



*[www.prudenza.nl](http://www.prudenza.nl)*

## The next generation EDW 2/3

*Balance of data warehouse process lies in Big Transformation*

Date published: November 25, 2008  
Published in: Database Magazine (Netherlands)  
Authors: Ronald Damhof, Lidwine van As

## The next generation EDW (2)

In the first part of this series of three articles, 'The next generation EDW', which was published in DB/M 5, we presented an EDW architecture that can meet the (continually) changing demands made of a modern EDW environment. It was observed that a method like the Data Vault offers an excellent framework for this sort of developments.

In terms of methodology, however, the Data Vault is still primarily focused on the central data warehouse. While the architecture offers a clear philosophy with regard to data quality, auditability, traceability, performance and standardisation, it provides few solid handholds or guidelines, if any, for the creation of data marts. And this despite the fact that – due to the nature of the Data Vault itself – this is precisely where the balance of the data warehouse process will lie: the 'Big T' or 'Major Transformation'.

We have noticed that the current uncertainty concerning the correct approach has held many people back from opting for next generation architecture. In the first part, we portrayed the next generation EDW pretty much as a black box. In this article, however, we take a peek into that box, while also proposing an approach for the Big T.

### Overall architecture

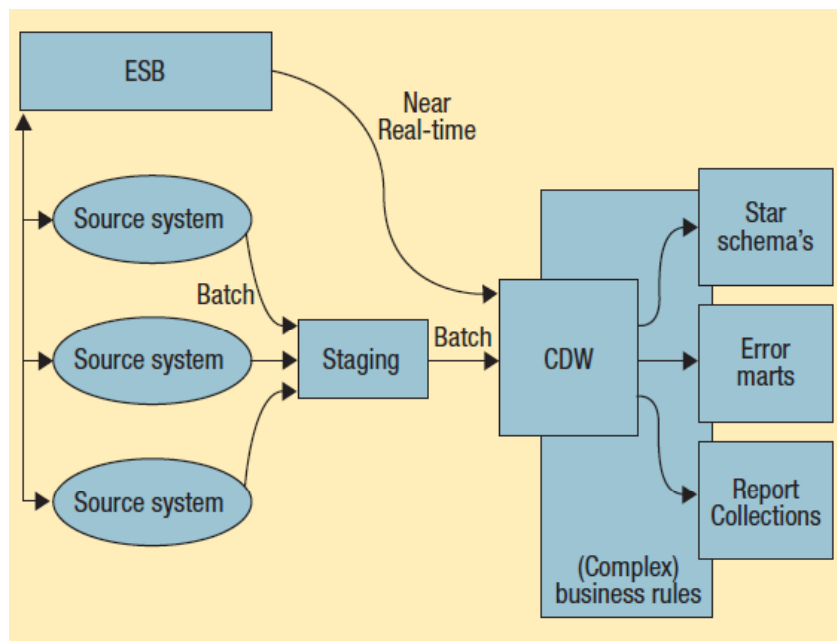


Figure 1: Next generation architecture (Source: D.Linsted, the business of Data Vault modelling)

Figure 1 shows the general architecture of the next generation EDW. The staging is a brief storage layer, which – in terms of structure – is as close as possible as being one-on-one to the source. This layer has already been extensively described by Dr. Kimball [1]. The CDW is the system of fact: the stored data in this case is as close as possible to that received from the source. This matter was examined at length in the first part of this 3-part series.

In terms of data logistics, the central data warehouse (CDW) lies beyond the 'push-pull point': a supply-driven approach is adopted up to and including the CDW, while everything beyond is demand-driven. Action is taken beyond the CDW only in the event of an actual client request, while all action taken is dictated by this client request.

### **Uncharted territory: the Big T**

As previously mentioned, the combination of the system-of-fact approach and the '100% of the data 100% of the time' principle, shifts the balance of data logistics from the source to the client to between the CDW and the data marts. To determine how we might best design this component, it is therefore essential that we establish why we have adopted this new approach to EDW. A few of the problems encountered with previous generations of data warehouses were extensively examined in part 1, in DB/M 5. These lay primarily in the fields of manageability, maintainability, extendibility, traceability and scalability. These drivers will therefore also largely determine the structure of the Big T, however, in doing so, we shall also strive to achieve:

- Transparency, clear structures, a clearly established position within the processing route for each component;
- Reuse of functionality (therefore restricting duplication);
- Simplicity of processing: Rather three simple procedures than a single complex one; modularisation of functionality. It should also be noted that the simplicity of an object or a procedure partly determines its reusability – we have learned from the field of object-orientation that simple objects offer a greater measure of reusability than complex ones;
- Standardisation; not only of processing stages (in other words, limiting the customisation of functionality), but also the underlying data model;
- Traceability; the same traceability and history-management requirements apply to the results of data derivation (business rules) and integration processes as to source data;
- Efficiency and performance optimisation; every possible measure is taken to make 'efficient' use of the available resources. For instance, there is no point in spending processing time on data that is going to be filtered out at a later stage. Careful attention therefore needs to be devoted to the sequencing of the process. The more efficient the data logistics process is, the more favourable its performance;

- Testability; there is a substantial threat of steadily increasing entwinement of data logistics, which may result in a steadily degenerative system in terms of testability as the EDW expands.

Let us first pause to examine more carefully what processing occurs in this Big T. (Please consult [1] for more in-depth insight into the ETL process). On the one hand, data-oriented processing is carried out: Processing that is solely related to the data itself, and therefore has an impact on either the data content, or the composition (and magnitude) of the dataset (Dr. Kimball refers to this as 'Cleansing & Conforming'). Consider, for instance: Checking and handling data quality issues; filtering/selection of data for further processing; integration and consolidation of raw data; data derivation. As the processing to be carried out does not impose any requirements on the modelling of the data, we are free to decide for ourselves. In addition, a series of processes is performed – closer to the data mart – which is designed to present previously processed data to the user in an adequate manner (which Dr. Kimball refers to as 'Data Delivery').

These processes are largely dictated by the requirements imposed by the end-user tooling; the method of data modelling is one of them.

Consider, for instance: remodelling into dimensions and fact tables (including compilation of aggregations); building cubes; flattening into a flat file, for data mining applications for instance, and any other conceivable user formats.

### **CDW+**

With a view to ensuring that the Big T remains transparent, it is only logical that it be laid out in a manner that reflects this division, so that each function and data element can be found in a logical location. We therefore subdivide it into a data-oriented CDW+ environment, and presentation-oriented Staging Out area (STO).

The execution of all data-oriented processing and the storage of the results are therefore housed in the CDW+ environment. As the processing we house in the CDW+ is not subject to any explicit requirements concerning the modelling method, we can decide on this for ourselves. In view of the benefits it offers in terms of history management and traceability, we also opt for Data Vault modelling in this case.

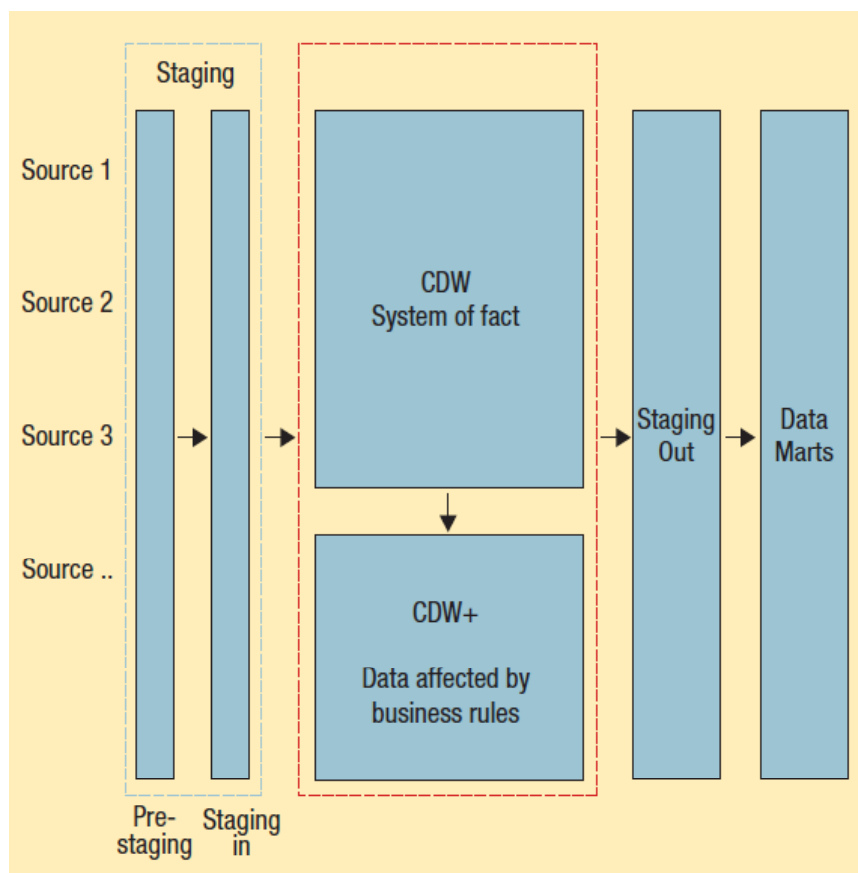
### **Transparency**

One danger that always looms when one defines the data logistics between a data warehouse and data marts, is the occurrence of entwinement of the data marts themselves. Data marts become entwined with one another in the event that data calculated in the one data mart is reused in another. This gradually renders data marts increasingly interdependent in terms of their data accumulation. These circumstances can be avoided, by not permitting data marts to reuse

one another's data; if no additional measures were taken, however, then this might lead to the duplication of derivation rules in several environments and at various stages of processing, with all the detrimental affects this entails.

The execution of all data-oriented processing is housed in the CDW+ environment

If we separate data-oriented processing entirely from presentation-oriented processing, and house it in the CDW+, then we create a central data warehouse for derived data as it were (see Figure 2). This maximises the options for the reuse of (the results of) business rules: Reusable elements are made available at the earliest possible stage, for reuse in other derivations and the composition of the data marts.



**Figure 2:** Next generation EDW architecture – CDW and CDW+.

### Efficiency and performance

The decision to perform processes that affect the composition and magnitude of the dataset as early as possible in the Big T, is also particularly prudent, with a view to promoting efficient process execution. After all, the less data that needs to be processed, the better it is for performance in general. Given that the data logistics functions to the CDW+ are Data Vault-

oriented, the optimum benefits of a zero-update strategy can be gained in terms of parallelization and scalability.

### **Truth**

In comparison to previous generations of EDW, the CDW+ comes closest to achieving the ideal of an integrated, clean data warehouse. However, this comparison does not entirely reflect the actual circumstances accurately. For instance, data from the CDW need not necessarily be fed through the CDW+ before being directed to a data mart. The raw data need only be fed through the CDW+ in the event that processing is required. The presence of the CDW+ therefore by no means implies doubling the storage capacity!

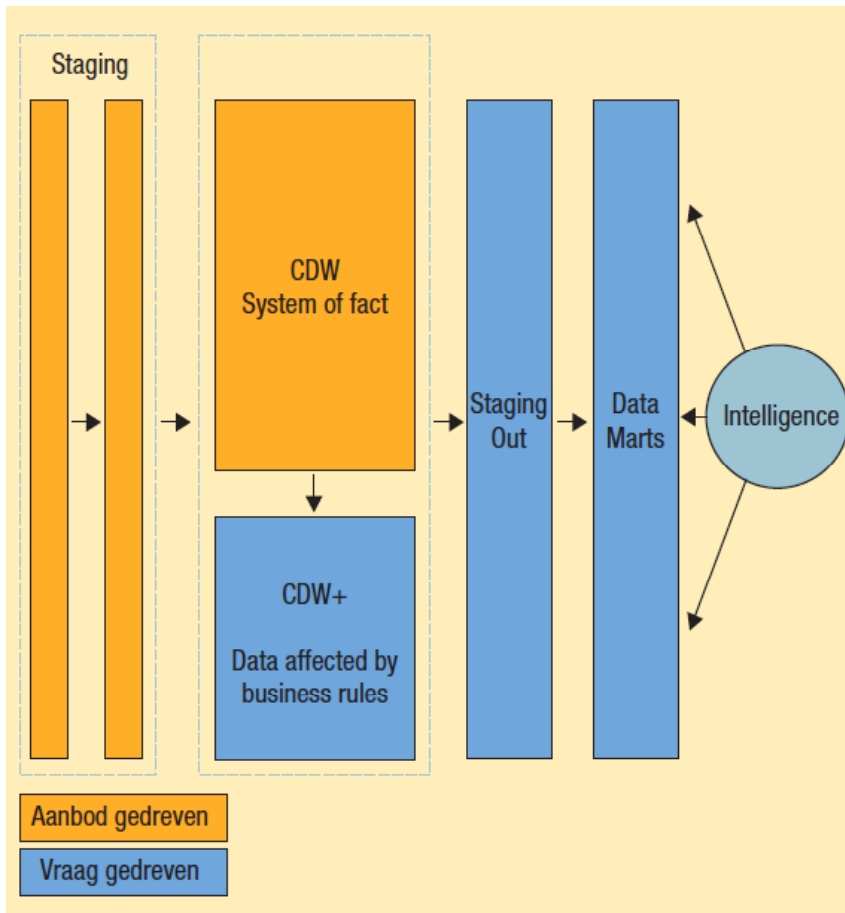
Nor does the CDW+ automatically contain a single version of the truth any longer. Due to the fact that the storage and processing of source data have been separated and housed in a CDW and a CDW+ respectively, the single version of the truth is no longer hard-wired in the design of the data warehouse, but several alternative strategies become available instead. Instead of striving to maintain a single version (while a switch to a different version remains an option, as the raw data remains available) several equivalent versions could exist alongside one another; or a 'common version of the truth' could be supported as an enterprise-wide version in addition to all local versions.

### **Standardisation**

Given that Data Vault modelling is applied for both CDW and CDW+, the two are basically extensions of one another. In practice, CDW+ would not even exist as a physically separate environment to the CDW, but would simply be an extension of it. For instance, a CDW+ satellite containing certain derived data could be attached to a CDW hub. However, with a view to ensuring manageability of the generation process, as well as clear domain separation between registration and enrichment in the interest of traceability, it is advisable to ensure that the two environments remain logically separated.

During the transformation from CDW/CDW+ to Staging Out, it makes no difference to the software whether the entity is called CDW- or CDW+: For instance, when modelling a dimension, it makes no technical difference whether the dimension data is accessed from a CDW- or a CDW+ hub – a hub is simply a hub. Transformations from CDW/CDW+ to Staging Out can therefore be largely standardised.

### Data marts (STO and DM), supply and demand- driven



**Figure 3:** Supply & demand driven.

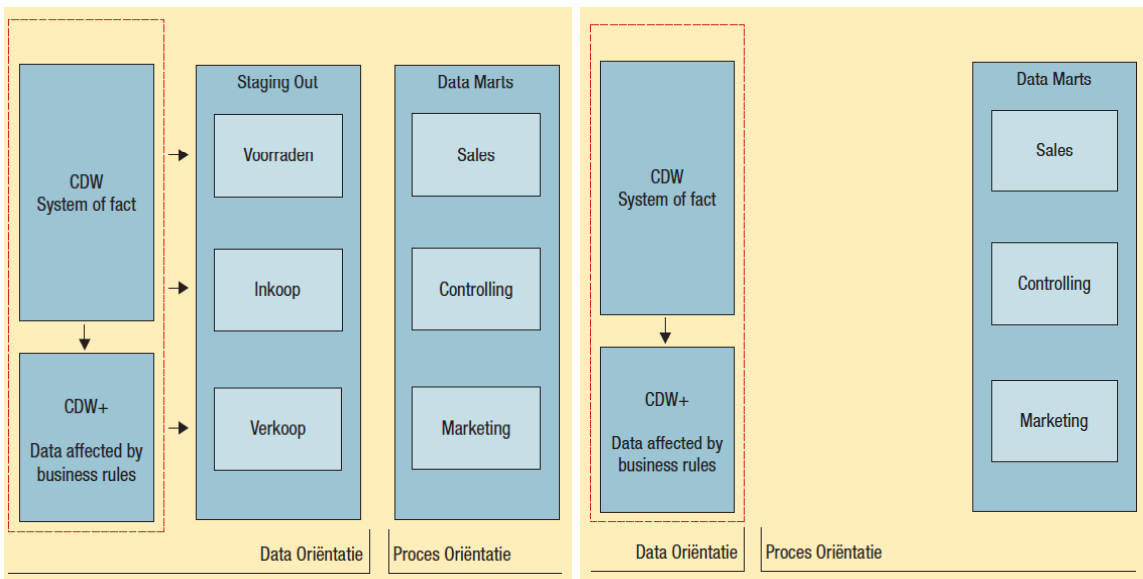
As shown in Figure 3, the data logistics and the data storage up to and including the CDW are supply-driven. This implies that the source largely dictates what is saved and in what manner. However, this changes beyond the CDW: A specific requirement – defined by the end-user – is necessary to justify the realisation of data logistics and data storage beyond the CDW. Up to and including the CDW+, all data is processed and prepared. In this respect, the CDW and CDW+ jointly form a 'staging-in' area for the data mart process. As mentioned in the previous sections, the CDW+ is modelled according to the Data Vault method; the Staging Out and the data mart layer are modelled in such a manner as to optimally serve the selected end-user tool: A query & reporting tool often requires dimensional (Dr. Kimball) storage, while a data mining tool prefers a flattened text file, and an analysis environment is best served by a multidimensional storage structure.

## Data & process orientation

Up to and including the Staging Out, data is stored in a data-oriented manner – regardless of the business process to be supported. This is essential – as previously mentioned in this article – to prevent the entwinement of data mart structures. From the data mart layer onwards, however, data is stored in a process-oriented fashion. How the data mart layer is ultimately implemented, depends on the Business Intelligence tooling and the metadata layer (e.g. the universe in Business Objects) provided. In the event of the use of tools with a rich functional metadata layer and advanced code generation (e.g. SQL/MDX) the data mart layer need comprise no more than the metadata layer of the BI tool in question. However, tools whose functional metadata layer and code generation are poor, call for a data mart layer which is implemented in the (R)DBMS. In such cases, the data mart layer can be ideally implemented as a (materialised) view layer in the Staging Out, in our opinion.

Staging Out and data mart layer are modelled in such a manner as to optimally serve the selected end-user tool

This distinction between Staging Out (data orientation) and Data Mart (process orientation) offers a vast range of possibilities, to which too little attention is paid in the field of data warehouse architectures, unfortunately. Figures 4 and 5 indicate the difference.



**Figure 4:** Staging out and data marts.

**Figure 5:** No staging out – data marts only.

Based on the assumption of three data-oriented datasets (stock, purchasing and sales data) and three process-oriented data marts (Sales, Marketing and Controlling) the following becomes an option:

- Some form of role-based security can be implemented relatively simply. If we assume that Marketing is to be granted insight into the margins, but not the net purchase prices per

supplier (the supplier dimension should not be visible to Marketing either), then this can be implemented relatively simply by allocating Marketing its own (materialised) view. In the alternative scenario (Figure 5), Marketing would have to be given its own data mart using ETL.

- The threat of proliferation of data logistics and data storage is implicitly counteracted, thanks to the fact that the different views that are to be allocated for the various business processes can be implemented on one and the same physical data mart. This in contrast to the creation of data marts with high levels of conformity, which would nevertheless have to be set up slightly differently for the various business processes. In the example shown in Figure 4, it is a relatively simple task to create the (materialised) views for Sales, Marketing and Controlling, with a short throughput time. In the example shown in Figure 5, however, this is far more difficult, as data marts have to be created on the basis of processes.
- By introducing this view layer, we are therefore introducing a push-pull point, which helps to ensure that not all users of the altered data mart need always experience the full repercussions of the changes. This is beneficial to management of the EDW, and change management in particular. If a fact is added to the purchasing data dataset, then – by definition – this change need only be implemented in the ETL step that this dataset completes. In the example shown in Figure 5, it may be the case that these changes need to be implemented in several ETL mappings. In terms of sustainability, the setup shown in Figure 4 also has a substantial influence on maintaining the testability of the EDW architecture.

And finally, the extent of entwinement up to and including the Staging Out is consciously subjected to stringent (architecture) control.

As indicated in the first part, sustainability remains an important characteristic of a next generation EDW in this respect also. What if the performance of a view in the data mart layer proves insufficient, though? Varied type-use (e.g., the one target group primarily submits ad hoc queries, while the other does much more reporting) can pose considerable difficulties when it comes to tuning physical datasets in the Staging Out. The solution in the next generation EDW is relatively simple; compartmenting. This amounts to isolating a group of users and the data that they use in a physical environment which is largely equipped with its own system resources (read: CPU, memory, storage). Within the context of the next generation EDW, however, there are various degrees of compartmenting:

- Materialising a data mart view on one or more physical Staging Out datasets.
- Materialising a data mart view, then physically isolating it on a different machine.
- Perhaps the best solution, however, is to materialise the data mart view, then physically isolate it on an appliance, i.e. a shared-nothing MPP (massively parallel processing) machine

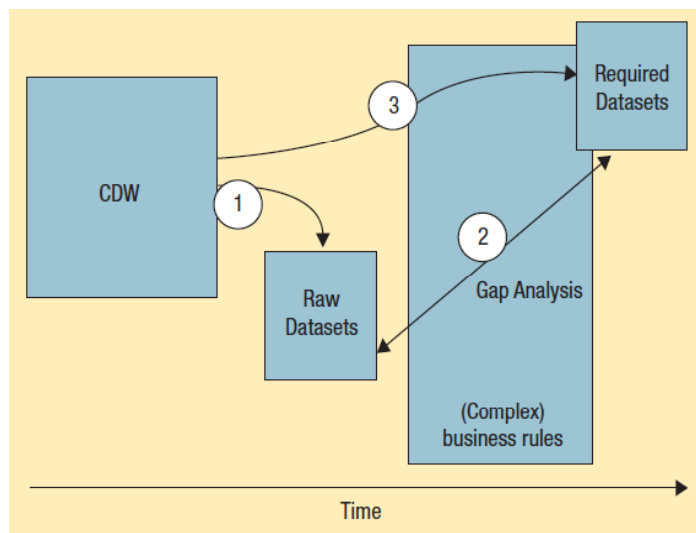
(often with a column-oriented database), which is capable of achieving unparalleled query performance in highly favourable management circumstances.

### The business dictates

End-users often find it difficult to define their requirements out of the blue. A lesson learned is that it is therefore often more prudent to set up a raw data mart, which can then be used as the starting point for the further refinement of the requirements (read: the business rules). Placing the data on-screen facilitates a huge efficiency drive in selecting the ultimate data mart design.

### Making raw data marts provides ample opportunities to determine the quality of the data at an early stage

The Data Vault-orientation of CDW and CDW+ offers fascinating opportunities to create such a raw data mart extremely rapidly (step 1, see Figure 6). Kimball dimensions consist of a group of HUBs and corresponding SATs, while Kimball facts comprise LINKs and corresponding SATs. And once again, largely standardised (and future generated) loading mechanisms can be used in the creation of such data marts. This raw data mart can subsequently be used to create the required data mart – together with the end-user – in an iterative manner (steps 2 and 3; see Figure 6). In practice, this has proven a much more successful method than attempting to establish and meet the information requirements in advance.



**Figure 6:** Raw datasets versus ultimately required datasets (Source:D.Linstedt)

Furthermore, making raw data marts offers ample opportunities to establish the quality of the data at an early stage. After all, this is the stage at which the user initially encounters the data. We would reiterate that the next generation EDW places the stage of defining poor data quality beyond the CDW. The end-user determines this quality on the basis of business rules; e.g., the

business rule 'if ORDER LINE no ARTICLESNO. then error'. It is standard practice in the next generation EDW to use such data quality lines to create error marts on the basis of which the end-user can report and act. This explicitly underlines the fact once more that data quality always remains the responsibility of the end-user (read: business), not only in terms of determining the data quality lines, but also reporting on the error-marts and responding to identified quality issues.

### **Conclusion**

We opened up the black box of the next generation EDW, then focused on the process beyond the Central Data Warehouse (CDW). As is the case for the process prior to the CDW, the process beyond the CDW is characterised by a considerable measure of standardisation, traceability, auditability, efficiency and performance. We therefore trust to have cleared up a few of the blind spots in the approach to the next generation EDW.

### **Literature**

1. R. Kimball & J. Caserta (2004), 'The Data Warehouse ETL Toolkit', Wiley.

**Lidwine van As** ([lidwine@grey-matter.nl](mailto:lidwine@grey-matter.nl)) is a Senior Consultant in the field of information architecture & management at Grey Matter.

**Ronald Damhof** ([ronald.damhof@prudenza.nl](mailto:ronald.damhof@prudenza.nl)) is a Senior Information Management Architect at Prudenza.