Virtual Data Warehousing

Implementation and Automation

Workshop with Roelant Vos

Agenda & Overview

Contents - day 1

Day 1 is focused on the core theoretical concepts and solution architecture for the Data Warehouse solution. Hybrid modelling techniques such as Data Vault are more than just a technical solution, and it is important to gain a clear understanding of the modelling approach and how it differs from classical modelling techniques as this has impacts on fundamental architecture decisions.

The contents include a refresher on intra-systems integration, implementation process approaches, data analysis and modelling for business process alignment, as well as the entities and constructs used in Data Vault modelling.

Using the Data Vault modelling techniques, a data model is derived from a sample case software product ('source') in a collaborative exercise. This sample model is used to demonstrate and explain various architecture, implementation and automation concepts.

At the end of the day, it is expected that the participants have a sound understanding of the basic architecture building blocks for modelling, designing and implementing a hybrid Data Warehouse.

Forenoon session 9am - noon

Session 1	45 min	Introduction
Session 2	60 min	Model Driven Design overview
Break	15 min	Morning tea
Session 3	30 min	Solution Design & Architecture
Session 4	30 min	Staging concepts (part 1)

Lunch break noon - 1pm

Afternoon session 1pm - 5pm

Session 5	60 min	Staging concepts (part 2)
Session 6	60 min	Investigate the data model (collaborative modelling workshop)
Break	15 min	Afternoon tea
Session 7	15 min	Understanding the patterns and metadata requirements
Session 8	90 min	Development considerations & pattern explanation – Hubs

Contents – day 2

Day 2 covers the advanced topics of managing context data in Satellites and Link-Satellites. These are the areas where changes of information over time is captured.

In addition to this, the conceptual and technical implications of parallel loading and managing of consistency of data are discussed and an introduction on the delivery of information is made.

Forenoon session 9am - noon

Session 1	60 min	Development & pattern considerations – Links
Session 2	45 min	Development & pattern considerations – Sats / Link-Sats part 1
Break	15 min	Morning tea
Session 3	60 min	Development & pattern considerations – Sats / Link-Sats part 2

Lunch break noon - 1pm

Afternoon session 1pm - 5pm

Session 4	30 min	Technical considerations
Session 5	30 min	Workflows and parallelism
Session 6	60 min	Control framework
Break	15 min	Afternoon tea
Session 7	30 min	Metadata model wrap-up
Session 8	30 min	Presentation Layer introduction
Session 8	60 min	Time-variance concepts – part 1

Contents – day 3

Day 3 covers more advanced patterns of Data Vault implementation. It also covers the transition from Data Vault to delivery layers ('Marts'), the application of business logic and technical considerations and solutions.

Forenoon session 9am - noon

Session 1	60 min	Time-variance concepts – part 2
Session 2	45 min	Point-in-Time and Bridge Tables
Break	15 min	Morning tea
Session 3	45 min	Derived tables
Session 4	15 min	Dimensions and Facts – part 1

Lunch break noon - 1pm

Afternoon session 1pm - 5pm

Session 5	120 min	Dimensions and Facts– part 2
Break	15 min	Afternoon tea
Session 6	60 min	Flexibility in development
Session 7	60 min	Wrap-up

Model Driven Design overview

The direction to move away from manually creating data integration logic, towards a more pattern-based approach directed by the information model, is called Model Driven Design.

Fundamentally this is about cultivating a mindset of flexibility in design by leveraging modular patterns and supporting technologies. This session provides an introduction of the thinking behind ETL generation and automation in general.

This session focuses on the:

- Required components for Model Driven Design
- Styles of ETL generation
- Guiding principles and requirements
- Family of hybrid modelling techniques
- Overview of the Data Vault concepts and core entity types
- Overview of Data Vault implementation patterns

Solution Design & Architecture

This session takes a step back from 'just' the Data Vault physical model and looks at the overall design. Now that there is a foundational understanding of the Data Vault modelling approach and its intent, the end-to-end architecture can be explained. This will provide a reference point for the advanced topics.

The topics following topics are covered:

- Overview of the layers and areas in a Data Vault architecture
- Architecture options & considerations, especially related to the interactions between difference concepts. What needs to be done where, and what are the impacts of certain design decisions?
- Combining multiple platforms and technologies
- Specifics and requirements of each area in the architecture
- 'Hard' and 'soft' business rules

Staging Concepts

A Data Vault system often explained in terms of Hubs, Links and Satellites. However, the way the source (Online Transactional Processing – OLTP) systems are interfaced with have far-reaching implications on the way the Data Vault system behaves as a whole.

For this reason, configuring a Staging Area is one of the most complex areas in a Data Vault architecture. Contents of this session include:

- Understanding different data staging approaches (patterns)
- Implications of date/time stamping, where and how to capture the Load Date / Time Stamp – and other dates
- Key requirements for a Staging Layer
- Persistent Staging Area (PSA) considerations
- Preparing to be near-real-time
- Supporting parallel processing
- Change Data Capture (CDC) and the impacts on Data Vault

Investigate the data model (modelling workshop)

How do you design a Data Vault model? This session explains the steps involved to define the target model. A Data Vault model is a representation of the business architecture and is a generic, central, model on to available information is mapped.

Using ETL generation and automation concepts is done using an end-to-end use-case based on the 'SaveMore' sample data set.

This session covers:

- The steps to model a Data Vault
- An investigation of a sample source system ('SaveMore' case)
- Discussion of the target Data Vault model

Understanding the patterns and metadata requirements

With the source and target models available, focus can be placed on the mapping metadata itself. What metadata is required and where is it located? Regardless whether you are developing a solution yourself or use Data Warehouse Automation software the metadata is the same. This session explains in detail what needs to be provided and creates an understanding how this metadata fits into the various ETL patterns.

The focus of this session is to provide the following:

- Overview of the required patterns
- Overview of the required metadata

Development & pattern considerations – Hubs

The list of Hub entities is the single most defining aspect of a Data Vault solution, and heavily influence subsequent design decisions. They are also the most straight-forward patterns to implement from a development perspective.

Given the critical nature of the role Hubs play it is important to ensure the implementation of Hub ETL 'just works'.

To achieve this the following considerations are discussed:

- Hub pattern, structure and implementation
- Parallism
- Technical types of Business Keys
- Hash keys, including generation across different platforms and collision
- Hub metadata & generation

Development & pattern considerations - Links

You could look at Hubs as if they were the 'joints' of the Data Vault model. In that case, the Links would be the 'bones'. Links manage relationships between business concepts and govern the granularity and Unit of Work (UoW) within the Data Vault. They require design and implementation choices specifically geared towards this.

This session covers the concepts that are specifically relevant to Link implementation:

- Link pattern, structure and implementation
- Regular
- Recursive and clustering mechanisms in Links (Transactional, Same-As and Hierarchical)
- Degenerate attributes in Links
- Link metadata & generation

Development & pattern considerations - Satellites and Link-Satellites

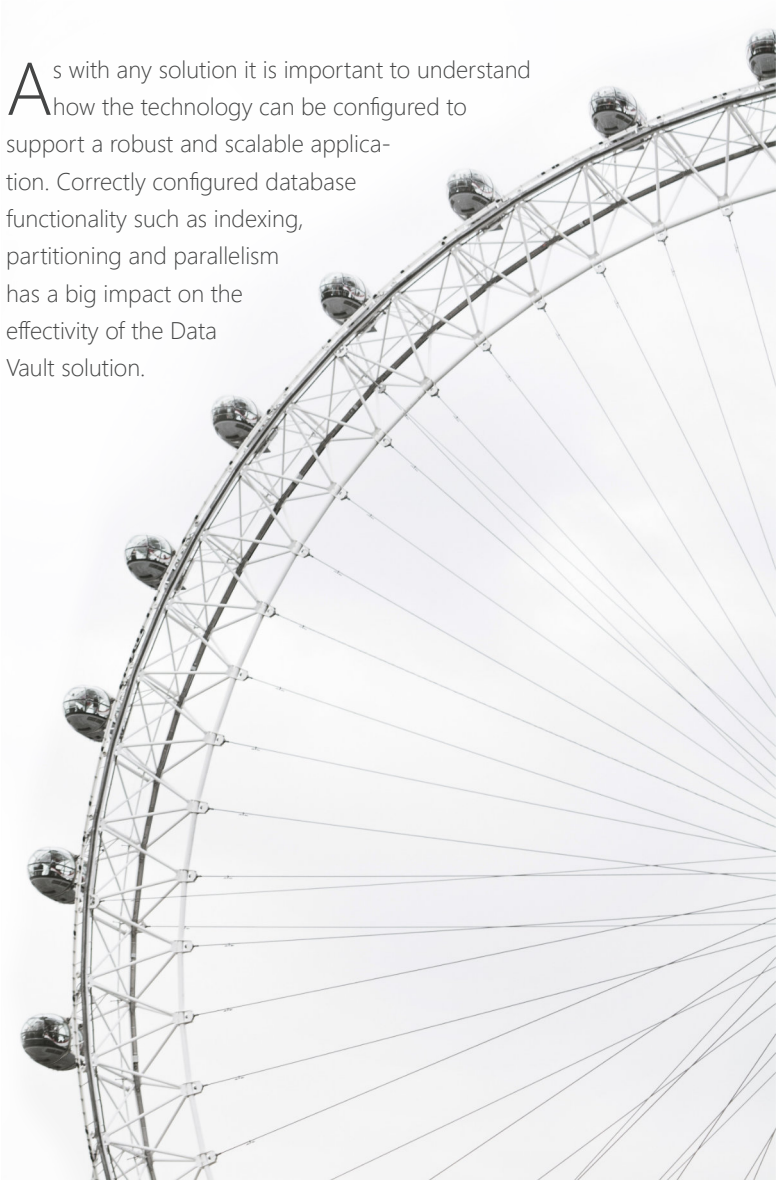
The Satellite and Link-Satellite entities is where Data Vault manages 'time variance': this is where the data changes are captured in time. Satellites provide the context for business concepts (Hubs) and relationships (Links).

The content of this session is geared towards defining a flexible approach for 'tracking changes' and covers concepts such as:

- Satellite pattern, structure and implementation (including Link-Satellites)
- Row (record) condensing, either considered independent or as part of CDC
- Change merging
- Attribute scope
- End-dating
- Zero records
- Multi-Active (multi-variant) Satellite approaches

Technical considerations

As with any solution it is important to understand how the technology can be configured to support a robust and scalable application. Correctly configured database functionality such as indexing, partitioning and parallelism has a big impact on the effectivity of the Data Vault solution.



This session will discuss considerations related to technologies such as:

- Compression
- Partitioning
- Indexing
- Filtering
- Referential Integrity
- Error handling

Workflows and parallelism

The introduction of Data Vault 2.0 has provided various ways parallel processing can be implemented. It is the intent to load data as soon as it is available, and this session explains what this means from a design and implementation perspective.

This session will discuss considerations related to technologies such as:

- Parallelism, considerations and impacts on the solution design
- Redundancy
- Referential Integrity in a parallel loading environment

Control framework

Most organisations have an ETL control framework in place, and every Data Warehouse Automation platform includes one. This session explains why an ETL control framework is essential for a Data Vault delivery, not only simply for auditing but as an integral part to ensure consistency of delivered information.

The following topics are covered in this session:

- Transaction isolation at application level – how can a Data Vault guarantee consistency?
- Examples of logical grouping for execution of load processes (options and considerations)
- Rollback and recovery
- Continuous parallel execution. Near real-time loading, considerations and impacts on the solution design

Metadata mapping wrap-up: finalising the core design

This session focuses on ensuring the design and mapping decisions are adequately implemented in the ETL generation and management environment.

The scope of this session covers:

- Updating the metadata with the specific Hub and Link design decisions
- Generating and running the target Data Vault solution (demonstration)

Time-variance concepts

Combining multiple time-variant tables into a single delivery is a key requirement to deliver data to marts. Handling potentially multiple timelines into a single delivery is the main focus point in this session. From a technical point of view the Point-In-Time (PIT) entity is explained in detail. This can provide a method to manage performance in delivering the Presentation Layer. By pushing down the complexities of managing time-variant data into a helper table the performance implications of certain design decisions can be balanced out.

The session covers the following topics:

- Time-variance concepts – 'date math'
- How to join Data Vault entities
- Timing issues and how to resolve these
- PIT pattern, structure and implementation
- Stacked versus continuous PIT approaches (temporal considerations)
- PIT concepts for data virtualisation
- Impacts of load order on PIT information

Bridge tables

Bridge tables provide similar functionality as PIT tables, but are specifically meant to support faster resolution of database joins.

The following is covered in this session:

- Bridge pattern, structure and implementation
- When to use Bridge tables, and when not to

Derived tables

The archetype entities can be reused to support deriving information (by applying business logic) to provide alternative perspectives of the data available in the Data Vault.

Information is not (necessarily) available in the source systems in the way it is ideally made available in the Data Vault. This means that it cannot be loaded directly and requires alternative patterns to manage this information in a Data Vault environment.

This session discusses the following items:

- Applying business logic, the 'front-room' versus the 'back-room'
- Technical options and considerations for implementation of transformations
- Driving Keys
- Impacts of date/time selection on derived tables

Dimensions and Facts

In a Data Vault approach, the Presentation Layer is (very broadly) defined as anything that is fit-for-purpose. This is on purpose, because the intent of Data Vault is to support the organisation in the clarification of requirements over time – as opposed to the requirement of having everything available upfront.

The default approach in Data Vault is to define 'Raw' Data Marts and iteratively adapt these to 'Information' Marts. Business logic is added iteratively in collaboration with business Subject Matter Experts (SMEs).

The Presentation Layer covers the following topics:

- Dimension pattern(s), structure and implementation
- Fact table pattern(s), structure and implementation
- Types of history
- Handling timing issues
- Switching time perspectives – how to match the business expectations?

Flexibility in development

With all the Data Vault patterns defined and understood, and with all the metadata available and documented there are many ways to implement even further changes to increase time-to-value.

The end-to-end Data Vault solution, when correctly designed, allows for various ways of scaling-up and scaling-out including cross platform deployments and mixtures of traditional RDBMS and service-based technologies.

This session provides an overview of what approaches are being used and further developed in the industry, and a view to the future. The Data Vault solution is not meant as a one-off development activity but a long-term solution that is intended to grow and adapt to the changing business needs.

The following topics are covered:

- Changing technical architectures – how to scale out?
- Using higher level of modelling architecture patterns to drive a Data Vault (physical model) approach
- Fact-oriented and domain specific language approaches for Data Vault
- State machines and containers for deployment and upscaling