Data Warehouse Design Patterns

Practical training with ready-to-use patterns to architect, implement and fully automate your data solution

Workshop with Roelant Vos

Contents – day 1

Day 1 is focused on the core concepts and architecture for the Data Warehouse solution. Hybrid modelling techniques are more than only a technical solution, and it is important to gain a clear understanding of the underlying ideas and how they differ from classical modelling techniques. These differences have impact on fundamental architecture decisions.

The content includes a refresher on intra-systems integration, implementation process approaches, data behaviour and modelling for business process alignment, as well as the entities and constructs used in hybrid data modelling.

A sample model is investigated and used to demonstrate and explain various architecture, implementation, code generation and process automation concepts.

At the end of the day, participants will have a sound understanding of the essential architecture building blocks ('patterns') for modelling, designing and implementing a hybrid Data Warehouse.

Forenoon sessions 9am – noon

Session 1	45 min	Introduction
Session 2	60 min	Model Driven Design overview
Break	15 min	Morning tea (~10:45-11:00)
Session 3	60 min	Solution Design & Architecture

Lunch break noon – 1pm (60 min lunch)

Afternoon sessions 1pm – 5pm

Session 4	90 min	Staging concepts
Session 5	30 min	Investigate the data model
Break	15 min	Afternoon tea (~15:00-15:15)
Session 6	15 min	Understanding the patterns and metadata requirements
Session 7	90 min	Development considerations & pattern explanation – Core Business Concepts

Contents – day 2

Day 2 covers the advanced topics of managing contextual data and the complexities around time-variance. These are the areas where changes in data over time are captured.

In addition to this, the way to manage relationships between data elements (Core Business Concepts) using Natural Business Relationships is also covered.

At this stage, the main modelling archetypes are available and the focus shifts to delivery with details on various orchestration approaches and how to embed these in a process control framework.

The control framework is an essential component to guarantee reliable information delivery and ties in to the conceptual and technical implications of parallel loading and managing of consistency of data.

The collection of patterns can then be combined into a managed delivery using process automation – as part of the implementation of release management and 'DevOps'.

The day finishes with an introduction on the delivery of information for consumption.

Forenoon sessions 9am – noon

Session 1	60 min	Development & pattern considerations – Natural Business Relationships
Session 2	45 min	Development & pattern considerations – Contextual data part 1
Break	15 min	Morning tea (~10:45-11:00)
Session 3	60 min	Development & pattern considerations – Contextual data part 2

Lunch break noon – 1pm (60 min lunch)

Afternoon sessions 1pm – 5pm

Session 4	30 min	Technical considerations
Session 5	30 min	Workflows and parallelism
Session 6	60 min	Control framework & orchestration
Break	15 min	Afternoon tea (~15:00-15:15)
Session 7	60 min	Release management and process automation – auto-refactoring and deployment
Session 8	15 min	Metadata model wrap-up
Sessiion 9	30 min	Presentation Layer / Information Delivery introduction

Contents – day 3

Day 3 is focused on the delivery of information for end-user consumption. This means investigating the transition from the core Data Warehouse layer to various forms of delivery layers ('Marts').

As part of the information delivery, the application of business logic and technical considerations are covered.

Forenoon sessions 9am – noon

Session 1	60 min	Time-variance concepts – part 1
Session 2	45 min	Time-variance concepts – part 2
Break	15 min	Morning tea (~10:45-11:00)
Session 3	60 min	Derived tables (including Point-in-Time calculations)

Lunch break noon – 1pm (60 min lunch)

Afternoon sessions 1pm – 5pm

Session 4	120 min	Dimensions and Facts
Break	15 min	Afternoon tea (~15:00-15:15)
Session 5	60 min	Applying business logic
Session 6	60 min	Wrap-up

Overview of topics

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 core implementation patterns

Solution Design & Architecture

This session takes a step back and looks at the overall design. Now that there is a foundational understanding of the 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 Warehouse 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 in a design
- Specifics and requirements of each area in the architecture
- Separation of concerns

Staging Concepts

Getting the data into the Data Warehouse environment is one of the most complex areas in the Data Warehouse architecture. This session covers the fundamental concepts that need to be included in any design and explores the impacts on the subsequent layers in the 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)

Investigate the data model

How do you design a Data Warehouse model? This session explains the steps involved to define the target model. A data model is a representation of the business. It 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 core Data Warehouse layer
- An investigation of a sample source system ('SaveMore' case)
- Discussion of the target data 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 – Core Business Concepts

The Core Business Concept entities are the single most defining aspect of a Data Warehouse 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 Core Business Concepts play it is important to ensure the implementation of ETL 'just works'.

To achieve this the following considerations are discussed:

- Pattern, structure and implementation
- Parallism
- Technical types of Business Keys
- Key distribution approaches
- Metadata & generation

Development & pattern considerations – Natural Business Relationships

You could look at Core Business Concepts as if they were the 'joints' of the model. In that case, the Natural Business Relationships would be the 'bones'.

Natural Business Relationships manage relationships between business concepts and govern the granularity and Unit of Work within the model. They require design and implementation choices specifically geared towards this.

This session covers the concepts that are specifically relevant to Natural Business Relationship implementation:

- Pattern, structure and implementation
- Recursive and clustering mechanisms
- Degenerate attributes
- Metadata & generation

Development & pattern considerations – Contextual Data

The contextual entities is where 'time variance' is managed, where the data changes are captured in time. These entities provide the context ('describe') for Core Business Concepts and Natural Business Relationships.

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

- Pattern, structure and implementation
- Row (record) condensing, either considered independent or as part of CDC
- Change merging
- Attribute scope
- End-dating
- Zero records
- Multi-Active (multi-variant) 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 Warehouse solution.

This session will discuss considerations related to technologies such as:

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

Workflows and parallelism

There are various ways how 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 & orchestration

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 reliable data 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 Warehouse 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

Release Management and process automation

This session focuses on how to organise and automate the workflow of activities to physically create the data integration processes and deploy them to a target environment.

Adequate release management has a significant impact on system reliability and the ability to quickly refactor or even virtualise. In this session the various required components to develop the Data Warehouse are automatically combined into a delivery.

The scope of this session covers:

- The format and integration of metadata
- Automation of ETL generation
- Automating a process workflow

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 time-variant 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

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 Warehouse

Information is not (necessarily) available in the source systems in the way it is ideally made available in the Data Warehouse. This means that it cannot be loaded directly and requires alternative patterns to be able to process the data.

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

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

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?

Applying Business Logic

Correctly applying the patterns in a clear architecture that 'separates concerns' goes a long way in simplifying working with data. With this in place, you have a flexible solution that can automatically be refactored or adjusted in workshops with Subject Matter Experts – while retaining auditability.

In essence, business logic is layered 'on top' of this solution to represent the adjustments required to make the data fit for purpose. This is the implementation of specific requirements provided by the consumers of the information.

The following topics are covered:

- Overview of locations to 'insert' business logic
- How to record and manage business logic
- Handling competing interpretations of data