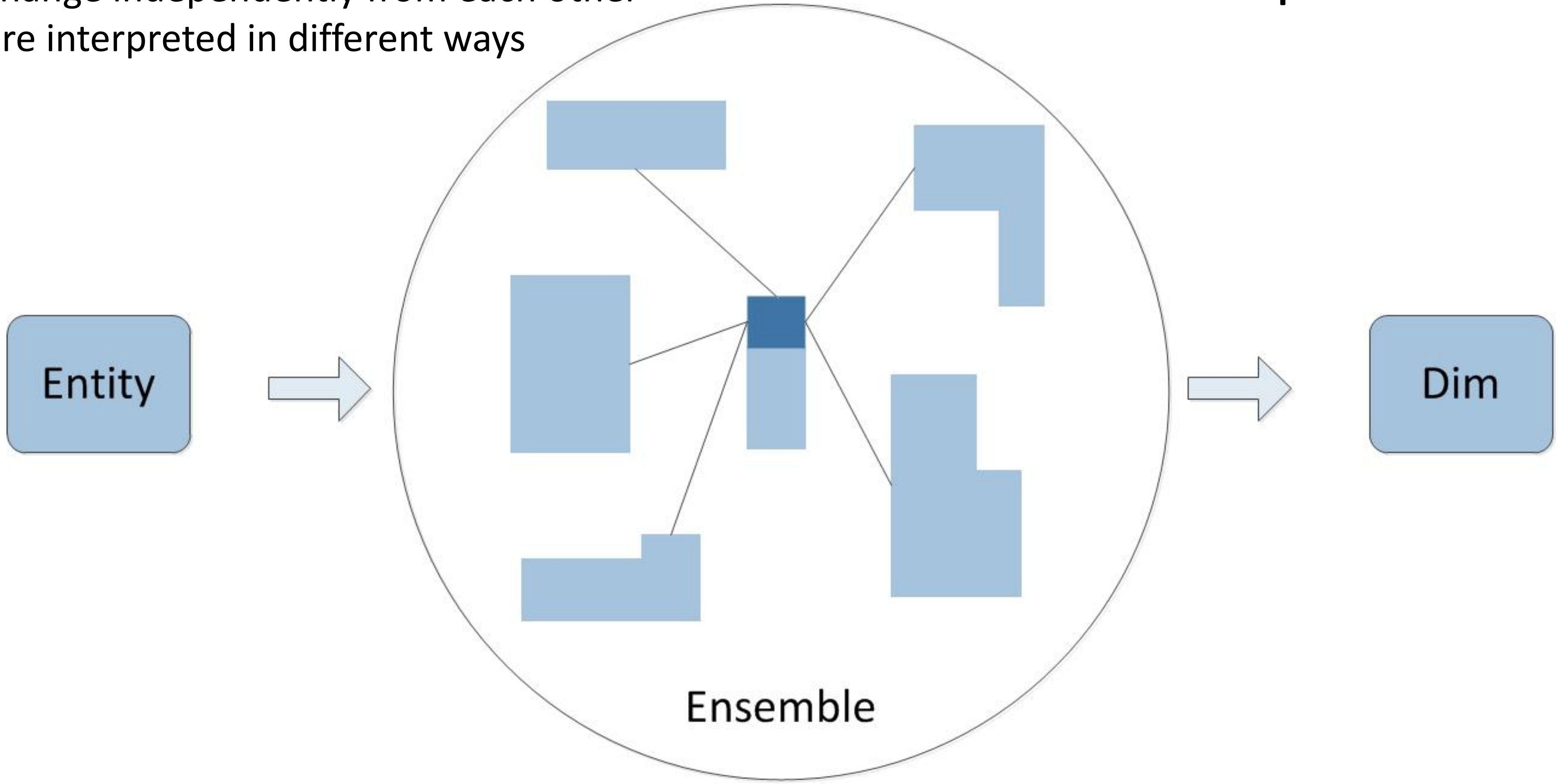


Data Vault 2.0 Modeling
Unified Decomposition™, Hans Hultgren

Separate:

- Things that change from things that don’t change
- Things that change independently from each other
- Things that are interpreted in different ways

All component parts act as a whole – an **Ensemble** or **Core Business Concept** and include three main types:

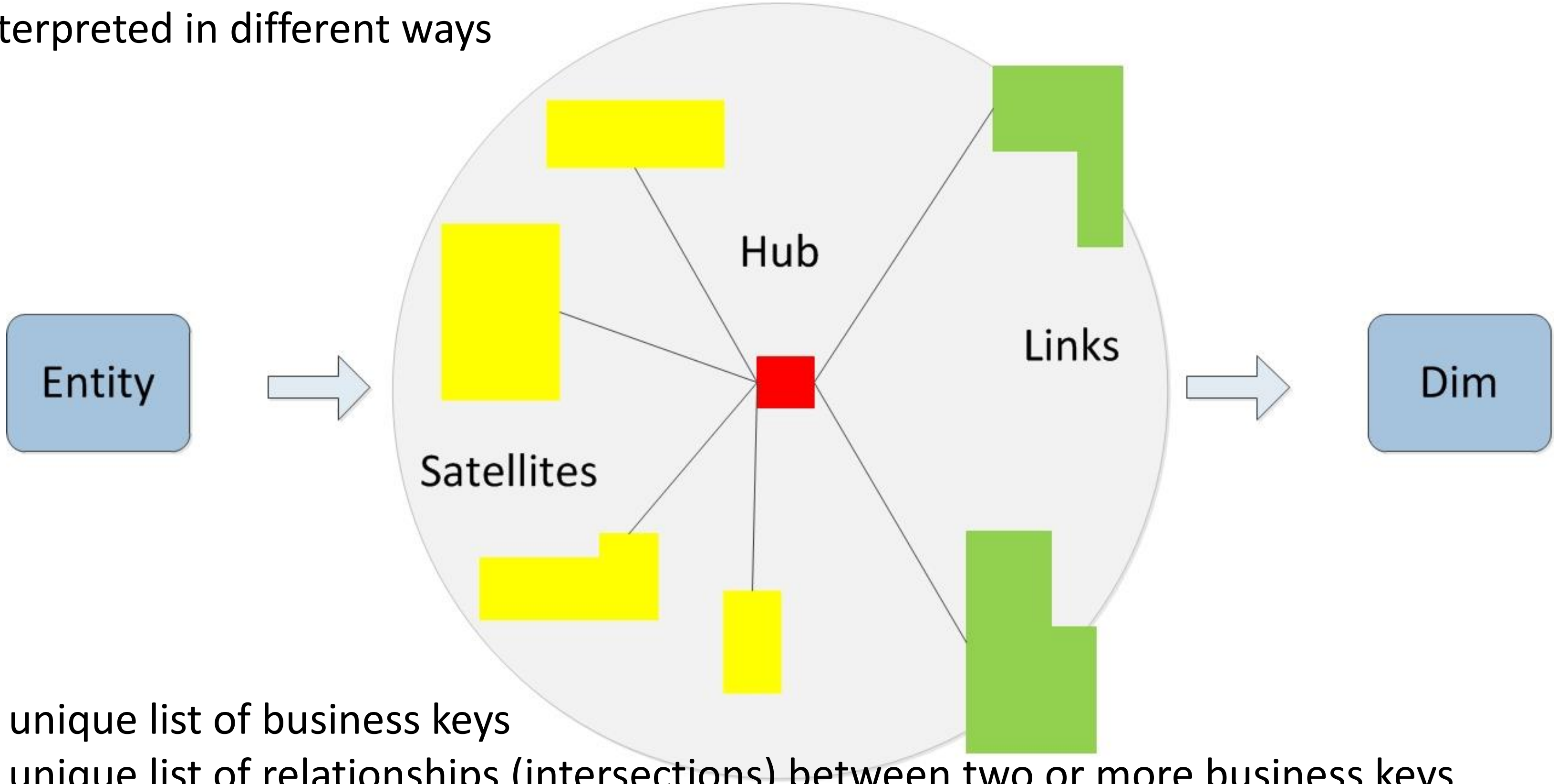


Data Vault 2.0 Modeling
Unified Decomposition™, Hans Hultgren

Separate:

- Things that change from things that don’t change
- Things that change independently from each other
- Things that are interpreted in different ways

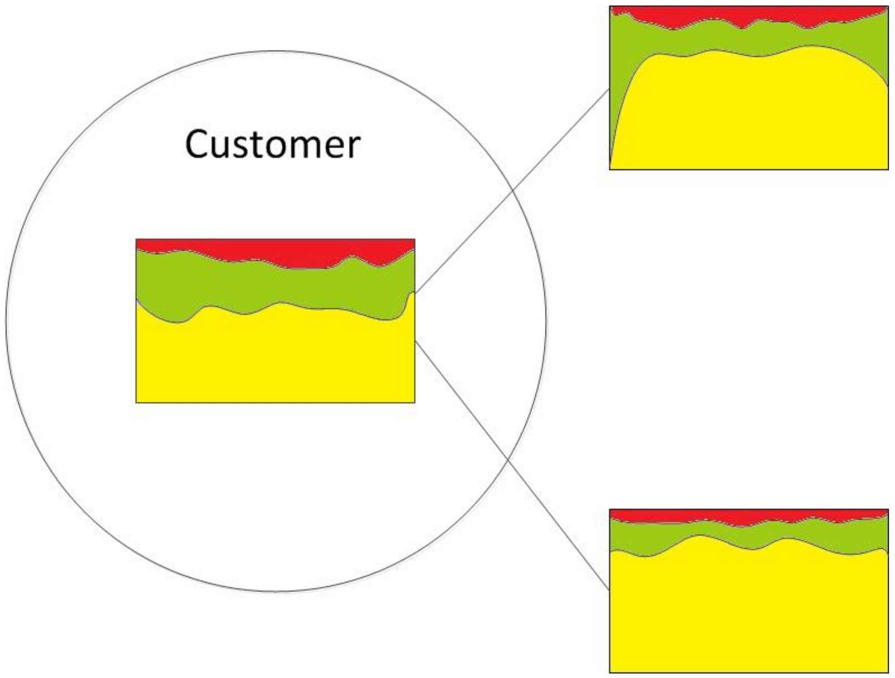
All component parts act as a whole – an **Ensemble** or **Core Business Concept** and include three main types:

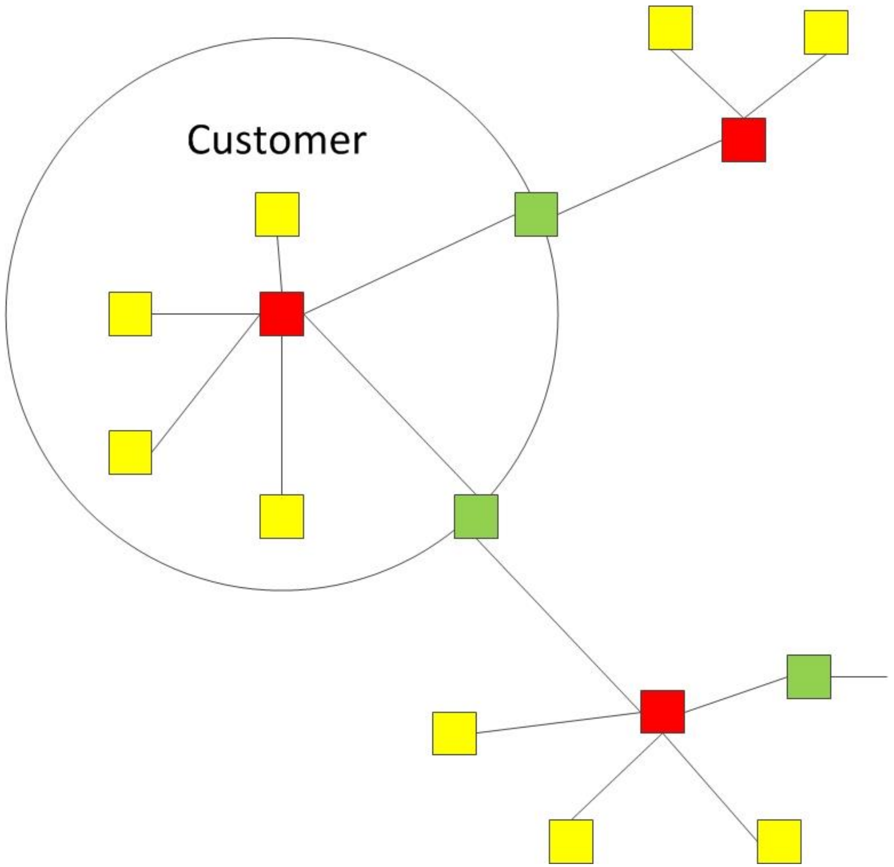


- **Hub** – unique list of business keys
- **Link** – unique list of relationships (intersections) between two or more business keys
- **Satellite** – delta driven qualitative and quantitative information (data that changes over time)

Data Vault 2.0 Modeling

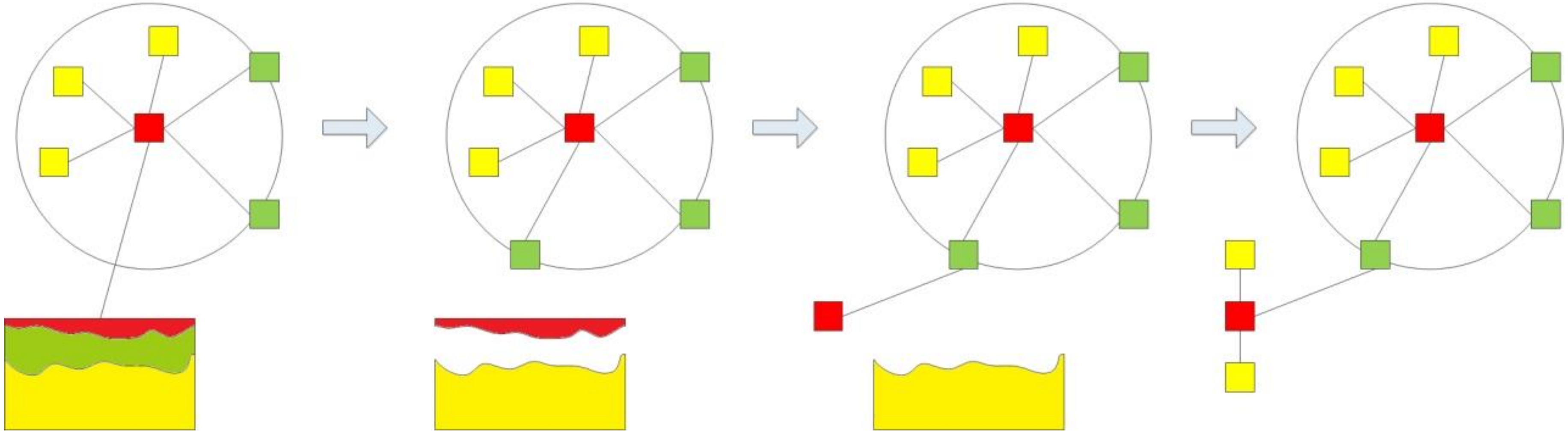
Example – Create a new model



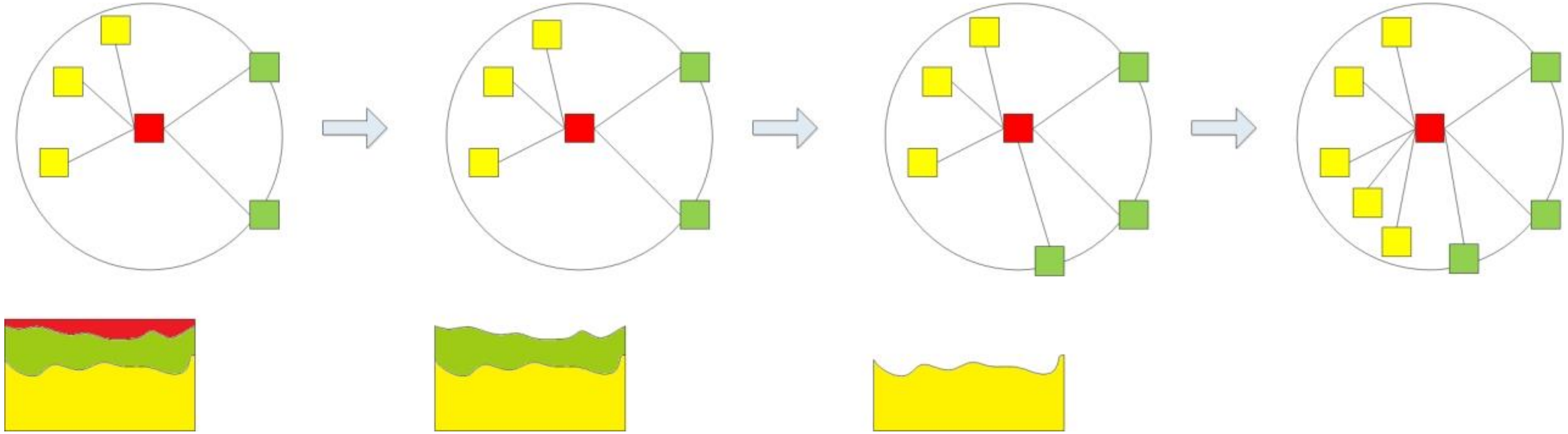


Data Vault 2.0 Modeling

Example – Extend an existing model

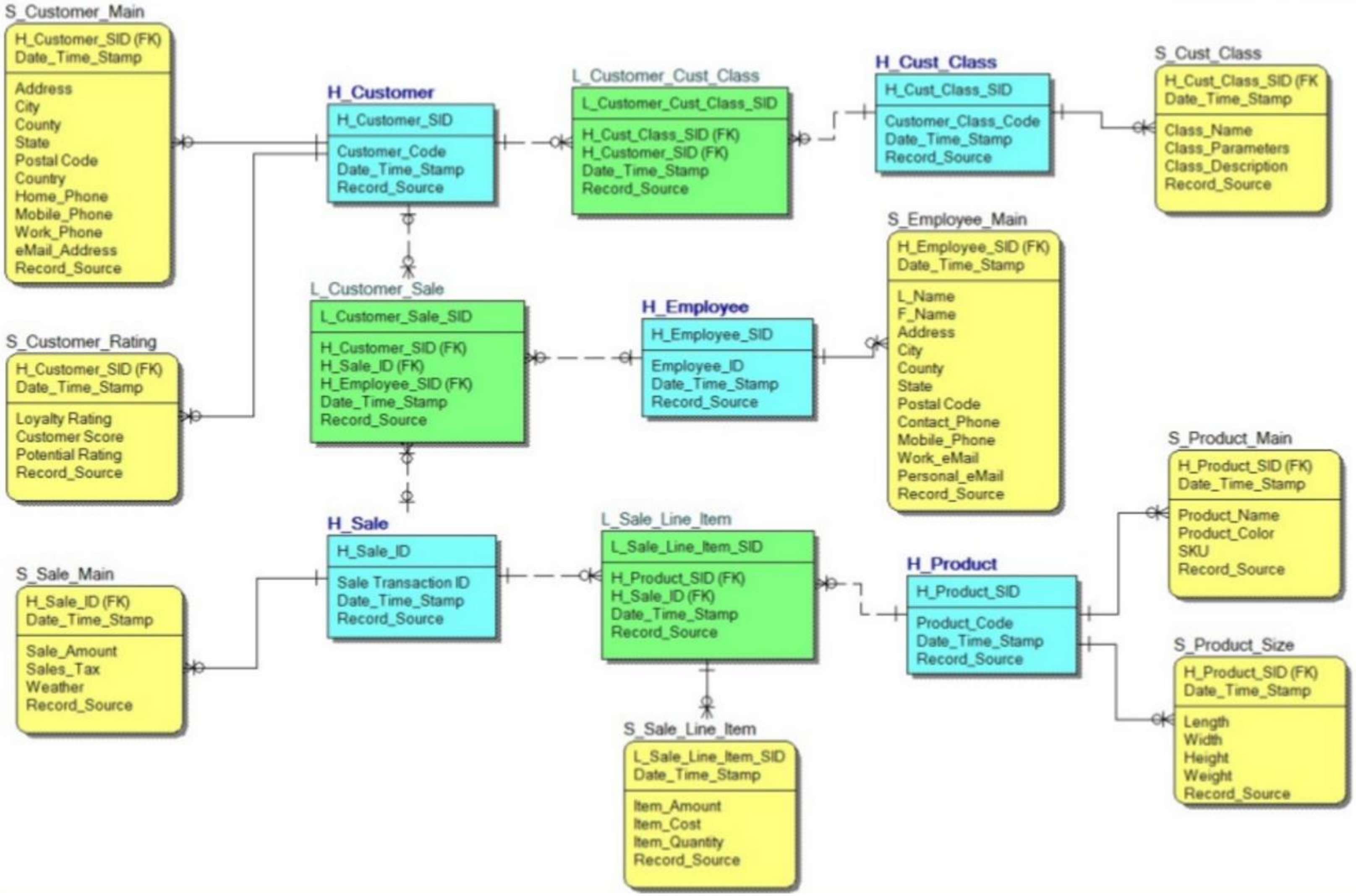


Adding new Concept to an existing model



Adding existing for the model Concept from another source

Data Vault Modeling Example



S_Customer_Main
H_Customer_SID (FK)
Date_Time_Stamp
Address
City
County
State
Postal Code
Country
Home_Phone
Mobile_Phone
Work_Phone
eMail_Address
Record_Source

H_Customer
H_Customer_SID
Customer_Code
Date_Time_Stamp
Record_Source

L_Customer_Cust_Class
L_Customer_Cust_Class_SID
H_Cust_Class_SID (FK)
H_Customer_SID (FK)
Date_Time_Stamp
Record_Source

H_Cust_Class
H_Cust_Class_SID
Customer_Class_Code
Date_Time_Stamp
Record_Source

S_Cust_Class
H_Cust_Class_SID (FK)
Date_Time_Stamp
Class_Name
Class_Parameters
Class_Description
Record_Source

S_Customer_Rating
H_Customer_SID (FK)
Date_Time_Stamp
Loyalty Rating
Customer Score
Potential Rating
Record_Source

L_Customer_Sale
L_Customer_Sale_SID
H_Customer_SID (FK)
H_Sale_ID (FK)
H_Employee_SID (FK)
Date_Time_Stamp
Record_Source

H_Employee
H_Employee_SID
Employee_ID
Date_Time_Stamp
Record_Source

S_Employee_Main
H_Employee_SID (FK)
Date_Time_Stamp
L_Name
F_Name
Address
City
County
State
Postal Code
Contact_Phone
Mobile_Phone
Work_eMail
Personal_eMail
Record_Source

S_Sale_Main
H_Sale_ID (FK)
Date_Time_Stamp
Sale_Amount
Sales_Tax
Weather
Record_Source

H_Sale
H_Sale_ID
Sale Transaction ID
Date_Time_Stamp
Record_Source

L_Sale_Line_Item
L_Sale_Line_Item_SID
H_Product_SID (FK)
H_Sale_ID (FK)
Date_Time_Stamp
Record_Source

H_Product
H_Product_SID
Product_Code
Date_Time_Stamp
Record_Source

S_Product_Main
H_Product_SID (FK)
Date_Time_Stamp
Product_Name
Product_Color
SKU
Record_Source

S_Product_Size
H_Product_SID (FK)
Date_Time_Stamp
Length
Width
Height
Weight
Record_Source

S_Sale_Line_Item
L_Sale_Line_Item_SID
Date_Time_Stamp
Item_Amount
Item_Cost
Item_Quantity
Record_Source

Sales Model example

The Data Vault Model allows to find appropriate balance between the need for real-time loading and the need for fully integrated data downstream in the data warehouse.

Why Data Vault?

- Much more stable model over the time (separated structure and content)
- Very flexible for changing business requirements even for back periods
- Quick data loading, suitable for real-time solutions
- Scalable solution, quickly integrate data on different platforms
- More readily absorb changes (improved agility)
- Respond well to new subject areas (incremental build)
- Innately manage historical time slices of data (historization)
- Provide full traceability back to source feeds (auditability, GDPR)
- Grow and adapt with minimal impact (lower TCO)
- Integrate, align & reconcile data (enterprise integration)
- Track, manage and report on exceptions (provides feedback loop)

General Architecture Implementation Specifics

- Centralized vs Distributed Data Warehouses
- Dependent vs Independent Data Marts
- Physical vs Virtual Data Marts
- Relational vs Multidimensional Data Marts
- Usage of Federated Data Sources
  - Horizontal Federation
  - Vertical Federation

Terminology

- Data Vault
- Single Source of Truth
- Single Source of Facts
- Unified Decomposition
- Horizontal / Vertical Data Federation
- Dependent / Independent Data Mart
- Physical / Virtual Data Mart

Project Setup

- Teams establishment
- Scope (depends on source data, could be altered to similar one)
  - A commercial bank asked your team to analyze their core banking system (CBS) data and to build a data warehouse, suitable for analyses
  - The bank provided the CBS model and the CBS model description
  - The proposed data modeling tool is Oracle Data Modeler (free)
  - The bank is available for providing additional information and to detail the requirements

Project Setup

- Project Documentation should include at least:
  - Short description of the task (requirements)
  - Justification of selected DWH building approach
  - Short description of the ETL process
  - Source data model (ERD)
  - Staging Area data model (ERD)
  - DWH data model (ERD)
  - A set of 2-3 useful (from business perspective) reports for the bank management
- Next steps
  - Download provided info and install a data modeling tool
  - Analyze the model and prepare a list with open questions
  - Prepare CBS model draft
  - Clarify all open questions