Virtual Data Warehousing

Implementation and Automation

Workshop with Roelant Vos

The Virtual Data Warehouse

Reducing barriers

In an inspiring presentation called 'Inventing on Principle', the US computer scientist Bret Victor discusses how ideas come into being and how sad it is when these do not manifest in reality, or are missed entirely, due to limitations in our (technical) environments.

To allow ideas to grow, creators need an immediate connection to what they are creating. This means that, as a creator, you need to be able to directly see what the effect of your changes are on what you are working on. But often this is not easily achieved.

If you take the example of coding, for instance, you first need to write code in a specific language and then compile the code to see the result of your work. The link to the code itself and the result can be hard to make if you don't compile, because just based on the code itself it is not always easy to image the outcome. And, it goes without saying that a certain amount of technical expertise is required for this.

How great would it be for any coder to directly see the impacts of their work in their application or website? How many ideas have been missed entirely because this direct connection was not available? How many thoughts never came up because of too much focus on the technical aspects?

Coding is a creative process, and modern software development techniques more and more allow for this kind of direct connection. You can directly see the effect of your code changes in your development environment. This means that, as a creator, you can see what feels right as an outcome while you are working. There is no need to read technical documentation, you immediately see what effects your changes have – like painting a canvas.

Data is also one of the elements that allow new ideas to fulfil their promise, and in many cases these ideas do not materialise because data cannot be adequately made available.

This is what the Virtual Data Warehouse as a concept and mindset intends to enable: providing a direct connection to data supporting exploration of any kind of exploration boosting creativity.

Thinking of Data Warehousing in terms of virtualisation is in essence about following the guiding principle to establish a direct connection to data.

It is about finding ways to seek simplification, to keep working on removing barriers to deliver data and information. It is about enabling ideas to flourish because data can be made available for any kind of discovery or assertion.

Providing a direct connection to data

Virtual Data Warehousing is the ability to present data for consumption directly from a raw data store by leveraging data warehouse loading patterns, information models and architecture. In a practical sense, a Virtual Data Warehouse is akin to deploying a set of views that mimic the ETL processes and physical (table) structures of the Data Warehouse schema – a schema that would otherwise be persisted.

In many Data Warehouse solutions, it is already considered a best practice to be able to 'virtualise' Data Marts in a similar way. The Virtual Data Warehouse takes this approach one step further by allowing the entire Data Warehouse to be refactored based on the raw transactions.

This ability requires a Persistent Historical Data Store, also known as a Persistent Staging Area where the data that is received by the Data Warehouse environment is stored as it has been received, at the lowest level and time-stamped. If data is retained this way, everything you do with your data can always be repeated at any time – deterministically.

In the best implementations, the Virtual Data Warehouse allows you to work at the level of simple metadata mappings, modelling and interpretation (business logic) while abstracting away the more technical details.

In some ways this can be seen as a paradox, because the path to make this simple can be sometimes very challenging indeed – also at a technical level.

Not the same as Data Virtualisation

A Virtual Data Warehouse is not the same as Data virtualisation. These two concepts are fundamentally different. Data virtualisation, by most definitions, is the provision of unified direct access to data across many (disparate) data stores.

It is a way to access and combine data without having to physically move the data across environments. Data virtualisation does not however focus on loading patterns and data architecture and modelling.

The Virtual Data Warehouse on the other hand is a flexible and manageable approach towards solving data integration and time variance topics using Data Warehouse concepts, essentially providing a defined schema-on-read.

The two concepts can work together. Data Virtualisation can be one of the ways how access to the (raw) data can be organised, and can server as a common access layer for consumption of information by the (virtual) Data Warehouse.

If we can generate everything...

The idea of the Virtual Data Warehouse was conceived as a result of working on improvements for generation of Data Warehouse loading processes (Extract, Transform and Load – or ETL). It is, in a way, an evolution in ETL generation thinking.

In the early days, by far the most ETL code was developed manually. Only relatively recently the generation of ETL processes has taken flight and at present many Data Warehouse solutions allow for most, if not all, of the required ETL to be generated using code generation techniques.

Data Vault, and similar hybrid approaches for data modelling, played a major role in this development because it provides elegant 'patterns' (templates) for various components of the Data Warehouse solution, including standard entity types and ETL processes. Data Vault allows for 'separation of concerns', and provides guidelines to isolate the main Data Warehouse functions into dedicated entity types, which can be loaded with standard patterns – and are generic enough to be automatically generated from metadata.

Combining Data Vault with a Persistent Historical Data Store provides additional functionality because it allows the designer to refactor parts of the Data Warehouse solution. In practice, this means that parts of the Data Warehouse can be truncated and reloaded in a deterministic way from the data available in the Persistent Historical Data Store.

At this point it is feasible to not only generate all ETL processes, but to generate the entire Data Warehouse itself. This

includes the data organised according to the desired Data Warehouse model - including the delivery in Marts.

Hybrid approaches for Data Warehousing are designed to be flexible, to be adaptable to accommodate changes in business use and interpretation. Working with data can be complex, and often the 'right' answer for the purpose is the result of a series of iterations where business Subject Matter Experts and data professionals collaborate.

This principle is a key foundation to design for flexibility in delivery (i.e. for virtualisation Marts) and underpins the extensibility of the hub-and-spoke (physical) data model. But it can also be applied to the design of the data model itself.

The same way that defining the correct business logic requires ongoing clarification before it can be considered fit-for-purpose, a data model is also typically the result of various iterations. This can include making changes to earlier modelling decisions, for instance regarding the selection of business keys and the (nature of the) relationships.

In other words, the Data Warehouse model itself is not always something you always can get right in one go. In fact, it can take a long time for a Data Warehouse model to stabilise, and in the current fast-paced environments this may even never be the case.

The Virtual Data Warehouse helps maintain both the mindset and capability for a data solution to keep evolving with the business, and to reduce technical debt on an ongoing basis. This mindset also enables some truly fascinating opportunities such as the ability to maintain version control of the data model, the metadata and their relationship - to be able to represent the entire Data Warehouse as it was at a certain point in time.

Similarly, it is possible to 'host' multiple versions of a Data Warehouse which can still be related to each other through the Persistent Historical Data Store. You can also leverage these concepts to gradually deploy changes in your information delivery (Marts, Semantic Layer), similar to best practices from the Data Integration world where changes in canonical (message) formats are slowly introduced to allow consuming parties to upgrade their adapters.

As a future outlook, the direction is to abstract away the remaining complexities to further allow a direct connection to be made between the data and its users. This is explained in detail in the paper titled 'the engine', work is ongoing on various Githubs to contribute to these ideas.

Adapting your data platform indefinitely

The Virtual Data Warehouse is enabled by virtue of combining the principles of ETL generation, hybrid data warehouse modelling concepts and a Persistent Historical Data Store. It is a way to create a more direct connection to the data because changes made in the metadata and models can be immediately represented in the information delivery.

Persisting of data in a more traditional Data Warehouse sense is always still an option, and may be required to deliver the intended performance. The deterministic nature of a Virtual Data Warehouse allows for dynamic switching between physical and virtual structured, depending on the requirements.

In many cases, this mix of physical and virtual objects in the Data Warehouses changes over time itself, when business focus changes. A good approach is to 'start virtual', and persist where required.

The better we can manage these environments and adapt them using metadata while abstracting away the underlying complexities, the stronger we can make the connection for our consumers to their data - and reap the benefits of their creativity.