TessQ Behavioural Telemetry — 1M Tick Dataset

The TessQ dataset contains **1,000,000 sequential ticks** of high-resolution behavioural telemetry generated by the TessQ engine — a synthetic, multi-organ behavioural system designed to encode microstructure dynamics into a stable, machine-learning-ready feature set.

TessQ transforms live market movements into a **behavioural manifold**: a structured, geometric representation of drift, sigma, tension, pressure, wave dynamics, and regime identity. All fields are **fully derived, entropy-encoded, and non-reversible**, ensuring complete commercial safety and compliance.

Each dataset is a **self-contained session** with:

- deterministic feature generation
- consistent schema across all releases
- no external dependencies
- stable organ behaviour
- interpretable geometric structure

This makes TessQ suitable for:

- machine learning
- quantitative research
- anomaly detection
- microstructure modelling
- synthetic intelligence training
- regime classification
- signal discovery

The dataset includes **multi-organ behavioural telemetry**, including:

- predictive envelope fields (drift, sigma, tension, curvature, energy)
- heartbeat latency and timing
- decision-kernel sentinel signals
- wave dynamics (length, amplitude, slope)
- microstructure edge alignment
- entropy-driven branching logic
- 4D manifold coordinates
- tesseract vertex mapping
- macro-state regime identity

These components combine into a **coherent behavioural space** that captures structural transitions, micro-events, and regime shifts with exceptional clarity.

Synthetic Expression of Real Behaviour

How Tess-QDE4D Derives Real Microstructure but Outputs Synthetic Telemetry

Tess-QDE4D is a behavioural engine that reacts to **real market micro-movements**, but expresses those movements through a **synthetic behavioural manifold**. This hybrid design provides the fidelity of real behavioural dynamics while maintaining the safety, compliance, and interpretability of synthetic telemetry.

TessQ does **not** expose raw market data. Instead, she transforms real microstructure into **derived, entropy-encoded behavioural fields** that cannot be reversed.

This section explains how that transformation works.

1. Real Microstructure as Input

Tess-QDE4D listens to real market behaviour at the micro-level, including:

- directional nudges
- volatility pulses
- compression/expansion cycles
- microstructure pressure
- drift/tension dynamics

These signals form the **behavioural stimulus** that the engine responds to.

However:

- no raw prices
- no timestamps
- no order book data
- no trades
- no exchange feed content are ever exposed in the dataset.

The real feed is **felt**, not shown.

2. Behavioural Interpretation Through Multi-Organ Processing

Once TessQ receives real micro-movements, she interprets them through seven behavioural organs:

- Prism — drift, sigma, tension, curvature, energy
- Heartbeat — timing, latency, rhythm
- Sentinel — pressure, consensus, micro-signals
- Wave — oscillation, amplitude, slope
- Edge — microstructure alignment

- QDE — entropy compression & branching
- Tesseract — 4D geometry & regime identity

Each organ transforms real behaviour into a **synthetic behavioural representation**.

None of these fields exist in the raw feed. They are **created** by the engine.

3. Synthetic Behavioural Manifold (4D State-Space)

Real microstructure behaviour is encoded into a **four-dimensional manifold**, represented by:

- t4d_x
- t4d_y
- t4d_z
- t4d_w

These coordinates do not correspond to price or time. They represent TessQ's **internal behavioural geometry**.

This manifold is synthetic, but the motion inside it is driven by real behavioural dynamics.

4. Tesseract Projection (16 Behavioural Attractors)

The 4D manifold is projected into a **tesseract scaffold** with 16 attractor vertices.

Each vertex corresponds to a **behavioural macro-state**, such as:

- Neutral Drift
- Bear Tense Stable
- Bull Calm Expansion
- Volatile Break
- Compression Phase
- Expansion Phase

These regimes are synthetic classifications derived from real behaviour.

5. Entropy-Encoded Telemetry (Non-Reversible Output)

Even though TessQ reacts to real behaviour, all outputs are:

- synthetic
- derived
- entropy-encoded
- non-reversible
- safe for commercial use

This ensures:

- no raw market data exposure
- no reconstruction risk
- no compliance issues
- no proprietary feed leakage

The dataset expresses **behaviour**, not price.

6. Real Behaviour → Synthetic Expression Pipeline

Here is the clean conceptual pipeline:

1. **Real micro-movement occurs**
2. TessQ senses it
3. Each organ interprets it
4. Behaviour is encoded into geometry
5. Geometry collapses into a regime
6. Telemetry is entropy-encoded
7. Synthetic behavioural fields are output

This pipeline preserves the **truth of behaviour** while removing the **risk of raw data**.

7. Why This Hybrid Model Is Valuable

This design gives you:

- real behavioural fidelity
- synthetic safety
- geometric interpretability
- regime stability
- multi-organ richness
- high-resolution telemetry
- ML-ready numeric fields

It is the best of both worlds:

Real behaviour, synthetic expression.

□ Summary

Tess-QDE4D derives her internal state from real market micro-movements, but expresses that behaviour through a synthetic, entropy-encoded behavioural manifold. This makes the dataset:

- behaviourally accurate
- geometrically structured
- regime-aware
- safe
- non-reversible

- ideal for ML and quant research

It is a **synthetic expression of real behaviour**, not a simulation and not a raw feed.

TessQ Engine — Organ Overview

Multi-Organ Behavioural Architecture

The TessQ engine processes market microstructure through a set of interacting behavioural organs. Each organ contributes a different perspective on market behaviour, and together they form a coherent, geometric behavioural manifold.

These organs do **not** represent raw market data. They are **synthetic behavioural encodings**, derived from TessQ's internal state and entropy-encoded for safety.

The following overview explains each organ in clear, accessible terms while preserving the proprietary mechanics of the engine.

1. Prism Organ

Predictive Envelope: Drift, Sigma, Tension, Curvature, Energy

The Prism Organ is the core behavioural estimator. It transforms micro-movements into a predictive envelope that describes short-horizon behavioural pressure.

It produces:

- **pr_drift** — directional behavioural drift
- **pr_sigma** — volatility-derived sigma estimate
- **pr_tension** — behavioural stress or compression
- **pr_curvature** — bending of the behavioural envelope
- **pr_energy** — internal energy state
- **pr_pred** — primary behavioural prediction
- **pr_latDelta / pr_latSigma / pr_latPressure** — lateral behavioural components
- **pr_regression** — regression-based behavioural estimate

Layman's explanation

Think of the Prism Organ as TessQ's "weather system." It describes whether behaviour is:

- drifting
- compressing
- expanding
- bending
- stabilising
- becoming energetic

These fields are widely used in:

- regime detection
- ML feature engineering
- volatility modelling

- anomaly detection

2. Heartbeat Organ

Timing, Latency, Behavioural Pulse

The Heartbeat Organ tracks TessQ's internal rhythm — how quickly the engine is processing behavioural changes and how much latency is present in the behavioural field.

It produces:

- **hb_tick_ms** — time between behavioural ticks
- **hb_latency_ms** — behavioural latency
- **hb_field** — heartbeat field strength

Layman's explanation

This organ acts like a "pulse monitor." It shows how fast TessQ is reacting and whether the behavioural field is tightening or relaxing.

Useful for:

- anomaly detection
- micro-event identification
- behavioural timing analysis

3. Sentinel Organ

Pressure, Consensus, Directional Micro-Signals

The Sentinel Organ provides lightweight behavioural micro-signals derived from internal pressure, consensus, and directional cues.

It produces:

- **dk_pressure**
- **dk_consensus**
- **dk_power**

Layman's explanation

These are TessQ's "micro-sensors." They help identify subtle behavioural shifts that often precede larger transitions.

Useful for:

- early regime hints
- microstructure pattern recognition

- ML feature enrichment

(This organ is described safely — no proprietary mechanics exposed.)

4. Wave Organ

Amplitude, Slope, Length — Behavioural Waveform

The Wave Organ encodes TessQ's internal waveform — how behaviour oscillates, expands, or contracts over time.

It produces:

- **wave_len** — behavioural wave length
- **wave_amp_bps** — amplitude
- **wave_slope_bps_s** — slope

Layman's explanation

This organ describes TessQ's "breathing pattern." It shows whether behaviour is rising, falling, flattening, or accelerating.

Useful for:

- pattern recognition
- signal decomposition
- ML time-series modelling

5. Edge Organ

Microstructure Alignment & Forward Cues

The Edge Organ identifies short-horizon structural edges and alignment cues within the behavioural field.

It produces:

- **edge_05ms** — micro-edge detection
- **edge_futureAligned** — forward alignment
- **edge_futureLead** — forward lead indicator

Layman's explanation

This organ acts like a "microstructure compass." It helps identify when behaviour is aligning with short-term structural edges.

Useful for:

- regime transitions
- micro-event detection
- ML feature refinement

(Again, safely described — no proprietary mechanics exposed.)

6. QDE Organ (Entropy Branching)

Entropy Compression, Branching, Best-Path Selection

The QDE Organ tracks entropy compression and branching behaviour inside the engine.

It produces:

- **qde_branch** — branch index
- **qde_best —** best behavioural path
- **qde_minEntropy** — minimum entropy state

Layman's explanation

This organ shows how "certain" or "uncertain" TessQ feels. Lower entropy often corresponds to clearer behavioural structure.

Useful for:

- anomaly detection
- regime stability analysis
- ML confidence modelling

7. Tesseract Organ (Behavioural Geometry)

4D Manifold, Vertex Mapping, Regime Identity

The Tesseract Organ is the geometric core of TessQ. It projects behavioural telemetry into a **4-dimensional manifold** and maps it onto a **16-vertex tesseract scaffold**.

It produces:

- **t4d_x, t4d_y, t4d_z, t4d_w** — manifold coordinates
- **t4d_vertex** — nearest attractor vertex
- **t4d_regime** — macro-state regime ID
- **t4d_regime_name —** human-readable regime name

Layman's explanation

This organ is TessQ’s “map of the world.” It shows which behavioural region the engine is currently in — similar to a weather zone or climate regime.

Useful for:

- regime classification
- structural transitions
- ML state-space modelling
- behavioural clustering

□ SCHEMA TABLE (Expanded & Updated)

Field-by-Field Descriptions with Behavioural Meaning & Usage Guidance

Below is the updated schema table for the TessQ behavioural telemetry dataset.

Core Session Fields

Field	Type	Description
ts	INTEGER	Sequential timestamp counter for the session. Used for ordering and time-series alignment.
tk	INTEGER	Tick index within the session. Identical to `ts` unless custom sampling is applied.
px	FLOAT	Synthetic price snapshot at tick. Not raw market data; used for behavioural alignment.
im	FLOAT	Immediate micro-movement indicator. Captures tiny directional nudges in behaviour.

Prism Organ — Predictive Envelope

Field	Type	Description
pr_drift	FLOAT	Directional behavioural drift. Indicates short-horizon behavioural lean. Useful for ML trend features.
pr_sigma	FLOAT	Volatility-derived sigma estimate. Higher values indicate behavioural expansion.
pr_tension	FLOAT	Behavioural stress or compression. Negative values often indicate tightening or strain.
pr_pred	FLOAT	Primary behavioural prediction output from the Prism Organ.
pr_latDelta	FLOAT	Lateral behavioural delta. Captures sideways movement in the behavioural field.
pr_latSigma	FLOAT	Lateral sigma estimate. Useful for detecting sideways volatility.
pr_latPressure	FLOAT	Lateral pressure metric. Indicates sideways behavioural force.
pr_regression	FLOAT	Regression-based behavioural estimate. Often smoother than `pr_pred`.
pr_curvature	FLOAT	Curvature of the behavioural envelope. High curvature indicates bending or inflection.
pr_energy	FLOAT	Internal energy state. Higher values indicate more active behavioural conditions.

Heartbeat Organ — Timing & Latency

Field	Type	Description
hb_tick_ms	FLOAT	Time between behavioural ticks (ms). Useful for anomaly detection and timing analysis.
hb_latency_ms	FLOAT	Behavioural latency measurement. Indicates how quickly the behavioural field is responding.
hb_field	FLOAT	Heartbeat field strength. Represents internal rhythmic stability.

Sentinel Organ — Pressure & Consensus

Field	Type	Description
dk_pressure	FLOAT	Behavioural pressure signal. Useful for identifying subtle shifts before larger transitions.
dk_consensus	FLOAT	Consensus strength within the behavioural field. Higher values indicate clearer structure.
dk_power	FLOAT	Combined micro-signal strength. Often used as a confidence or stability indicator.

Position & PnL Fields

Field	Type	Description
side	STRING	Position side (LONG / SHORT / FLAT). Behavioural, not financial.
anchor	FLOAT	Session anchor value. Used for normalisation and behavioural reference.
anchor_ts	INTEGER	Timestamp of anchor selection.
pos	FLOAT	Synthetic position size. Behavioural representation only.
real	FLOAT	Realised synthetic PnL component.
unreal	FLOAT	Unrealised synthetic PnL component.
total	FLOAT	Total synthetic PnL.
sess_hi	FLOAT	Session high watermark.
sess_lo	FLOAT	Session low watermark.
kill	BOOLEAN	Killswitch flag. Indicates behavioural shutdown conditions.
syn_pnl	FLOAT	Synthetic PnL estimate. Useful for ML reward modelling.
exit_armed	BOOLEAN	Behavioural exit condition flag.
size_mult	FLOAT	Position size multiplier. Behavioural scaling factor.

Wave Organ — Behavioural Waveform

Field	Type	Description
wh	FLOAT	Wave height metric. Indicates behavioural amplitude.
pp	FLOAT	Pressure point metric. Often used to detect turning points.
tess_pred	FLOAT	TessQ behavioural prediction.

Field	Type	Description
tess_actual	FLOAT	Actualised behavioural output.
wave_len	FLOAT	Behavioural wave length. Longer waves indicate slower oscillations.
last_exit_ts	INTEGER	Timestamp of last behavioural exit event.
wave_amp_bps	FLOAT	Wave amplitude in basis points.
wave_slope_bps_s	FLOAT	Wave slope (bps/s). Indicates rising or falling behavioural momentum.

Edge Organ — Microstructure Alignment

Field	Type	Description
edge_05ms	BOOLEAN	Micro-edge detection flag. Indicates short-horizon structural edges.
edge_futureAligned	BOOLEAN	Forward alignment indicator. Behaviour aligns with short-term structure.
edge_futureLead	BOOLEAN	Forward lead signal. Behaviour is leading short-term structure.

QDE Organ — Entropy & Branching

Field	Type	Description
qde_branch	INTEGER	Branch index in entropy-driven behavioural tree.
qde_best	INTEGER	Best behavioural branch at this tick.
qde_minEntropy	FLOAT	Minimum entropy state. Lower values indicate clearer behavioural structure.

Tesseract Organ — Behavioural Geometry & Regime Identity

Field	Type	Description
t4d_x	FLOAT	4D manifold coordinate (X). Behavioural position in geometric space.
t4d_y	FLOAT	4D manifold coordinate (Y).
t4d_z	FLOAT	4D manifold coordinate (Z). Often aligned with tick progression.
t4d_w	FLOAT	4D manifold coordinate (W).
t4d_vertex	INTEGER	Nearest attractor vertex (0–15).
t4d_regime	INTEGER	Regime ID corresponding to vertex.
t4d_regime_name	STRING	Human-readable regime name (e.g., “Bear Tense Stable”).

Behavioural Geometry

4D Manifold, Tesseract Scaffold, Attractor Vertices & Regime Identity

The TessQ engine encodes behavioural telemetry into a **geometric state-space**, allowing complex microstructure behaviour to be represented in a stable, interpretable form. This geometric layer is built on a **4-dimensional manifold** projected into a **tesseract scaffold —** a structured behavioural map with **16 attractor vertices**.

This section explains the geometric concepts in clear, accessible terms.

1. The 4D Behavioural Manifold

TessQ represents behaviour inside a **four-dimensional space**, defined by the fields:

- **t4d_x**
- **t4d_y**
- **t4d_z**
- **t4d_w**

These coordinates do **not** correspond to price, time, or any market metric. They represent TessQ's **internal behavioural position —** similar to how a weather model tracks temperature, pressure, humidity, and wind direction simultaneously.

Layman's explanation

Think of the 4D manifold as TessQ's "map of behaviour." Each coordinate shows where the engine is inside its behavioural landscape.

Usage

Researchers use these coordinates for:

- clustering
- regime classification
- manifold learning
- dimensionality reduction
- behavioural similarity analysis

2. The Tesseract Scaffold (16 Behavioural Vertices)

The 4D manifold is projected into a **tesseract** — a geometric structure with **16 vertices**. Each vertex represents a **behavioural attractor**, or "macro-state," that TessQ can collapse into.

These vertices act like **weather zones**:

- some are calm
- some are tense
- some are volatile
- some are transitional
- some are expansionary
- some are compressive

The field **t4d_vertex** identifies which vertex TessQ is closest to at each tick.

Layman's explanation

The tesseract is TessQ's "climate map." Each vertex is a different behavioural climate.

Usage

Vertex identity is useful for:

- regime detection
- state-space modelling
- behavioural segmentation
- ML classification tasks

3. Attractor Collapse & Behavioural Stability

As behaviour evolves, TessQ moves through the 4D manifold. When the engine approaches a vertex, the behavioural geometry **collapses** into that attractor.

This collapse is stable and repeatable — meaning the same behavioural conditions consistently map to the same vertex.

Layman's explanation

When behaviour becomes clear, TessQ "snaps" into the nearest behavioural climate.

Usage

Attractor collapse is useful for:

- identifying regime transitions
- detecting structural shifts
- modelling behavioural stability

- anomaly detection

4. Regime Identity (t4d_regime & t4d_regime_name)

Each vertex corresponds to a **regime** — a human-readable behavioural macro-state.

Examples include:

- **Bear Tense Stable**
- **Bull Calm Expansion**
- **Volatile Break**
- **Neutral Drift**
- **Compression Phase**
- **Expansion Phase**

The fields:

- **t4d_regime**
- **t4d_regime_name**

provide the numeric and human-readable regime identity.

Layman's explanation

Regimes are TessQ's "weather reports." They describe the overall behavioural climate at each tick.

Usage

Regime identity is widely used for:

- ML classification
- strategy segmentation
- regime-aware modelling
- behavioural clustering
- structural analysis

Regime Dictionary

Behavioural Macro-States Mapped to Tesseract Vertices

The TessQ engine assigns a behavioural **regime** at each tick based on its position within the 4D manifold and proximity to one of the 16 tesseract vertices. Each regime represents a **macro-state** — a stable behavioural climate that reflects tension, drift, sigma, curvature, pressure, and wave dynamics.

Regimes are **synthetic behavioural classifications**, not market predictions. They provide a structured way to segment behaviour, detect transitions, and build regime-aware models.

Below is the regime dictionary used in TessQ datasets.

Regime 0 — Neutral Drift

Low tension, low sigma, stable drift

A calm behavioural state with minimal compression or expansion. Behaviour drifts gently without strong directional pressure.

Characteristics

- low drift magnitude
- low sigma
- low tension
- stable curvature
- weak wave amplitude

Usage

- baseline modelling
- low-volatility clustering
- stable ML training windows

Regime 1 — Neutral Compression

Behaviour tightening, tension rising

A transitional state where behavioural tension increases but drift remains modest.

Characteristics

- rising tension
- sigma stable or slightly contracting
- curvature begins to bend inward

- heartbeat latency may tighten

Usage

- pre-transition detection
- anomaly windows
- compression-phase modelling

Regime 2 — Neutral Expansion

Behaviour loosening, sigma rising

Opposite of compression — behaviour expands outward with increasing sigma.

Characteristics

- sigma rising
- tension easing
- drift still modest
- wave amplitude increases

Usage

- expansion-phase modelling
- volatility clustering
- ML feature enrichment

Regime 3 — Bear Tense Stable

High tension, negative drift, stable structure

A stable bearish behavioural climate with elevated tension and negative drift.

Characteristics

- negative drift
- tension elevated
- sigma moderate
- curvature inward
- heartbeat latency stable

Usage

- bearish regime modelling
- structural stability analysis
- ML classification tasks

Regime 4 — Bear Tense Break

High tension, negative drift, structural break emerging

A more volatile bearish state where tension peaks and sigma begins to expand.

Characteristics

- strong negative drift
- tension high
- sigma rising
- curvature sharp
- wave slope steepening

Usage

- break detection
- transition modelling
- anomaly identification

Regime 5 — Bear Expansion

Negative drift with expanding sigma

A bearish expansion phase where volatility increases and behaviour becomes more expressive.

Characteristics

- negative drift
- sigma expanding
- tension easing
- wave amplitude rising

Usage

- volatility modelling
- expansion-phase clustering
- ML regime segmentation

Regime 6 — Bear Volatile

High sigma, unstable tension, negative drift

A volatile bearish climate with unstable tension and large behavioural swings.

Characteristics

- high sigma
- tension oscillating
- drift negative
- wave amplitude large

Usage

- anomaly detection
- high-volatility modelling
- stress-testing ML models

Regime 7 — Bear Collapse

Behaviour collapsing inward, tension releasing

A collapse phase where tension releases sharply and behaviour snaps inward.

Characteristics

- tension dropping
- sigma compressing
- drift negative but weakening
- curvature flattening

Usage

- collapse-phase modelling
- regime transition detection
- ML boundary identification

Regime 8 — Bull Calm Stable

Positive drift, low sigma, stable structure

A calm bullish climate with gentle upward drift and low volatility.

Characteristics

- positive drift
- low sigma
- low tension
- stable curvature

Usage

- bullish regime modelling

- stable ML windows
- low-noise clustering

Regime 9 — Bull Calm Expansion

Positive drift, sigma rising

A bullish expansion phase where behaviour becomes more expressive.

Characteristics

- positive drift
- sigma rising
- tension easing
- wave amplitude increasing

Usage

- expansion-phase modelling
- volatility clustering
- ML feature enrichment

Regime 10 — Bull Volatile

High sigma, unstable tension, positive drift

A volatile bullish climate with large behavioural swings.

Characteristics

- high sigma
- tension oscillating
- drift positive
- wave amplitude large

Usage

- anomaly detection
- high-volatility modelling
- stress-testing ML models

Regime 11 — Bull Break

Positive drift with structural break emerging

A bullish break phase where tension rises and sigma expands.

Characteristics

- strong positive drift
- tension rising
- sigma expanding
- curvature sharp

Usage

- break detection
- transition modelling
- anomaly identification

Regime 12 — Bull Collapse

Behaviour collapsing inward, tension releasing

A collapse phase where bullish behaviour snaps inward.

Characteristics

- tension dropping
- sigma compressing
- drift positive but weakening
- curvature flattening

Usage

- collapse-phase modelling
- regime transition detection

Regime 13 — Volatile Break

High sigma, high curvature, unstable tension

A high-energy break state with unstable structure and large behavioural swings.

Characteristics

- sigma very high
- tension unstable
- curvature sharp
- wave slope steep

Usage

- anomaly detection
- high-volatility clustering

- ML stress-testing

Regime 14 — Volatile Expansion

High sigma, tension easing, behaviour expanding

A volatile expansion phase where behaviour becomes expressive and outward-moving.

Characteristics

- sigma high
- tension easing
- drift variable
- wave amplitude large

Usage

- volatility modelling
- expansion-phase clustering

Regime 15 — Volatile Collapse

High sigma collapsing inward

A collapse phase where volatile behaviour snaps inward and stabilises.

Characteristics

- sigma compressing
- tension dropping
- drift variable
- curvature flattening

Usage

- collapse-phase modelling
- regime transition detection

INTENDED USE CASES

Practical Applications for Machine Learning, Quant Research & Behavioural Modelling

The TessQ behavioural telemetry dataset is designed for teams who need **high-resolution, regime-aware, synthetic behavioural features** for modelling, research, and signal discovery. All fields are numeric, dense, and fully entropy-encoded, making the dataset ideal for modern ML workflows and quantitative pipelines.

Below are the primary use cases, rewritten with clearer explanations and practical guidance.

1. Machine Learning & Feature Engineering

TessQ provides a rich set of behavioural features suitable for:

- supervised learning
- unsupervised clustering
- reinforcement learning
- manifold learning
- state-space modelling

The dataset includes:

- drift
- sigma
- tension
- curvature
- energy
- wave dynamics
- entropy compression
- regime identity
- tesseract coordinates

These features allow ML teams to:

- build predictive models
- extract high-signal features
- train regime-aware architectures
- perform dimensionality reduction
- cluster behavioural states
- detect transitions and anomalies

Layman's explanation

ML teams can treat TessQ like a "synthetic behavioural fingerprint" of the market — dense, stable, and ready for modelling.

2. Quantitative Research & Strategy Development

Researchers can use TessQ to study:

- microstructure behaviour
- volatility regimes
- drift and tension cycles
- expansion/compression phases
- structural transitions
- behavioural clustering
- regime stability

The multi-organ architecture provides a structured view of behaviour that is not visible in traditional OHLCV data.

Layman's explanation

Quants can explore how behaviour changes over time, how regimes form, and how transitions occur — without needing raw market feeds.

3. Regime Detection & Market State Classification

The tesseract scaffold and regime identity fields allow teams to classify behaviour into **macro-states**, such as:

- Bear Tense Stable
- Bull Calm Expansion
- Volatile Break
- Neutral Drift

These regimes are derived from TessQ's geometric manifold and provide a stable, interpretable segmentation of behavioural climate.

Layman's explanation

Regimes act like "weather zones" — helping models understand whether behaviour is calm, tense, expanding, or breaking.

4. Anomaly & Event Detection

High-resolution telemetry enables detection of:

- micro-events
- structural anomalies
- behavioural breaks
- tension spikes
- entropy collapses
- wave surges

- edge alignment flips

These events often precede larger behavioural transitions.

Layman's explanation

TessQ can highlight unusual behaviour that traditional price data would miss.

5. Backtesting & Simulation

Each dataset is a **self-contained session**, making it ideal for:

- controlled backtests
- simulation environments
- synthetic strategy evaluation
- behavioural replay
- model benchmarking

Because the schema is consistent across all datasets, teams can easily compare sessions or run rolling simulations.

Layman's explanation

You can treat each dataset like a "synthetic market replay" with stable behavioural signals.

6. Signal Research & Alpha Discovery

The entropy-encoded behavioural fields provide a rich environment for:

- signal exploration
- hypothesis testing
- feature correlation studies
- behavioural pattern mining
- synthetic alpha discovery

Researchers can identify relationships between drift, sigma, tension, wave dynamics, entropy compression, and regime identity.

Layman's explanation

TessQ gives you a safe playground for discovering new behavioural signals without touching raw market data.

7. Waveform & Pattern Analysis

Wave-based metrics (length, amplitude, slope) support:

- pattern recognition

- waveform decomposition
- oscillation modelling
- behavioural rhythm analysis
- cycle detection

These features are particularly useful for time-series ML and signal processing.

Layman's explanation

The wave organ shows how behaviour "breathes" — rising, falling, flattening, or accelerating.

8. Behavioural Geometry & State-Space Modelling

The 4D manifold and tesseract scaffold allow teams to:

- model behaviour in geometric space
- cluster behavioural states
- detect attractor transitions
- build regime-aware ML architectures
- perform manifold learning
- visualise behavioural trajectories

Layman's explanation

The geometric layer turns behaviour into a map — making it easier to see where the engine is and where it's going.

KEY FEATURES

High-Resolution, Multi-Organ Behavioural Telemetry

The TessQ dataset provides a dense, interpretable, and regime-aware behavioural feature set designed for machine learning, quantitative research, and microstructure modelling. All fields are synthetic, entropy-encoded, and safe for commercial use.

Below are the key features of the TessQ behavioural telemetry system, rewritten with expanded detail and clearer usage guidance.

1. High-Resolution Telemetry (1,000,000 Ticks)

Each dataset contains **1 million sequential ticks**, capturing behavioural dynamics with millisecond-level responsiveness. This resolution allows researchers to observe:

- micro-events
- structural transitions
- regime shifts
- tension cycles
- volatility expansions
- behavioural collapses

Why it matters

High-resolution data reveals patterns that traditional OHLCV feeds cannot show, making TessQ ideal for advanced ML and microstructure research.

2. Multi-Organ Behavioural Architecture

TessQ processes market movements through multiple behavioural organs, including:

- Prism (drift, sigma, tension, curvature, energy)
- Heartbeat (latency, timing, behavioural pulse)
- Sentinel (pressure, consensus, micro-signals)
- Wave (amplitude, slope, length)
- Edge (microstructure alignment)
- QDE (entropy compression & branching)
- Tesseract (4D manifold & regime identity)

Why it matters

This architecture provides a **layered behavioural perspective**, giving ML and quant teams access to rich, multi-dimensional signals.

3. Fully Derived & Entropy-Encoded Fields

All fields are:

- synthetic
- derived
- entropy-encoded
- non-reversible
- safe for commercial use

No raw market data or proprietary feed content is included.

Why it matters

You get high-signal behavioural features without compliance risk or data licensing complexity.

4. Consistent Schema Across All Datasets

Every dataset follows the same schema, making ingestion and modelling straightforward.

Why it matters

Teams can:

- train models across multiple sessions
- run rolling simulations
- compare behavioural regimes
- build unified pipelines
- avoid schema drift

This consistency is essential for enterprise-grade ML workflows.

5. Behavioural & Predictive Signals

The dataset includes high-signal behavioural metrics such as:

- drift
- sigma
- tension
- curvature
- energy
- pressure
- consensus
- wave dynamics
- entropy compression
- regime identity

Why it matters

These fields provide a structured behavioural fingerprint that supports:

- predictive modelling
- feature engineering
- anomaly detection
- regime classification
- signal discovery

6. Synthetic but Market-Aligned Behaviour

TessQ responds to real market movements while producing **fully synthetic behavioural telemetry**.

Why it matters

You get realistic behavioural structure without exposing raw market data or proprietary feeds.

7. Self-Contained Sessions

Each dataset is a standalone behavioural session with:

- deterministic feature generation
- no external dependencies
- stable organ behaviour
- consistent geometric structure

Why it matters

This makes TessQ ideal for:

- back-testing
- simulation
- replay environments
- model benchmarking

8. Machine-Learning Ready

All fields are:

- numeric
- dense
- stable
- free of missing values
- aligned with ML best practices

Why it matters

ML teams can use TessQ immediately without cleaning, imputation, or pre-processing.

9. Ideal for Regime Detection

The tesseract scaffold and regime identity fields provide a clear, interpretable segmentation of behavioural climate.

Why it matters

Regime-aware models outperform regime-agnostic ones in:

- volatility modelling
- anomaly detection
- predictive pipelines
- reinforcement learning
- clustering

DATA GENERATION NOTES

How TessQ Produces Synthetic, High-Resolution Behavioural Telemetry

The telemetry in this dataset is generated by the **TessQ behavioural engine**, a synthetic multi-organ system designed to encode market microstructure into a stable, high-signal behavioural manifold. TessQ operates in real time, responding to live market movements while producing **fully derived, entropy-encoded features** that cannot be reversed into raw market data.

TessQ does **not** store or expose raw exchange feeds. All outputs are synthetic behavioural representations derived from the engine's internal state.

1. Multi-Organ Behavioural Processing

TessQ processes incoming micro-movements through a set of interacting behavioural organs:

- **Prism Organ —** drift, sigma, tension, curvature, energy
- **Heartbeat Organ —** latency, timing, behavioural pulse
- **Sentinel Organ —** pressure, consensus, directional micro-signals
- **Wave Organ —** amplitude, slope, length
- **Edge Organ —** microstructure alignment & forward cues
- **QDE Organ —** entropy compression & branching
- **Tesseract Organ —** 4D manifold, attractor vertices, regime identity

Each organ contributes a different behavioural perspective. Together they form a **coherent behavioural manifold** that evolves tick-by-tick.

Layman's explanation

TessQ looks at behaviour from multiple angles at once — tension, drift, volatility, pressure, wave shape, entropy, and geometric position — then fuses these into a single behavioural snapshot.

2. Behavioural Manifold & Tesseract Projection

The engine encodes behaviour inside a **4-dimensional manifold**, represented by:

- t4d_x
- t4d_y
- t4d_z
- t4d_w

This manifold is projected into a **tesseract scaffold** with **16 attractor vertices**, each representing a behavioural macro-state (regime).

The engine collapses into the nearest vertex based on geometric proximity, producing:

- **t4d_vertex**
- **t4d_regime**
- **t4d_regime_name**

Layman's explanation

TessQ places each tick inside a behavioural "map," then identifies which behavioural climate it belongs to.

3. Entropy Encoding & Non-Reversibility

All fields are **entropy-encoded**, meaning they cannot be reversed into raw market data. This encoding ensures:

- commercial safety
- compliance with data licensing rules
- protection of proprietary feed content
- protection of internal engine mechanics

Layman's explanation

The dataset shows behaviour, not price feeds. It's impossible to reconstruct market data from TessQ telemetry.

4. Deterministic Feature Generation

Each dataset is produced using a **deterministic generation process**, ensuring:

- identical schema across sessions
- stable organ behaviour
- reproducible results
- consistent feature ranges
- predictable ML ingestion

Layman's explanation

Every dataset behaves the same way — same fields, same structure, same organ outputs — making it easy to train models across multiple sessions.

5. Real-Time Behavioural Response

TessQ operates in real time, processing micro-movements as they occur. The engine reacts to:

- volatility expansions
- tension spikes
- drift shifts
- wave surges

- entropy collapses
- structural breaks
- regime transitions

These reactions are encoded into the behavioural fields.

Layman’s explanation

TessQ “feels” the behaviour of the market and expresses it through its organs.

6. Synthetic but Market-Aligned

Although fully synthetic, TessQ’s behavioural responses are aligned with real market conditions. This alignment ensures:

- realistic behavioural structure
- meaningful regime transitions
- interpretable tension cycles
- credible volatility patterns
- useful ML signals

Layman’s explanation

The behaviour is real enough to be useful, but synthetic enough to be safe.

7. Self-Contained Sessions

Each dataset is a standalone behavioural session with:

- no external dependencies
- no cross-session state
- no historical carryover
- no forward-looking leakage

This ensures:

- clean back-testing
- reproducible modelling
- stable ML pipelines
- consistent ingestion

Layman’s explanation

Every dataset is its own world — clean, isolated, and ready for modelling.

8. No Raw Market Data Included

TessQ does not expose:

- raw exchange feeds
- proprietary market data
- timestamps from exchanges
- order book information
- trade-level data

All fields are derived from TessQ's internal behavioural state.

Layman's explanation

The dataset is safe, synthetic, and compliant — no raw market data is ever included.

SAMPLE SQL QUERIES

Exploring, Modelling & Extracting Behavioural Geometry from TessQ Telemetry

These examples demonstrate how to explore, analyse, and model TessQ behavioural telemetry using standard SQL in Snowflake. They include both traditional behavioural queries and advanced geometric/tesseract-based analysis.

1. Preview the first 100 ticks

sql

```
SELECT *
FROM TESSQ_DATASET
ORDER BY ts
LIMIT 100;
```

Useful for verifying ingestion, schema, and field availability.

2. Inspect core behavioural fields

sql

```
SELECT ts, px, pr_drift, pr_sigma, pr_tension, pr_curvature
FROM TESSQ_DATASET
ORDER BY ts;
```

Shows the Prism Organ’s predictive envelope.

3. Identify regime shifts using sigma & tension

sql

```
SELECT ts, pr_sigma, pr_tension, t4d_regime_name
FROM TESSQ_DATASET
WHERE pr_sigma > 0.25
   OR pr_tension < -0.10
ORDER BY ts;
```

Useful for detecting structural transitions.

4. Compute rolling averages for ML feature engineering

sql

```
SELECT
    ts,
    px,
    AVG(pr_sigma) OVER (ORDER BY ts ROWS BETWEEN 50 PRECEDING AND CURRENT
ROW) AS sigma_ma_50,
    AVG(pr_drift) OVER (ORDER BY ts ROWS BETWEEN 50 PRECEDING AND CURRENT
ROW) AS drift_ma_50
FROM TESSQ_DATASET;
```

Creates smoothed features for ML pipelines.

5. Detect high-energy behavioural events

sql

```
SELECT ts, pr_energy, t4d_regime_name
FROM TESSQ_DATASET
WHERE pr_energy > 0.50
ORDER BY ts;
```

Energy spikes often precede regime transitions.

6. Analyse wave dynamics

sql

```
SELECT ts, wave_len, wave_amp_bps, wave_slope_bps_s
FROM TESSQ_DATASET
ORDER BY ts;
```

Useful for waveform modelling and cycle detection.

7. Build a simple prediction baseline

sql

```
SELECT
    ts,
    tess_pred,
    tess_actual,
    (tess_actual - tess_pred) AS error
FROM TESSQ_DATASET;
```

Provides a baseline error metric.

8. Extract sentinel (decision-kernel) behaviour

sql

```sql
SELECT ts, dk_pressure, dk_consensus, dk_power
FROM TESSQ_DATASET
ORDER BY ts;
```

Useful for micro-signal analysis.

9. Identify anomaly windows

sql

```sql
SELECT ts, pr_sigma, pr_tension, pr_energy
FROM TESSQ_DATASET
WHERE pr_energy > 0.40
   OR ABS(pr_tension) > 0.15
ORDER BY ts;
```

Detects behavioural outliers.

10. Inspect tesseract coordinates (4D manifold)

Understanding behavioural geometry

sql

```sql
SELECT
    ts,
    t4d_x,
    t4d_y,
    t4d_z,
    t4d_w
FROM TESSQ_DATASET
ORDER BY ts;
```

Shows how behaviour moves through the 4D manifold.

11. Extract regime identity from tesseract vertices

sql

```
SELECT
    ts,
    t4d_vertex,
    t4d_regime,
    t4d_regime_name
FROM TESSQ_DATASET
ORDER BY ts;
```

Provides a clean view of macro-state behaviour.

12. Count how many ticks fall into each regime

sql

```
SELECT
    t4d_regime_name,
    COUNT(*) AS tick_count
FROM TESSQ_DATASET
GROUP BY t4d_regime_name
ORDER BY tick_count DESC;
```

Shows dominant behavioural climates.

13. Detect regime transitions (vertex changes)

Find moments where behaviour jumps between attractors

sql

```
SELECT
    ts,
    t4d_vertex,
    LAG(t4d_vertex) OVER (ORDER BY ts) AS prev_vertex
FROM TESSQ_DATASET
QUALIFY t4d_vertex != prev_vertex;
```

Useful for transition modelling and anomaly detection.

14. Measure regime duration

sql

```
SELECT
    t4d_regime_name,
    MIN(ts) AS start_ts,
    MAX(ts) AS end_ts,
    (MAX(ts) - MIN(ts)) AS duration_ticks
FROM TESSQ_DATASET
GROUP BY t4d_regime_name
ORDER BY duration_ticks DESC;
```

Shows how long each regime persists.

15. Cluster behaviour using tesseract coordinates

Simple geometric clustering in SQL

sql

```
SELECT
    ROUND(t4d_x, 1) AS x_bin,
    ROUND(t4d_y, 1) AS y_bin,
    ROUND(t4d_z, 1) AS z_bin,
    ROUND(t4d_w, 1) AS w_bin,
    COUNT(*) AS cluster_size
FROM TESSQ_DATASET
GROUP BY x_bin, y_bin, z_bin, w_bin
ORDER BY cluster_size DESC;
```

Creates coarse geometric clusters without ML.

16. Identify high-entropy vs low-entropy behavioural states

sql

```
SELECT
    ts,
    qde_minEntropy,
    t4d_regime_name
FROM TESSQ_DATASET
WHERE qde_minEntropy < 0.10
   OR qde_minEntropy > 0.50
ORDER BY ts;
```

Low entropy = stable structure High entropy = unstable or transitional behaviour

17. Combine geometry + tension + sigma for advanced analysis

sql

```
SELECT
    ts,
    pr_tension,
    pr_sigma,
    t4d_vertex,
    t4d_regime_name
FROM TESSQ_DATASET
WHERE pr_sigma > 0.20
   AND pr_tension < -0.05
ORDER BY ts;
```

Useful for detecting structural breaks.

18. Visualise behavioural trajectories (export-friendly)

Prepare data for external plotting tools

sql

```
SELECT
    ts,
    t4d_x,
    t4d_y,
    t4d_z,
    t4d_w,
    t4d_regime_name
FROM TESSQ_DATASET
ORDER BY ts;
```

This dataset can be exported to Python/R for manifold visualisation.

Advanced SQL Examples — Tess-QDE4D Telemetry

□ Manifold Trajectory (4D Behavioural Path)

sql

```
SELECT
    ts,
    t4d_x,
    t4d_y,
    t4d_z,
    t4d_w
FROM TESSQ_DATASET
ORDER BY ts;
```

□ Attractor Vertex Transitions (Tesseract Jumps)

sql

```
SELECT
    ts,
    t4d_vertex,
    LAG(t4d_vertex) OVER (ORDER BY ts) AS prev_vertex
FROM TESSQ_DATASET
QUALIFY t4d_vertex != prev_vertex;
```

□ Regime Duration (Macro-State Stability)

sql

```
SELECT
    t4d_regime_name,
    MIN(ts) AS start_ts,
    MAX(ts) AS end_ts,
    MAX(ts) - MIN(ts) AS duration_ticks
FROM TESSQ_DATASET
GROUP BY t4d_regime_name
ORDER BY duration_ticks DESC;
```

□ Entropy Collapse Events (QDE Stability Signals)

sql

```
SELECT
    ts,
    qde_minEntropy,
    t4d_regime_name
FROM TESSQ_DATASET
WHERE qde_minEntropy < 0.08
ORDER BY ts;
```

□ Branching Path Analysis (QDE Multi-Path Evaluation)

sql

```
SELECT
    ts,
    qde_branch,
    qde_best,
    qde_minEntropy
FROM TESSQ_DATASET
ORDER BY ts;
```

□ Regime-Aligned Manifold Clustering (4D Binning)

sql

```
SELECT
    t4d_regime_name,
    ROUND(t4d_x, 1) AS x_bin,
    ROUND(t4d_y, 1) AS y_bin,
    ROUND(t4d_z, 1) AS z_bin,
    ROUND(t4d_w, 1) AS w_bin,
    COUNT(*) AS cluster_size
FROM TESSQ_DATASET
GROUP BY t4d_regime_name, x_bin, y_bin, z_bin, w_bin
ORDER BY cluster_size DESC;
```

□ Curvature Surges (Prism Structural Bends)

sql

```
SELECT
    ts,
    pr_curvature,
    t4d_regime_name
FROM TESSQ_DATASET
WHERE pr_curvature > 0.30
ORDER BY ts;
```

□ Waveform Break Detection (Wave Organ Acceleration)

sql

```
SELECT
    ts,
    wave_len,
    wave_amp_bps,
    wave_slope_bps_s
FROM TESSQ_DATASET
WHERE wave_slope_bps_s > 0.50
ORDER BY ts;
```

□ Edge Alignment Flips (Microstructure Forward Cues)

sql

```
SELECT
    ts,
    edge_futureAligned,
    edge_futureLead,
    LAG(edge_futureAligned) OVER (ORDER BY ts) AS prev_align
FROM TESSQ_DATASET
QUALIFY edge_futureAligned != prev_align;
```

□ Regime Transition Map (From → To)

sql

```
SELECT
    ts,
    t4d_regime_name,
    LAG(t4d_regime_name) OVER (ORDER BY ts) AS prev_regime
FROM TESSQ_DATASET
QUALIFY t4d_regime_name != prev_regime;
```

□ High-Energy Behavioural Events (Prism Energy Spikes)

sql

```
SELECT
    ts,
    pr_energy,
    t4d_regime_name
FROM TESSQ_DATASET
WHERE pr_energy > 0.45
ORDER BY ts;
```

□ Manifold Velocity (4D Movement Speed)

sql

```
SELECT
    ts,
    SQRT(
        POWER(t4d_x - LAG(t4d_x) OVER (ORDER BY ts), 2) +
        POWER(t4d_y - LAG(t4d_y) OVER (ORDER BY ts), 2) +
        POWER(t4d_z - LAG(t4d_z) OVER (ORDER BY ts), 2) +
        POWER(t4d_w - LAG(t4d_w) OVER (ORDER BY ts), 2)
    ) AS manifold_velocity
FROM TESSQ_DATASET;
```

□ Manifold Acceleration (Second-Order Behavioural Motion)

sql

```
WITH vel AS (
    SELECT
        ts,
        SQRT(
            POWER(t4d_x - LAG(t4d_x) OVER (ORDER BY ts), 2) +
            POWER(t4d_y - LAG(t4d_y) OVER (ORDER BY ts), 2) +
            POWER(t4d_z - LAG(t4d_z) OVER (ORDER BY ts), 2) +
            POWER(t4d_w - LAG(t4d_w) OVER (ORDER BY ts), 2)
        ) AS v
    FROM TESSQ_DATASET
)
SELECT
    ts,
    v,
    v - LAG(v) OVER (ORDER BY ts) AS manifold_acceleration
FROM vel;
```

□ Entropy-Weighted Regime Confidence

sql

```
SELECT
    ts,
    t4d_regime_name,
    1 - qde_minEntropy AS regime_confidence
FROM TESSQ_DATASET
ORDER BY ts;
```

□ Full Behavioural Fusion Snapshot (All Organs Together)

sql

```
SELECT
    ts,
    pr_drift,
    pr_sigma,
    pr_tension,
    dk_pressure,
    dk_consensus,
    wave_amp_bps,
    qde_minEntropy,
    t4d_vertex,
    t4d_regime_name
FROM TESSQ_DATASET
ORDER BY ts;
```

DATASET METADATA & VERSIONING

Naming, Versioning, Schema Stability & Session Independence

Each TessQ dataset is delivered as a **standalone behavioural session** with a consistent schema, deterministic feature generation, and stable organ behaviour. This section explains how datasets are named, versioned, and structured for reproducibility and enterprise-grade ingestion.

□ Dataset Naming Convention

Each dataset follows the naming pattern:

Code
```
TESSQ_<SIZE>_<SESSIONID>_<VERSION>
```

Examples

- `TESSQ_1M_20240621_V1`
- `TESSQ_500K_20240621_V1`
- `TESSQ_2M_20240621_V2`

Fields

- **SIZE** — number of ticks (e.g., 1M, 500K, 2M).
- **SESSIONID —** unique identifier for the session (typically a date or session code).
- **VERSION —** version number for that specific dataset.

Layman’s explanation

The name tells you how big the dataset is, when it was generated, and which version it is.

□ Versioning Policy

Although this is the **first public release**, TessQ datasets follow a clear versioning policy for future updates.

V1 — Initial Release

The first published version of a dataset.

V2, V3, etc. — Only Published If:

- schema improvements are made
- additional fields are added
- corrections or enhancements are applied
- organ behaviour is refined

- geometric structure is expanded

All versions remain available for:

- reproducibility
- back-testing
- long-term research
- model comparison

Layman's explanation

If the dataset ever changes, the version number increases — so you always know exactly what you're working with.

□ Schema Stability

All TessQ datasets share the same schema. No breaking schema changes are introduced without incrementing the version number.

This ensures:

- consistent ingestion
- predictable ML workflows
- stable feature engineering
- cross-session comparability
- long-term reproducibility

Layman's explanation

Every dataset has the same fields in the same order — no surprises.

□ Session Independence

Each dataset is a **self-contained behavioural session** with:

- no dependencies on previous sessions
- no forward-looking leakage
- no historical carryover
- no shared state
- no cross-session memory

This guarantees:

- clean back-testing
- isolated behavioural replay
- reproducible modelling
- stable ML pipelines

Layman’s explanation

Every dataset is its own world — nothing leaks in or out.

□ Metadata Included in Every Dataset

Each dataset includes the following metadata fields:

- **tick count** — number of ticks in the session
- **session ID** — unique identifier
- **version number** — dataset version
- **schema version** — schema revision
- **generation timestamp** — when the dataset was produced

This metadata ensures:

- traceability
- compatibility
- reproducibility
- auditability

Layman’s explanation

You always know exactly what dataset you’re using, when it was made, and what version it is.

TessQ Engine Overview

Synthetic Behavioural Manifold Engine for Microstructure Interpretation

The TessQ engine is a **multi-organ behavioural system** designed to interpret market microstructure and encode it into a stable, high-signal synthetic telemetry stream. It operates as a **state-space, event-driven engine**, transforming micro-movements into a structured behavioural manifold that captures drift, sigma, tension, curvature, pressure, consensus, wave dynamics, entropy compression, and regime identity.

TessQ does not expose raw market data. All outputs are **fully synthetic, derived, entropy-encoded, and non-reversible**.

1. Multi-Organ Behavioural Architecture

TessQ processes incoming behavioural inputs through several interacting organs. Each organ contributes a different behavioural perspective:

Prism Organ

Encodes the predictive envelope:

- drift
- sigma
- tension
- curvature
- energy
- lateral components
- regression estimates

Heartbeat Organ

Tracks internal timing and responsiveness:

- tick interval
- latency
- heartbeat field strength

Sentinel Organ

Provides lightweight micro-signals:

- pressure
- consensus
- directional cues

Wave Organ

Encodes behavioural oscillation:

- amplitude
- slope
- length

Edge Organ

Identifies short-horizon structural alignment:

- micro-edge detection
- forward alignment
- forward lead

QDE Organ

Tracks entropy compression and branching:

- branch index
- best path
- minimum entropy

Tesseract Organ

Provides geometric structure and regime identity:

- 4D manifold coordinates
- attractor vertex
- regime ID
- regime name

Layman's explanation

Each organ looks at behaviour from a different angle — tension, drift, volatility, pressure, wave shape, entropy, geometry — and together they form a complete behavioural picture.

2. Behavioural Interference & Multi-Path Evaluation

TessQ evaluates multiple behavioural pathways simultaneously. This is inspired by **interference-based processing**, where coherent behavioural signals reinforce each other while noise is suppressed.

The engine continuously balances:

- expansion vs compression
- drift vs tension
- volatility vs stability
- entropy vs structure
- wave amplitude vs slope
- edge alignment vs misalignment

Layman's explanation

TessQ compares many possible behavioural interpretations at once and chooses the most stable one.

3. Behavioural Manifold (4D State-Space)

All organ outputs feed into a **4-dimensional behavioural manifold**, represented by:

- **t4d_x**
- **t4d_y**
- **t4d_z**
- **t4d_w**

These coordinates describe TessQ's internal behavioural position.

They do **not** correspond to price, time, or market data. They represent the engine's **behavioural geometry**.

Layman's explanation

The manifold is TessQ's "map of behaviour." Each coordinate shows where the engine is inside its behavioural landscape.

4. Tesseract Scaffold (16 Attractor Vertices)

The 4D manifold is projected into a **tesseract** — a geometric structure with **16 attractor vertices**.

Each vertex represents a **behavioural macro-state**, such as:

- Neutral Drift
- Bear Tense Stable
- Bull Calm Expansion
- Volatile Break
- Compression Phase
- Expansion Phase

The engine collapses into the nearest vertex based on geometric proximity.

Layman's explanation

The tesseract is TessQ's "climate map." Each vertex is a different behavioural climate.

5. Regime Identity

Each vertex corresponds to a **regime**, represented by:

- **t4d_regime** — numeric ID
- **t4d_regime_name** — human-readable name

Regimes provide a stable, interpretable segmentation of behavioural climate.

Layman's explanation

Regimes are TessQ's "weather reports" — describing whether behaviour is calm, tense, expanding, or breaking.

6. Entropy Encoding & Safety

All outputs are entropy-encoded, ensuring:

- non-reversibility
- commercial safety
- compliance with market-data rules
- protection of proprietary feeds
- protection of internal engine mechanics

Layman's explanation

The dataset shows behaviour, not raw market data — and cannot be reversed.

7. Deterministic, Schema-Stable Telemetry

TessQ produces deterministic behavioural telemetry with:

- consistent schema
- stable organ behaviour
- reproducible outputs
- predictable feature ranges
- no missing values

Layman's explanation

Every dataset behaves the same way — same fields, same structure — making it easy to train models across multiple sessions.

8. Real-Time Behavioural Response

The engine reacts to micro-events in real time, including:

- volatility expansions
- tension spikes
- drift shifts
- wave surges
- entropy collapses
- structural breaks
- regime transitions

These reactions are encoded into the behavioural fields.

Layman's explanation

TessQ "feels" the behaviour of the market and expresses it through its organs.

9. Synthetic but Market-Aligned

Although fully synthetic, TessQ's behavioural responses are aligned with real market conditions, ensuring:

- realistic behavioural structure
- meaningful regime transitions
- credible volatility patterns
- interpretable tension cycles
- useful ML signals

Layman's explanation

The behaviour is real enough to be useful, but synthetic enough to be safe.

Independent Synthetic Behavioural Data Creator

TML Mapping is an independent behavioural-data creator specialising in **high-resolution synthetic telemetry** for machine learning, quantitative research, and microstructure modelling. The organisation focuses on building structured, interpretable behavioural datasets derived from multi-organ synthetic engines, including the TessQ behavioural manifold system.

TML Mapping produces **fully synthetic, entropy-encoded datasets** designed for:

- regime-aware modelling
- microstructure analysis
- ML feature engineering
- anomaly detection
- behavioural clustering
- state-space research

All datasets are generated using proprietary behavioural engines that transform market movements into **non-reversible synthetic telemetry**, ensuring complete commercial safety and compliance.

□ Our Focus

TML Mapping specialises in:

- synthetic behavioural manifolds
- multi-organ telemetry systems
- geometric regime classification
- high-resolution microstructure modelling
- structured behavioural datasets for ML pipelines

The goal is to provide researchers and ML teams with **clean, consistent, high-signal behavioural data** that is easy to ingest, easy to model, and rich enough to support advanced research.

□ Our Approach

TML Mapping follows three core principles:

1. Synthetic First

All datasets are fully synthetic and entropy-encoded. No raw market data is stored, exposed, or reconstructable.

2. Behavioural Clarity

Datasets emphasise behavioural structure — drift, sigma, tension, curvature, pressure, wave dynamics, entropy, and regime identity — rather than traditional OHLCV formats.

3. Schema Stability

Every dataset follows a consistent schema, enabling:

- reproducible modelling
- long-term research
- cross-session comparisons
- stable ML ingestion

□ Who Uses Our Data

TML Mapping datasets are designed for:

- quantitative researchers
- machine learning teams
- financial analytics groups
- synthetic intelligence developers
- academic researchers
- behavioural-modelling specialists

These users rely on TessQ telemetry for:

- regime detection
- anomaly identification
- feature engineering
- microstructure analysis
- synthetic alpha discovery
- state-space modelling

□ Why Synthetic Behavioural Data Matters

Synthetic behavioural telemetry provides:

- high-signal structure
- regime-aware segmentation
- micro-event clarity
- geometric interpretability
- ML-ready numeric fields
- safe, compliant data access

It allows teams to explore behavioural dynamics without the licensing, compliance, or reconstruction risks associated with raw market feeds.

□ Provider Philosophy

TML Mapping believes that behavioural modelling should be:

- **interpretable**
- **geometric**
- **regime-aware**
- **synthetic**
- **safe**
- **consistent**
- **machine-learning friendly**

The TessQ engine embodies this philosophy by providing a structured behavioural manifold with multi-organ telemetry and tesseract-based regime identity.

CONTACT & SUPPORT INFORMATION

Assistance, Technical Support & Dataset Guidance

For questions, technical assistance, or support related to TessQ behavioural telemetry datasets, please use the contact details below. Support is available for:

- dataset ingestion
- schema clarification
- behavioural field interpretation
- regime identity usage
- SQL examples and modelling guidance
- integration into ML or quant pipelines
- general product questions

All inquiries are handled directly by the provider.

□ Primary Contact

TML Mapping — Behavioural Data Provider Email: support@tmlmapping.com **Location:** Stirling, Scotland, United Kingdom **Hours:** Monday–Friday, 09:00–17:00 (UK time)

□ Support Coverage

Support includes:

- help understanding behavioural fields
- guidance on regime identity and tesseract geometry
- assistance with SQL queries and analysis
- clarification on organ outputs
- dataset usage recommendations
- troubleshooting ingestion issues
- documentation questions

□ Response Times

Typical response times:

- **Standard inquiries:** within 1–2 business days
- **Technical questions:** within 2–3 business days
- **Complex modelling guidance:** scheduled individually

□ Additional Assistance

If you require:

- custom dataset sizes
- extended behavioural sessions
- specialised telemetry fields
- integration support
- research collaboration

Please reach out via email to discuss options.