DATA VAULT AND HQDM PRINCIPLES

Geoff Collins
Georgia Southern University
gc010007@georgiasouthern.edu

Matt Hogan
Georgia Southern University
mh07183@georgiasouthern.edu

Mark Shibley
Georgia Southern University
ms08611@georgiasouthern.edu

Connard Williams
Georgia Southern University
cw06983@georgiasouthern.edu

Vladan Jovanovic
Georgia Southern University
vladan@georgiasouthern.edu

ABSTRACT

The paper explores applicability of high quality data modeling (HQDM) principles for data vault modeling.

Keywords

Data model quality, Data Vault

INTRODUCTION

Despite over 30 years of industrial experiences with data modeling and numerous fine guidebooks (Barker 1989, Tillman 1993, Schmidth 1999, Silverston 2001, Halpin and Morton 2008 among other) and some focused on reusable models (Hay 1995, Silverston and Agnew 2009, Blaha 2010, Hay 2011) there is no generally accepted set of explicit principles for data modeling that has withstood the test of time (Hay, 1995). Only a few sources explicitly addressed quality of data models (Bruce 1992, Reingruber 1994, and to some extent Simsion and Witt 2005). A high quality data models (HQDM) are desirable, and to assure data quality data models need to be constructed according to principles defining model quality. Recently (West 2003, West 2011) elaborated six such HQDM principles by exposing typical problems in data modeling for databases. While additional characterization of data modeling principles deserves its own treatment, our objective in this paper is to explore West’s HQDM principles in the context of Data Vault (Linstedt 2011) modeling for data warehouses.

To be of high quality data models must satisfy at least the following set of desirable properties (West, 2011):

- Meet the data requirement.
- Be clear and unambiguous (not just to the authors)
- Be stable in the face of changing data requirements.
- Be flexible in the face of changing business practices.
- Be reusable by others.
- Be consistent with other models covering the same scope.
- Be able to reconcile conflicts with other data models

The HQDM principles based on (West, 2003, West 2011) are succinctly expressed (Lubyansky, 2009) as:

1. Entity tables should represent the underlying nature of an object, not its role.
2. Entity tables should be part of subtype/super-type hierarchies.
3. Activities and associations should be entity tables, not relationships.
4. Relationships should only attach the “noun” entity tables (like people, places, and assets) to activity and association entity tables.
5. Candidate attributes of tables should be checked to see if they are foreign keys.

6. Tables other than pure intersection tables should have artificially generated unique identifiers.

The HQDM principles are outlined in a framework that is developed on an ontological basis. That is, rather than scanning for repeating patterns and anomalies in the data and eliminating them by introducing new entities (normalization), West proposes that data modeling be approached by identifying what the subjects-to-be-modeled actually represent in their respective contexts. When analyzing what the data represents in practice, this ontological approach already entails the broad scope of business. Taking the normalization approach can lead to analysis in a vacuum and lacks the power of integration across the whole business. Hence, the ontological approach has value for designing the enterprise architecture as well as the data models.

WHAT IS DATA VAULT MODELING?

Data Vault modeling is developed by Linstedt (Linsted, 2011) as a means of structuring data for data warehouses as systems of permanent records, to absorb structural changes without any data alterations, a characteristics that separates it from other modeling approaches (Jovanovic and Bojicic, 2012).

Data Vault modeling is based on the concepts of hubs, links, and satellites:

- The purpose of a Hub is to store no more than the core business concept. Hubs never contain foreign keys but store the business key, a uniquely generated sequence id, the timestamp when the data was loaded (first recognized), and the source of the data.
- A Link defines a relationship between business concepts. Links consist of the uniquely generated sequence ids from the hubs (one or more), a warehouse sequence id, the timestamp when the data was loaded, and the data source.
- Satellites contain the descriptive information. They provide context to the business concepts and are used to track changes over time. Multiple satellite entities can describe one hub or one link, separated by crate of change In addition to the descriptive attributes, satellites contain: the uniquely generated sequence ids from the hub, or from the link to which it is attached, the timestamp when the data was loaded, and the source of the data.

A REPRESENTION OF HIGH QUALITY DATA MODELS CONCEPTS USING DATA VAULT

The first principle for high quality data modeling entity types is that entity types should represent, and be named after, the underlying nature of an object, not the role it plays in a particular context (West, 2011). Defining the underlying nature of an entity reduces redundancy and increases flexibility. Figure 1 below does not follow this principle, and while it may model a realistic business relationship, flexibility is lacking. If another employee role is added (such as janitor), the structure of the model needs to be modified. Figure 2 is a more efficient representation. This principle is embedded in Data Vaults. Data Vault Hub and Links are basic entities with minimal attributes describing the basic nature of an object. Satellites are descriptors and contextualize the Hub and Link.

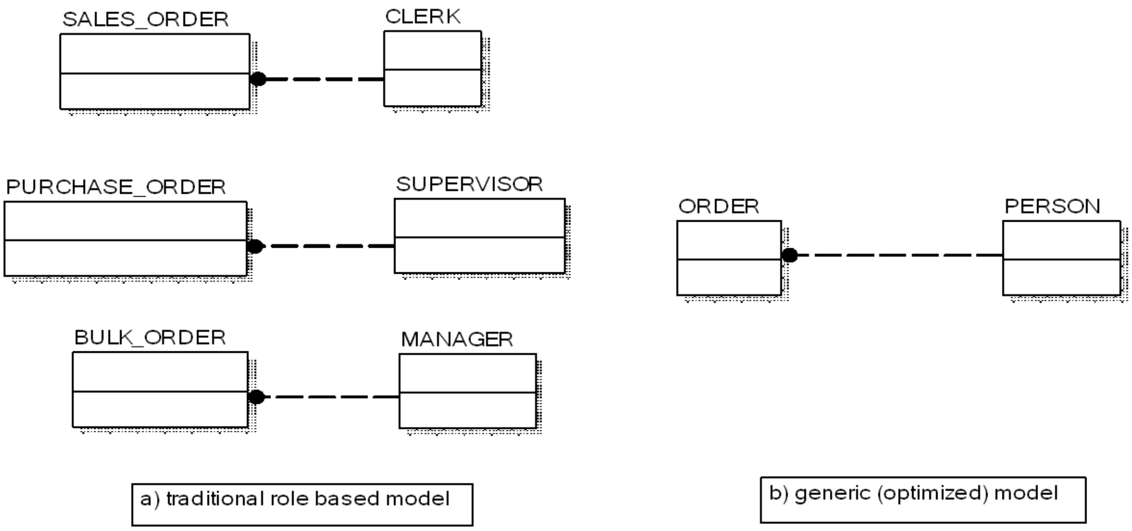

Figure 1. Role oriented (a) vs. generalized data model (b)

But what if a single data vault model needs to integrate the three existing sources (each of the pairs of role based entities is treated as a separate data source) represented by Figure 1 (a), into a data warehouse? Model (b) is still a basis for a data vault, and Figure 2 gives a canonical data vault representation (for emphasize Hubs are shown on blue, links on red and satellites on yellow background) if following assumptions hold: i) employee_id is a key for all persons regardless of role, and ii) that all instances of orders are unique and identified by business key (unique to the source that is). In Figure 2 satellites are declared per source with corresponding attributes bundled for each under single name for example Manager_attributes.

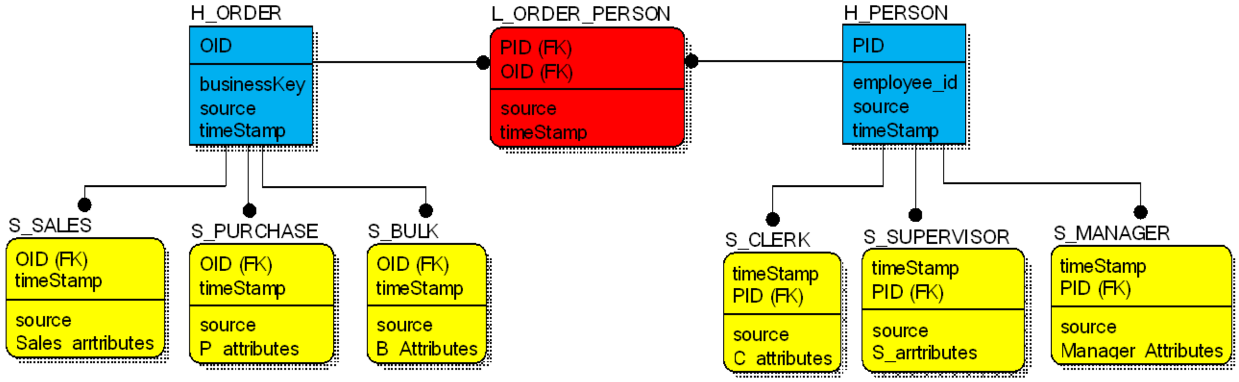

Figure 2. Data vault representation of integrated hubs

The second HQDM principle is that entity types should be part of a sub-type/super-type hierarchy of generic entity types in order to define a universal context for the model, and to avoid duplication of concepts and data. So at the outset we must note that HQDM principle 2, useful for databases may not be advisable for data warehouses. The core innovation in a Data Vault is establishment of permanent identities for fundamental entities (Hubs) i.e. for what exists empirically (an ontological basis) and that identity is literally the key of top level supertypes. Expressing deep inheritance relationships with Data Vault is not straight forward, Jovanovic and Bojicic 2012 addressed but one possible approach for conceptual modeling with data vault notation and other similar ideas for physical data vault modeling were presented elsewhere (Linstedt 2011, Graziano 2011, Hultgren 2012, and Mangano, 2013). In our correspondence with Linstedt, he specifically noted “To support inheritance would require breaking the modeling constructs and thereby not producing a Data Vault model at all. You would basically break flexibility that the Data Vault offers.” A simple sub-type/super-type relationship can be modeled with the Hub serving as the super-type and one or more Satellites servings as the sub-types, say S_Organization and S_Minor (in Figure 3).

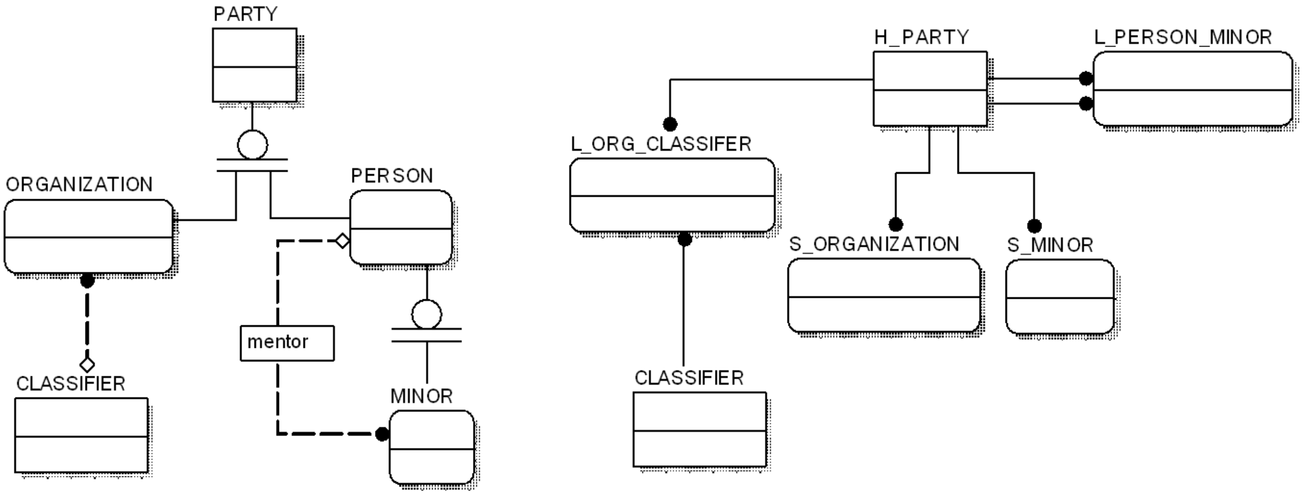

Figure 3. An example of spource data mopdel with generalization hierarchy and its data vault model counterpart

The third HQDM principle states that the associations between entities should be stored in separate entities, and is also one of the core concepts in data vaults. Whereas in HQDM terminology such as association is called a "verb," in data vault this is known as a link. To illustrate this let us say that a Person (serving as a Clerk) may complete a Sales type Order. At the same time, another Person (serving as a Supervisor) may submit a Purchase type Order. In such situation, there are two different associations between the Person and Order entities and need to be represented in their own Link entities, see Figure 4.

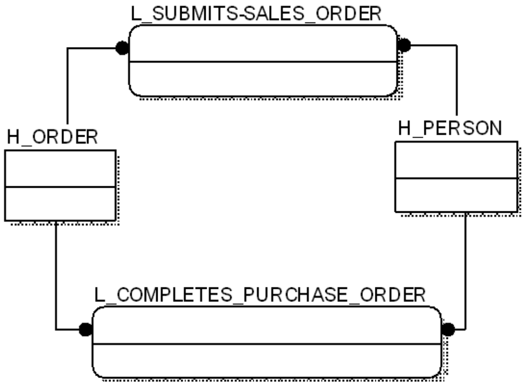

Figure 4. Data Vault representation of Person – Order relationships as links (modified from Figure 2.)

The fourth HQDM principle states that entities storing business concepts and their attributes (also referred to as the "nouns") should only be connected to one another through the association entities (links) described previously in the third principle. Using Figure 4 as a reference, the Person and Order entities are the "nouns" in HQDM terminology, whereas in Data Vault they are considered Hubs. As with the third principle, this is the same concept with different names.

The fifth principle states that candidate attributes of tables should be checked to see if they are foreign keys. Applying this HQDM principle to the Data Vault model translates to checking Hubs to ensure that only one business key, be it simple or composite, exists in each Hub. Where multiple business keys are found, the additional keys are moved out to create new Hubs. This requirement, though may not be obvious, is in direct support of normalization which produces higher normal forms by removing transitive and inter-key dependencies. Applying this principle literally results in sixth normal form, where each hub attribute has its own satellite (Jovanovic and Bojicic, 2012) removing all updating dependencies from attributes. If an attribute is a foreign key, it represents some other Hub entity and needs to be placed with the Hub entity for which it holds the business key. The Hub entity must never contain foreign keys. If the Hub structure is compromised (i.e., the modeling standards are not adhered to), then the integrity of the data and the flexibility of the model are immediately compromised (Lindstedt 2011). The problem to avoid here is encoding business rules directly into an entity's attributes. Consider one potential set of attributes for the entity Party in Figure 5.

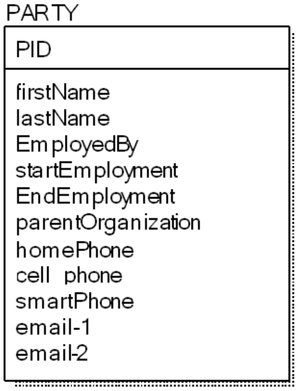

Figure 5. Non-normalized Party Entity

Note that the attributes of Party relate to different business concepts such as Contact Information and Employment. This is further complicated if Party may be either an Organization or Person. To adhere to the HQDM rule above, the Party entity needs to be expanded to remove business logic out of the Party entity and represent it relationally, for example as shown in Figure 6.

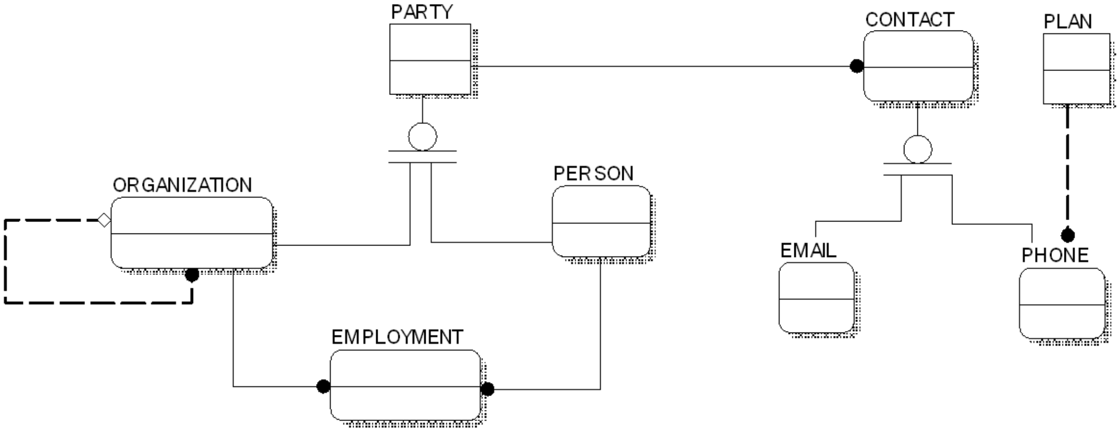

Figure 6. Expanded Party Entity

Focusing mainly on the Contact business view for Party, we have the partial Data Vault model (using IDEF1x notation) to show the impact in Figure 7.

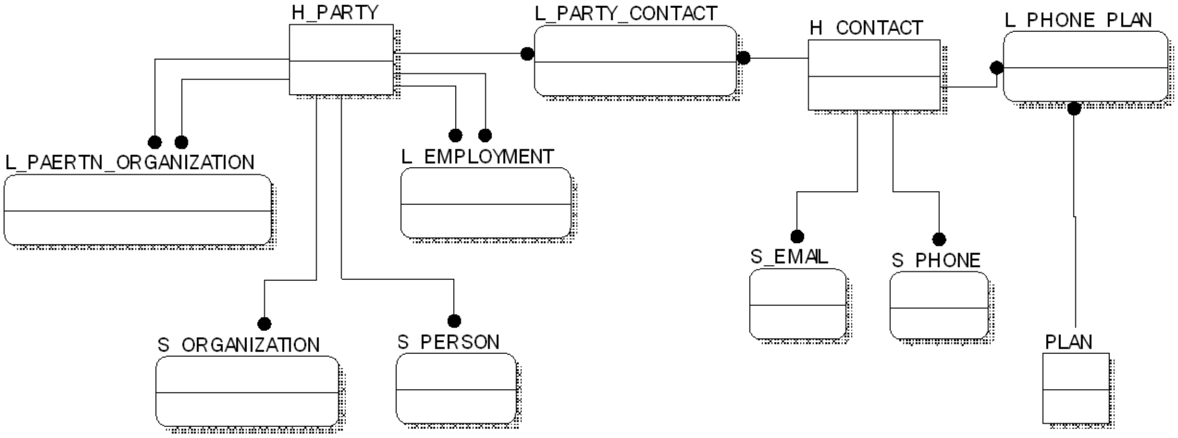

Figure 7. Partial Data Vault model emphasizing removal of foreign keys to create Contact Hub

The sixth HQDM principle states that entities used for storing business concepts and their attributes should use artificially generated unique identifiers. In order to avoid the coupling of business logic to entity identifiers West (2011) suggests that data modelers use a new attribute to serve as an artificially generated unique identifier for each entity. While analysis of an entity may uncover existing attributes that are unique to an entity, use of these attributes as identifiers can often limit the flexibility and adaptability of the model in light of future change. The only exception to this rule would be when an entity is serving to link (or associate) two other entities and has a derived unique identifier that results directly from the combination of the other entities' artificially generated unique identifiers. To see how Data Vault supports this principle, we can look back at Figure 2. In contrast to its relational counterpart, the entities in this Data Vault model will always have artificially generated unique identifier attributes. More specifically, the Order Hub and Product Hub have a database-managed unique identifier attribute. The Order Product Link and the Order Satellite will generally either also have their own unique identifier attributes or they can have a combination of the associated entities' artificially generated unique identifiers as done in Figure 2. A recommended method of uniquely identifying satellite entities in Data Vault is to use a load timestamp as part of the unique identifier, which also supports the sixth principle.

CONCLUSION

Applying the principles of HQDM to Data Vault modeling was surprisingly direct. Entity types (hubs) should be created based on underlying object nature, and their associations stored separately (as links). Candidate attributes should not be foreign keys, and only one business key (possibly composite) exists in each hub. The HQDM principle states that entities should use artificially generated identifiers. Data Vaults also require unique sequence identifiers for Hubs and consequently Links and their Satellites (identifier composed of parent key and time). Data Vault modeling advises against multiple layers of hierarchical relationships as data vaults are designed to be able to flexibly satisfy future data requirements. Among future research directions, related to HQDM (and similar principles) and evolving data vault modeling technology, are rigorous formalization/characterization of data model quality attributes and empirical evaluations of long term qualities of evolving data models (data warehouse data vault schemas) built following such principles.

REFERENCES

1. Barker R. (1989) CASE*Method: Tasks and Deliverables. Addison-Wesley,
2. Blaha M. (2010) Patterns of Data Modeling, CRC Press,
3. Bruce T. (1992) Designing Quality Databases with IDEF1X Information Models, Dorset House,
4. Graziano, K. (2011) Introduction to Data Vault Modeling, http://kentgraziano.files.wordpress.com/2012/02/introduction-to-data-vault-modeling.pdf,

5. Halpin, T. and Morgan, T. (2008) “Information Modeling and Relational Databases, 2 edition, Morgan Kaufmann,
6. Hay, D. (1995) Data Model Patterns: Conventions of Thought, Dorset House Publishing,
7. Hay D. (2011) Enterprise Model Patterns, Technics Publications,
8. Hultgren, H. (2011). Modeling the Agile Data Warehouse with Data Vault, New Hamilton,
9. Jovanovic V. and Bojicic, I. 2012. Conceptual Data Vault Model, Proceedings of the Southern Association for Information Systems Conference, Atlanta, GA, USA March 23rd-24th: 131-132.
10. Lubyansky A. (2009) Using Data Model Patterns to Build High-Quality Data Models, www.ppc.com/assets/pdf/white-papers/Data-Model-Patterns.pdf.
11. Linstedt, D (2011) Supercharge Your Data Warehouse, Create Space Independent Publishing,
12. Mangano D. (2013) The Integrated Data Hub- Next Generation Data Warehouse,
13. Ponniah, P. (2007) Data Modeling Fundamentals: A Practical Guide for IT Professionals, John Wiley,
14. Reingruber M., Gregory W. (1994) The Data Modeling Handbook- A Best Practice Approach to Building Quality Data Models, John Wiley,
15. Umanath, N. and Schamell R. (2017) Data Modeling and Database Design, Thomson,
16. Silverston, L. (2001) The Data Model Resource Book: A Library of Universal Data Models by Industry Type- Volume 2 revised edition. John Wiley,
17. Silverston L (2009) The Data Model resource Book- Volume 3, Universal Patterns for Data Modeling, John Wiley,
18. Simsion, G. and Witt G. (2005) Data Modeling Essentials, 3ed., Morgan Kaufmann,
19. Schmitdt B., (1999) Data Modeling for Information Professionals, Prentice Hall,
20. Tillman G. (1993) A Practical Guide to Logical Data Modeling, McGraw Hill,
21. West, M. (2003). Developing High Quality Data Models (Version 2.0). EPISTLE.
22. West, M. (2011). Developing High Quality Data Models, Morgan Kaufmann.