MODULE 3

Data Modelling for Decision Support

Designing Data Structures That Serve Decisions, Not Storage

Why Module 3 Exists

Up to now, we have defined:

- **The decision** (Module 1)
- **How success is judged** via KPIs (Module 2)

Module 3 answers the next critical question:

"How must the data be structured so that these KPIs are reliable, explainable, and fast to compute?"

This is where many analysts fail—by analyzing *whatever shape the data arrives in*.
Professionals **reshape data to reflect the business problem**.

WHAT YOU WILL PRODUCE IN MODULE 3

Portfolio Artifact:
🧩 **Decision-Support Data Model (Logical + Physical)**

By the end of this module, you will have:

- A clear analytical data model
- Defined grain (level of detail)
- Fact vs dimension separation
- A schema that directly supports your KPIs

This artifact is critical for credibility in analytics and engineering-adjacent roles.

PART 1 — CORE CONCEPTS (EXPLAINED)

1. Transactional Data ≠ Analytical Data

MT5 trade history is **transactional**:

- Optimized for execution
- Event-by-event

- Not decision-friendly

Decision analytics requires **analytical data**:

- Aggregated
- Contextualized
- Consistent across time

We do not "query" transactional data repeatedly.
We **model** it once and reuse it safely.

---

2. Grain: The Most Important Concept in Data Modelling

Grain answers:

What does one row represent?

Examples:

- One row per trade
- One row per trading session
- One row per trading day
- One row per EA configuration

You **must define grain before creating any table**.

Poor grain decisions lead to:

- Duplicate counts
- Incorrect KPIs
- Misleading drawdown calculations

---

3. Fact Tables vs Dimension Tables (Lightweight, Practical)

We will use a **simplified star schema**, not academic overkill.

- **Fact tables** store measurements (numbers)
- **Dimension tables** store context (labels, categories)

Your KPIs will live on **fact tables**.

PART 2 — APPLYING THIS TO YOUR MT5 EA CONTEXT

Your KPI Requirements (Derived from Module 2)

Your data model must support:

- Single-trade risk exposure
- Floating & max drawdown
- Profit consistency (daily)
- Recovery factor
- Trade duration drift
- Session-based diagnostics
- Symbol-level comparison (Step Index vs others)

This immediately tells us:
👉 **One table is not enough**

PART 3 — TARGET DATA MODEL (LOGICAL)

Core Fact Tables (Recommended)

1️⃣ Fact_Trades

Grain: One row per executed trade

Purpose:
Supports:

- Risk exposure
- Trade duration
- Per-trade P/L
- Scalping behaviour

Key Fields

- TradeID (PK)
- EntryTime
- ExitTime
- Symbol
- Direction (Buy/Sell)
- LotSize
- EntryPrice
- ExitPrice
- StopLoss
- TakeProfit

- Profit
- AccountEquityAtEntry
- HoldingTimeSeconds
- SessionID (FK)

---

2 Fact_Sessions

Grain: One row per trading session per day

Purpose:
Supports:

- Session drawdown
- Time-of-day analysis
- Behavioural clustering

Key Fields

- SessionID (PK)
- TradeDate
- SessionName (Asian / London / NY / Synthetic)
- SessionStartEquity
- SessionEndEquity
- SessionMaxDrawdown
- TradesCount
- NetSessionProfit

---

3 Fact_Daily_Performance

Grain: One row per trading day

Purpose:
Supports:

- Profit consistency
- Recovery factor
- KPI rollups

Key Fields

- TradeDate (PK)
- StartingBalance
- EndingBalance
- NetDailyProfit
- MaxDailyDrawdown
- TradesCount

- PositiveDayFlag (Y/N)

Dimension Tables

Dim_Symbol

- SymbolID (PK)
- SymbolName
- InstrumentType (Step / Volatility / Other)
- ContractSize
- TickValue

Dim_Session

- SessionID (PK)
- SessionName
- StartTime
- EndTime

PART 4 — WHY THIS MODEL WORKS (TEACHING MOMENT)

KPI Traceability Example

KPI: Maximum Floating Drawdown

- Source: Fact_Sessions.SessionMaxDrawdown

KPI: Trade Duration Drift

- Source: Fact_Trades.HoldingTimeSeconds

KPI: Positive Day Ratio

- Source: Fact_Daily_Performance.PositiveDayFlag

Every KPI has:

- A single source of truth
- A defined grain
- No ambiguous joins

This is **decision-safe Modelling**.

PART 5 — MODULE 3 EXERCISE

Your Data Modelling Task

You will now produce **Module 3's portfolio artifact**.

Deliverable Requirements

1. Logical Data Model (Mandatory)

Provide:

- List of fact tables
- Grain for each
- Key fields per table
- Relationship explanation

You may present this as:

- Structured text, or
- A simple diagram (ASCII or image), or
- A table-by-table description

---

2. KPI-to-Table Mapping (Mandatory)

For each KPI from Module 2, state:

- Which table(s) it uses
- Why that grain is appropriate

Example:

"Single-Trade Risk Exposure is derived from Fact_Trades at trade-level grain to avoid aggregation distortion."

---

3. Design Justification (Short Narrative)

Answer:

- Why multiple fact tables are required
- Why transactional MT5 data alone is insufficient
- What analytical errors this model prevents

PART 6 — QUALITY BAR

Your model must:

- Explicitly define grain
- Avoid duplicated measures
- Support all KPIs without workaround logic
- Be explainable to a non-technical stakeholder

If you cannot explain *why* a table exists, it does not belong.

YOUR NEXT ACTION

Produce **Module 3 — Data Model for Decision Support (v1.0)** and submit it here.

I will:

- Review it as a data architect would
- Score it using a rubric
- Stress-test it against your KPIs
- Refine it until it is portfolio-grade

You are now entering the **engineering–analytics crossover zone**. This module is where that becomes visible.