Data Warehouse Use Cases — Needed vs. Overkill

When a data warehouse is genuinely needed

- **BI/reporting** — ad hoc slicing across dimensions (region, time, product) at speed; row-stores choke on this at scale.
- **Historical/point-in-time analysis** — cohort retention, trend analysis, "what did exposure look like on a given date." Needs immutable historical snapshots that production systems don't retain.
- **ML feature engineering / training data prep** — building wide feature tables by joining large historical fact/dimension sources; needs efficient scans/joins at volume.
- **Regulatory & compliance reporting** — audits need reproducible, consistent historical figures with lineage, not a moving production DB.
- **Cross-domain analytics / data mesh consumption** — joining multiple business domains' data together; exactly the "N sources, M consumers, no shared contract" problem centralized modeling solves.
- **Forecasting & capacity planning** — needs large historical time series at consistent granularity.
- **Aggregate-level anomaly/trend detection** — e.g. "revenue by region dropped 20% week over week"; inherently an aggregation-over-time problem.

Common thread: all need to scan/aggregate across large historical volumes with consistent, centrally-defined meaning — the two root causes (columnar reads + fact/dimension governance).

When a data warehouse is overkill or the wrong tool

- **Single-record, low-latency lookups** (get this customer's cart/profile) — the OLTP point-lookup pattern; columnar storage actually hurts here. Use the operational DB.

- **Real-time operational decisions inside a live transaction** (fraud scoring at checkout, inventory decrement) — needs millisecond latency and current state; warehouses are typically minutes-to-hours behind via batch/CDC.
- **Full-text search / relevance ranking** — not built for inverted indices or scoring; use Elasticsearch/OpenSearch.
- **Unstructured/semi-structured blob storage** — raw images, PDFs, logs, un-modeled JSON; forcing this into a star schema is premature structure — use a data lake (or the raw layer of a lakehouse) first.
- **High-frequency small writes** — IoT sensor ingestion event-by-event; warehouses are read-optimized/batch-oriented. Use a time-series DB or streaming store.
- **Simple, low-volume reporting for a small team** — a few thousand rows, one monthly summary; the ETL/governance/infra overhead isn't justified. A well-indexed Postgres view or spreadsheet suffices.
- **Graph-shaped queries** — path-finding, fraud rings, network traversal; star schemas/columnar engines aren't built for relationship traversal. Use a graph database.

Common thread: low-latency point access, unstructured content, full-text relevance, high-frequency writes, or relationship traversal are all problems a warehouse wasn't designed to solve.

Grouped by business domain

Telecom

- *Needs it:* CDR analysis across billions of records, churn prediction feature tables, network capacity planning, regulatory usage/coverage reporting.
- *Overkill/wrong tool:* real-time call routing, live network fault detection (streaming/time-series), subscriber lookup during a support call (OLTP).

Retail / e-commerce

- *Needs it:* sales analysis across store/product/time, inventory trend forecasting, customer segmentation/LTV, marketing attribution.
- *Overkill/wrong tool:* checking current stock for one SKU at checkout, product search/autocomplete (search engine), real-time cart/session state.

Consulting

- *Needs it:* multi-client benchmarking and longitudinal engagement analysis, internal utilization/profitability reporting across practices and time.

- *Overkill/wrong tool:* a single short engagement's ad hoc analysis on a one-off dataset — a spreadsheet or lightweight Postgres/DuckDB is enough.

Industry / manufacturing

- *Needs it:* production yield analysis across plants/lines/time, supply chain and demand forecasting, quality trend analysis joining defects to equipment/shift/supplier.
- *Overkill/wrong tool:* real-time sensor/IoT ingestion and shop-floor anomaly alerts (time-series/streaming), machine-level current-status lookups.

Finance / banking

- *Needs it:* regulatory historical reporting, risk exposure trend analysis, AML pattern detection across large transaction histories.
- *Overkill/wrong tool:* fraud scoring at the moment of a transaction (real-time), account balance lookup (OLTP).

Healthcare

- *Needs it:* population health trend analysis, claims/cost analysis across providers and time, research cohort analysis.
- *Overkill/wrong tool:* pulling a single patient's current chart during a visit (OLTP), clinical decision support needing sub-second current data.

Pattern across every domain: warehouse = historical, cross-dimensional, aggregatable analysis at real volume. Anything current-state, single-record, sub-second, or small-scale is the wrong job for it, regardless of industry.

Revision #2
Created 31 August 2026 09:05:48 by N8N
Updated 28 September 2026 14:39:55 by EMB