

Excellent — you’re essentially building the “**interaction corpus**” or **experience graph** for the AI agent, where each *parent scenario* (1, 2, 3, 5) serves as a thematic cluster.

Below is a structured, high-fidelity metadata set describing 12 total conversation threads — each labeled with:

- **Conversation ID & Title**
 - **Parent Scenario**
 - **Initiator** (who begins — employee or AI agent)
 - **Primary Data Inputs** (what the conversation uses or references)
 - **Context Summary** (how the thread fits within workflow)
 - **Example Flow / Follow-ups** (what kind of dialogue would unfold)
-

Parent 1: Demand Forecasting & Inventory Optimization

Conversation 1.1 — “Regional Stockout Anomaly”

- **Initiator:** Employee
- **Data:** `Sales_Week_42_Regional.csv`
 - Columns: `Store_ID, SKU, Units_Sold, Forecasted_Units, Date, Region`
- **Context:** Analyst uploads recent weekly sales data after noticing an unexpected stockout in the Southeast region.
- **Flow:**
 - Employee: “We ran out of cold brew faster than expected — can you check the forecast error pattern?”

- Agent compares forecast vs. actual demand, clusters outliers by temperature and promo activity, and later suggests model recalibration using weather features.
-

Conversation 1.2 — “Holiday Demand Adjustment”

- **Initiator:** AI Agent
 - **Data:** Internal model outputs (no upload; uses prior forecasts).
 - **Context:** The agent pre-emptively flags that past three years show a 40 % demand uplift for certain SKUs two weeks before Thanksgiving.
 - **Flow:**
 - Agent proposes early shipment adjustments.
 - Employee asks for confidence intervals and transport impact before approving.
 - Follow-up conversation links to Scenario 2 on logistics.
-

Conversation 1.3 — “SKU Rationalization Proposal”

- **Initiator:** Employee
- **Data:** [LongTail_SKU_Performance.xlsx](#)
 - Contains `SKU_ID`, `Avg_Weekly_Sales`, `Shelf_Space_Cost`, `Margin`, `Substitutability_Score`.
- **Context:** Employee wants to reduce low-velocity SKUs.
- **Flow:**
 - Agent identifies 10 % of SKUs contributing <1 % of sales but occupying 8 % of shelf space.

- Conversation evolves into cost-benefit reasoning — “If we delist SKU 5432, how much warehouse space do we free?”
-



Parent 2: Supply Chain Disruption Response

Conversation 2.1 — “Port Congestion Alert”

- **Initiator:** AI Agent
 - **Data:** External feed: `Global_Shipping_Status_API.json` + internal supplier ETAs.
 - **Context:** Agent detects port delays at Long Beach impacting incoming electronics.
 - **Flow:**
 - Agent sends summary with expected delay days per supplier.
 - Employee asks for alternate routing feasibility and cost deltas.
 - Follow-up uses demand data from Conversation 1.2 to re-prioritize shipments.
-

Conversation 2.2 — “Alternate Supplier Simulation”

- **Initiator:** Employee
- **Data:** `Supplier_Performance_Q3.csv`
 - Fields: `Supplier_ID`, `Product_Line`, `Lead_Time`, `Unit_Cost`, `On_Time_Rate`, `Region`.
- **Context:** Evaluating replacement suppliers due to recurring lateness.
- **Flow:**
 - Agent performs optimization comparing reliability-adjusted cost.

- Employee asks for risk score heatmap; discussion extends into Scenario 5.
-

Conversation 2.3 — “Disaster Response Playbook Drill”

- **Initiator:** AI Agent
 - **Data:** Synthetic scenario input file: `Hurricane_Simulation_GulfCoast.yaml`.
 - Includes simulated facility downtimes, road closures, energy cost spikes.
 - **Context:** Routine test of agent’s crisis planning model.
 - **Flow:**
 - Agent runs “what-if” chain, suggests inventory reallocation from inland hubs.
 - Employee debriefs model behavior, logs improvements to future automation.
-

\$ Parent 3: Dynamic Pricing & Promotion Analysis

Conversation 3.1 — “Price Elasticity Drift Check”

- **Initiator:** Employee
- **Data:** `Price_Elasticity_Model_v5.pkl + Promo_History_Jan-Apr.csv`.
 - Promo history: `Date, SKU, Discount%, Units_Sold, Competitor_Price`.
- **Context:** Analyst suspects price sensitivity is changing post-inflation.
- **Flow:**
 - Agent re-fits elasticity curves and flags items where cross-price effect flipped sign.

- Leads to Conversation 3.2 for testing targeted promotions.
-

Conversation 3.2 — “Targeted Discount Pilot”

- **Initiator:** AI Agent
 - **Data:** Prior elasticity outputs + customer segmentation table.
 - **Context:** Agent proposes A/B testing a 5 % discount on low-income geographies.
 - **Flow:**
 - Employee asks for revenue neutrality projections.
 - Agent simulates outcomes under varying redemption rates; outputs dashboard.
-

Conversation 3.3 — “Competitor Price Shock Review”

- **Initiator:** Employee
 - **Data:** `Market_Intel_DailyFeed.json` (competitor scraping).
 - **Context:** Competitor abruptly drops prices on select SKUs.
 - **Flow:**
 - Agent models expected cannibalization and suggests short-term promotions or bundle offerings.
 - Conversation links back to inventory predictions from Scenario 1 to verify supply sufficiency.
-



Parent 5: Vendor & Supplier Performance Monitoring

Conversation 5.1 — “Chronic Delay Investigation”

- **Initiator:** Employee
 - **Data:** [Vendor_Scorecard_Annual.xlsx](#)
 - Columns: [Vendor_ID](#), [Avg_Lead_Time](#), [Delay%](#), [Return_Rate](#), [Quality_Score](#).
 - **Context:** Employee notices persistent late shipments from Vendor #342.
 - **Flow:**
 - Agent correlates delay with regional weather and driver shortage data.
 - Recommends split-lane sourcing and automatic penalty computation.
-

Conversation 5.2 — “Automated Scorecard Generation”

- **Initiator:** AI Agent
 - **Data:** Warehouse telemetry + ERP extracts.
 - **Context:** Monthly summary automation task.
 - **Flow:**
 - Agent emails procurement heads the top 5 improving and top 5 declining vendors.
 - Employee questions why Vendor #219 dropped sharply; agent traces to packaging defect logs.
-

Conversation 5.3 — “Supplier Carbon Footprint Audit”

- **Initiator:** Employee
- **Data:** [Sustainability_2025_Targets.docx](#) + [Supplier_CO2_Emissions.csv](#).

- **Context:** Corporate sustainability team evaluates vendor compliance.
 - **Flow:**
 - Agent aggregates CO₂ per SKU-mile, highlights non-compliant partners.
 - Leads to discussion about substituting greener regional suppliers — cross-links to Scenario 2.2.
-

Structural Observations

- **Total conversations:** 12
- **Conversations referencing other threads:** ≈ 5 (demonstrating temporal continuity).
- **Data variety:** structured CSVs (sales, suppliers), semi-structured JSON (feeds, simulations), documents (policy goals).
- **Agent roles:** diagnostic (“detect anomaly”), prescriptive (“simulate adjustment”), and generative (“draft report”).
- **Temporal links:** Scenarios 1 ↔ 2 (demand vs. logistics), 3 ↔ 1 (pricing vs. inventory), 5 ↔ 2 (supplier vs. routing).

PARENT 1:

Awesome — below is a **ready-to-paste Cypher bundle** that creates all nodes and relations for **Parent 1 (Demand Forecasting & Inventory Optimization)** covering the three conversations we defined earlier:

- **Conversation 1.1 — Regional Stockout Anomaly**
- **Conversation 1.2 — Holiday Demand Adjustment**
- **Conversation 1.3 — SKU Rationalization Proposal**

I also included a **Reasoning Bank** (as an external registry) with 2–3 entries per conversation and wired those IDs into `reasoning_pointer_ids` on key nodes. If you later want to persist the reasoning bank as actual nodes, we can map them to one of the six labels (e.g., fold into `AgentAction` as “Playbooks”) or create a side store and keep the pointers only.

Reasoning Bank (external registry; referenced by `reasoning_pointer_ids`)

/*

RB-P1-11A: “Forecast error triage should first segment by controllable vs. non-controllable drivers. Always check (a) data freshness/lateness in POS feeds, (b) unmodeled exogenous signals such as weather or local promos, (c) stockout censorship: realized demand < true demand when inventory hits zero; use lost-sales estimation (censored demand) before retraining.”

RB-P1-11B: “When heat or weather drives demand spikes, add a regional, lagged weather feature set: max temp, heat index, precipitation, and 7/14-day moving averages. Prefer hierarchical forecasting (SKU→category→store cluster) with reconciliation to avoid unstable SKU-level estimates.”

RB-P1-11C: “Before changing safety stock, simulate service-level impact with fill-rate targets; use quantile forecasts (P90) for top risk SKUs; otherwise you will overreact to outliers.”

RB-P1-12A: “Holiday uplift patterns are **shape + timing**. Reuse past-years’ seasonal shape as priors, then let current signals (ad spend, search interest, pay-period proximity) adjust amplitude. Enforce calendar effects with Fourier terms and event dummies.”

RB-P1-12B: “Always include **network constraints** in holiday recommendations: DC capacity, trailer availability, carrier cutoffs. If logistics can’t satisfy the uplift, adjust demand plan to feasible levels; don’t create orphan demand.”

RB-P1-12C: "Confidence bands matter to execs. Publish P10/P50/P90 and expected service level impact. Tie to financial deltas (GM\$, carry cost, markdown risk)."

RB-P1-13A: "SKU rationalization: use a composite score = (margin × demand stability) – (space cost × handling cost) + substitutability penalty. Protect 'destination' and 'trip-driver' SKUs even with low velocity."

RB-P1-13B: "Before delisting, run **switching analysis** (cross-elasticities). If high substitutability exists, inventory turns improve with minimal revenue loss; if not, delisting risks basket erosion."

RB-P1-13C: "Pilot first: pick 5–10 similar stores, A/B test delistings for 8–12 weeks; monitor sales mix, out-of-stocks on substitutes, shopper complaints; only then scale chain-wide."

*/

```
// =====
// Parent 1 — Conversation 1.1
// "Regional Stockout Anomaly"
// =====

// Nodes
MERGE (n:UserRequest {id: 'N2001'})
SET n.text = 'Analyst flagged unexpected stockouts for cold brew coffee in Southeast region; requests root-cause on forecast error vs. actual, with attention to weather and promotions; asks whether safety stock and reorder points should be adjusted.',
n.conv_id = '2025-11-12_WMT_P1_C11',
n.ingestion_time = '2025-11-12T15:05:00-05:00',
n.update_time = '2025-11-12T15:05:00-05:00',
n.embedding_id = 'emb_N2001',
n.tags =
["demand_forecasting", "stockout", "forecast_error", "southeast", "cold_brew", "safety_stock", "reorder_point", "weather", "promotion", "root_cause"],
n.reasoning_pointer_ids = ['RB-P1-11A', 'RB-P1-11B', 'RB-P1-11C'],
n.user_role = 'Supply Planning Analyst',
n.user_id = 'u_1248';

MERGE (n:DataSource {id: 'N2002'})
SET n.text = 'Sales_Week_42_Regional.csv — Store- and SKU-level weekly POS extracts including forecast vs. actual and on-hand inventory.',
n.conv_id = '2025-11-12_WMT_P1_C11',
n.ingestion_time = '2025-11-12T15:05:02-05:00',
n.update_time = '2025-11-12T15:05:02-05:00',
n.embedding_id = 'emb_N2002',
n.tags = ["pos", "sales", "inventory", "forecast_vs_actual", "sku", "store", "region", "csv"],
n.reasoning_pointer_ids = ['RB-P1-11A'],
n.source_type = 'csv',
n.doc_pointer = 's3://wmt/pos/Sales_Week_42_Regional.csv',
```

```
n.relevant_parts = 'columns: Store_ID, Region, SKU, Date, Units_Sold, Forecasted_Units,  
On_Hand, On_Order';
```

```
MERGE (n:DataSource {id: 'N2003'})  
SET n.text = 'Promo_History_Oct.csv — Local and chain-level promotions for beverages;  
discount %, feature/display flags.',  
n.conv_id = '2025-11-12_WMT_P1_C11',  
n.ingestion_time = '2025-11-12T15:05:03-05:00',  
n.update_time = '2025-11-12T15:05:03-05:00',  
n.embedding_id = 'emb_N2003',  
n.tags = ["promotion", "discount", "feature", "display", "beverages", "csv"],  
n.reasoning_pointer_ids = ['RB-P1-11A'],  
n.source_type = 'csv',  
n.doc_pointer = 's3://wmt/promos/Promo_History_Oct.csv',  
n.relevant_parts = 'columns: Date, Store_ID, SKU, DiscountPct, FeatureFlag, DisplayFlag';
```

```
MERGE (n:DataSource {id: 'N2004'})  
SET n.text = 'Weather_SE_2w.csv — Daily max temperature, heat index, precipitation by  
county for the Southeast; includes rolling means.',  
n.conv_id = '2025-11-12_WMT_P1_C11',  
n.ingestion_time = '2025-11-12T15:05:04-05:00',  
n.update_time = '2025-11-12T15:05:04-05:00',  
n.embedding_id = 'emb_N2004',  
n.tags = ["weather", "heat_index", "temperature", "precipitation", "regional", "csv"],  
n.reasoning_pointer_ids = ['RB-P1-11B'],  
n.source_type = 'csv',  
n.doc_pointer = 's3://wmt/external/weather/Weather_SE_2w.csv',  
n.relevant_parts = 'columns: County_FIPS, Date, MaxTemp, HeatIndex, Precip,  
MA7_MaxTemp, MA7_HeatIndex';
```

```
MERGE (n:Event {id: 'N2005'})  
SET n.text = 'Early-season heatwave in Southeast region (Oct 10–Oct 18) caused temperature  
anomalies vs. 5-year average; likely demand shock for cold beverages.',  
n.conv_id = '2025-11-12_WMT_P1_C11',  
n.ingestion_time = '2025-11-12T15:05:05-05:00',  
n.update_time = '2025-11-12T15:05:05-05:00',  
n.embedding_id = 'emb_N2005',  
n.tags = ["heatwave", "demand_shock", "seasonality", "weather_event", "southeast"],  
n.reasoning_pointer_ids = ['RB-P1-11B'],  
n.source_type = 'System Incident',  
n.start_date = '2025-10-10',  
n.end_date = '2025-10-18';
```

```
MERGE (n:AgentAction {id: 'N2006'})  
SET n.text = 'Censored-demand correction + feature augmentation. Steps: (1) Identify stockout
```

periods (On_Hand==0) and estimate lost sales via Bayesian shrinkage to category/store cluster; (2) Refit hierarchical demand model (SKU→category→cluster) with weather features (MaxTemp, HeatIndex, MA7) and promo dummies; (3) Recompute safety stock using P90 forecast and target fill rate 97%.'
 n.conv_id = '2025-11-12_WMT_P1_C11',
 n.ingestion_time = '2025-11-12T15:05:06-05:00',
 n.update_time = '2025-11-12T15:05:06-05:00',
 n.embedding_id = 'emb_N2006',
 n.tags =
 ["hierarchical_forecasting", "censored_demand", "lost_sales", "safety_stock", "feature_engineering", "weather_features"],
 n.reasoning_pointer_ids = ['RB-P1-11A','RB-P1-11B','RB-P1-11C'],
 n.status = 'success',
 n.parameter_field = 'fill_rate_target=0.97; hierarchy=SKU>Category>StoreCluster; features=[MaxTemp,HeatIndex,PromoFlags,MA7]';

 MERGE (n:AgentAnswer {id: 'N2007'})
 SET n.text = 'Root cause: (a) heatwave anomaly drove +18–26% uplift in iced coffee/cold brew; (b) local display promos compounded spike; (c) observed sales are censored by stockouts.
 Recommendation: increase safety stock for top 25 cold brew SKUs in SE by +12% (P90 basis), update reorder points, and schedule mid-week cross-dock transfers from surplus stores.',
 n.conv_id = '2025-11-12_WMT_P1_C11',
 n.ingestion_time = '2025-11-12T15:05:07-05:00',
 n.update_time = '2025-11-12T15:05:07-05:00',
 n.embedding_id = 'emb_N2007',
 n.tags =
 ["recommendation", "stockout_root_cause", "safety_stock_increase", "transfer", "demand_uplift", "forecasting"],
 n.reasoning_pointer_ids = ['RB-P1-11C'],
 n.analysis_types = ["root_cause", "hierarchical_forecast", "inventory_policy_update"],
 n.metrics = ["fill_rate", "stockout_rate", "lost_sales", "forecast_bias"];

 // Edges
 MATCH (a {id: 'N2001'}), (b {id: 'N2002'})
 MERGE (a)-[r:RELATES {id: 'E4001'}]->(b)
 SET r.text = 'Request prompts upload of weekly POS and inventory extracts to quantify forecast error and stockout censorship.',
 r.weight = 0.92,
 r.tags = ["trigger", "evidence", "pos", "inventory"],
 r.created_time = '2025-11-12T15:05:15-05:00';

 MATCH (a {id: 'N2003'}), (b {id: 'N2006'})
 MERGE (a)-[r:RELATES {id: 'E4002'}]->(b)
 SET r.text = 'Promotion history used to construct promo dummies and interaction terms in refit.',
 r.weight = 0.86,

```

r.tags = ["feature_engineering", "promotion", "model_refit"],
r.created_time = '2025-11-12T15:05:16-05:00';

MATCH (a {id: 'N2004'}), (b {id: 'N2006'})
MERGE (a)-[r:RELATES {id: 'E4003'}]->(b)
SET r.text = 'Weather feed provided heat index and moving averages for augmented demand model.',
r.weight = 0.88,
r.tags = ["weather", "features", "hierarchical_model"],
r.created_time = '2025-11-12T15:05:17-05:00';

MATCH (a {id: 'N2005'}), (b {id: 'N2006'})
MERGE (a)-[r:RELATES {id: 'E4004'}]->(b)
SET r.text = 'Heatwave event contextualizes anomalies; informs inclusion of exogenous regressors.',
r.weight = 0.84,
r.tags = ["context", "exogenous", "anomaly"],
r.created_time = '2025-11-12T15:05:18-05:00';

MATCH (a {id: 'N2006'}), (b {id: 'N2007'})
MERGE (a)-[r:RELATES {id: 'E4005'}]->(b)
SET r.text = 'Refitted model and censoring correction produce inventory policy recommendation and transfer plan.',
r.weight = 0.93,
r.tags = ["analysis_dependency", "inventory_policy", "recommendation"],
r.created_time = '2025-11-12T15:05:19-05:00';

// =====
// Parent 1 — Conversation 1.2
// "Holiday Demand Adjustment"
// =====

// Nodes
MERGE (n:AgentAction {id: 'N2011'})
SET n.text = 'Proactive holiday uplift detection. Scans last 3 years of pre-Thanksgiving windows for beverage, baking, and disposable serveware; learns seasonal shape and suggests DC pre-pulls and store-level allocation shifts.',
n.conv_id = '2025-11-12_WMT_P1_C12',
n.ingestion_time = '2025-11-12T15:06:10-05:00',
n.update_time = '2025-11-12T15:06:10-05:00',
n.embedding_id = 'emb_N2011',
n.tags =
["seasonality", "holiday", "uplift_detection", "allocation", "prepull", "beverages", "baking", "serveware"]
,
n.reasoning_pointer_ids = ['RB-P1-12A', 'RB-P1-12B', 'RB-P1-12C'],

```

```

n.status = 'success',
n.parameter_field = 'lookback_years=3; event_window=T-14..T+3;
priors=Fourier+event_dummies';

MERGE (n:DataSource {id: 'N2012'})
SET n.text = 'Demand_History_2019_2024.parquet — multi-year store/SKU demand with
calendar features and holiday flags.'
n.conv_id = '2025-11-12_WMT_P1_C12',
n.ingestion_time = '2025-11-12T15:06:11-05:00',
n.update_time = '2025-11-12T15:06:11-05:00',
n.embedding_id = 'emb_N2012',
n.tags = ["historical_demand","parquet","calendar","holiday_flags","store_sku"],
n.reasoning_pointer_ids = ['RB-P1-12A'],
n.source_type = 'parquet',
n.doc_pointer = 's3://wmt/demand/Demand_History_2019_2024.parquet',
n.relevant_parts = 'partition keys: year, month; features: holiday, week_of_year, pay_period';

MERGE (n:DataSource {id: 'N2013'})
SET n.text = 'Ops_Constraints_DC_Network.xlsx — distribution center capacity, trailer
constraints, carrier cutoffs, labor windows.'
n.conv_id = '2025-11-12_WMT_P1_C12',
n.ingestion_time = '2025-11-12T15:06:12-05:00',
n.update_time = '2025-11-12T15:06:12-05:00',
n.embedding_id = 'emb_N2013',
n.tags = ["constraints","dc_capacity","carrier_cutoff","ops","excel"],
n.reasoning_pointer_ids = ['RB-P1-12B'],
n.source_type = 'xlsx',
n.doc_pointer = 'sharepoint://ops/Ops_Constraints_DC_Network.xlsx',
n.relevant_parts = 'tabs: DC_capacity, Carrier_Cutoffs, Labor_Shifts';

MERGE (n:AgentAnswer {id: 'N2014'})
SET n.text = 'Recommendation: advance-ship top holiday SKUs (baking staples, canned
pumpkin, beverage multipacks) by +1.5 days to constrained DCs; store-level allocations
adjusted by historical uplift patterns; publish P10/P50/P90 demand bands with expected fill-rate
impact and GM$ deltas.'
n.conv_id = '2025-11-12_WMT_P1_C12',
n.ingestion_time = '2025-11-12T15:06:13-05:00',
n.update_time = '2025-11-12T15:06:13-05:00',
n.embedding_id = 'emb_N2014',
n.tags = ["holiday_recommendation","allocation","prepull","confidence_bands","financials"],
n.reasoning_pointer_ids = ['RB-P1-12C'],
n.analysis_types = ["seasonal_uplift","allocation_planning","capacity_feasibility"],
n.metrics = ["fill_rate","service_level","gross_margin","days_of_supply"];

```

```

MERGE (n:UserRequest {id: 'N2015'})
SET n.text = 'Merch lead requests confidence bands and explicit operations feasibility before approving allocation shifts; asks to quantify GM$ trade-offs.',
n.conv_id = '2025-11-12_WMT_P1_C12',
n.ingestion_time = '2025-11-12T15:06:14-05:00',
n.update_time = '2025-11-12T15:06:14-05:00',
n.embedding_id = 'emb_N2015',
n.tags =
["approval","confidence_intervals","ops_feasibility","financial_tradeoff","merchandising"],
n.reasoning_pointer_ids = ['RB-P1-12C'],
n.user_role = 'Merchandising Lead',
n.user_id = 'u_0911';

MERGE (n:Event {id: 'N2016'})
SET n.text = 'Thanksgiving 2025 seasonal prep window (Nov 13–Nov 27) requiring synchronized allocation and DC labor planning.',
n.conv_id = '2025-11-12_WMT_P1_C12',
n.ingestion_time = '2025-11-12T15:06:15-05:00',
n.update_time = '2025-11-12T15:06:15-05:00',
n.embedding_id = 'emb_N2016',
n.tags = ["thanksgiving","seasonality","allocation_window","labor_planning"],
n.reasoning_pointer_ids = ['RB-P1-12B','RB-P1-12C'],
n.source_type = 'Calendar',
n.start_date = '2025-11-13',
n.end_date = '2025-11-27';

// Edges
MATCH (a {id: 'N2012'}), (b {id: 'N2011'})
MERGE (a)-[r:RELATES {id: 'E4011'}]->(b)
SET r.text = 'Historical demand used to learn seasonal shape and amplitude priors.',
r.weight = 0.87,
r.tags = ["seasonality","priors","historical"],
r.created_time = '2025-11-12T15:06:20-05:00';

MATCH (a {id: 'N2013'}), (b {id: 'N2011'})
MERGE (a)-[r:RELATES {id: 'E4012'}]->(b)
SET r.text = 'Operational constraints bound feasible pre-pulls and allocation shifts.',
r.weight = 0.9,
r.tags = ["constraints","feasibility","ops"],
r.created_time = '2025-11-12T15:06:21-05:00';

MATCH (a {id: 'N2011'}), (b {id: 'N2014'})
MERGE (a)-[r:RELATES {id: 'E4013'}]->(b)
SET r.text = 'Proactive detection and feasibility check generate quantified recommendation with P10/P50/P90.'

```

```

r.weight = 0.93,
r.tags = ["analysis_dependency", "confidence_bands", "recommendation"],
r.created_time = '2025-11-12T15:06:22-05:00';

MATCH (a {id: 'N2015'}), (b {id: 'N2014'})
MERGE (a)-[r:RELATES {id: 'E4014'}]->(b)
SET r.text = 'Stakeholder approval contingent on bands and GM$; answer delivers both.',
r.weight = 0.82,
r.tags = ["approval", "finance", "confidence"],
r.created_time = '2025-11-12T15:06:23-05:00';

MATCH (a {id: 'N2016'}), (b {id: 'N2014'})
MERGE (a)-[r:RELATES {id: 'E4015'}]->(b)
SET r.text = 'Seasonal event window grounds timing of allocation moves.',
r.weight = 0.8,
r.tags = ["calendar", "timing", "seasonality"],
r.created_time = '2025-11-12T15:06:24-05:00';

// Cross-conversation bridge (1.1 → 1.2)
MATCH (a {id: 'N2007'}), (b {id: 'N2014'})
MERGE (a)-[r:RELATES {id: 'E4016'}]->(b)
SET r.text = 'Lessons from weather-driven uplift and safety-stock recalibration inform holiday
allocation strategy for beverage SKUs.',
r.weight = 0.76,
r.tags = ["knowledge_transfer", "uplift", "inventory_policy", "seasonality"],
r.created_time = '2025-11-12T15:06:25-05:00';

// =====
// Parent 1 — Conversation 1.3
// “SKU Rationalization Proposal”
// =====

// Nodes
MERGE (n:UserRequest {id: 'N2021'})
SET n.text = 'Category manager proposes rationalizing long-tail SKUs; requests an
evidence-based shortlist for delisting that preserves basket economics and store experience.',
n.conv_id = '2025-11-12_WMT_P1_C13',
n.ingestion_time = '2025-11-12T15:07:00-05:00',
n.update_time = '2025-11-12T15:07:00-05:00',
n.embedding_id = 'emb_N2021',
n.tags =
["sku_rationalization", "long_tail", "delisting", "basket_economics", "category_management"],
n.reasoning_pointer_ids = ['RB-P1-13A', 'RB-P1-13B', 'RB-P1-13C'],
n.user_role = 'Category Manager',
n.user_id = 'u_3371';

```

```
MERGE (n:DataSource {id: 'N2022'})  
SET n.text = 'LongTail_SKU_Performance.xlsx — velocity, margin, shelf space cost, handling  
cost, demand stability, substitutability score.',  
n.conv_id = '2025-11-12_WMT_P1_C13',  
n.ingestion_time = '2025-11-12T15:07:01-05:00',  
n.update_time = '2025-11-12T15:07:01-05:00',  
n.embedding_id = 'emb_N2022',  
n.tags = ["sku", "margin", "space_cost", "substitutability", "excel"],  
n.reasoning_pointer_ids = ['RB-P1-13A', 'RB-P1-13B'],  
n.source_type = 'xlsx',  
n.doc_pointer = 'sharepoint://category/LongTail_SKU_Performance.xlsx',  
n.relevant_parts = 'columns: SKU_ID, Avg_Weekly_Sales, MarginPct, SpaceCost,  
HandlingCost, CoV, Substitutability';
```

```
MERGE (n:AgentAction {id: 'N2023'})  
SET n.text = 'Composite rationalization scoring and switching analysis. Steps: (1) Build  
composite score = (margin × demand stability) – (space+handling) – (substitutability_penalty);  
(2) Estimate cross-elasticities using historical price/promo shocks; (3) Flag destination/trip-driver  
SKUs to protect; (4) Simulate store-level A/B pilots for 10 candidate delistings.',  
n.conv_id = '2025-11-12_WMT_P1_C13',  
n.ingestion_time = '2025-11-12T15:07:02-05:00',  
n.update_time = '2025-11-12T15:07:02-05:00',  
n.embedding_id = 'emb_N2023',  
n.tags = ["rationalization", "composite_score", "cross_elasticity", "pilot_design", "simulation"],  
n.reasoning_pointer_ids = ['RB-P1-13A', 'RB-P1-13B', 'RB-P1-13C'],  
n.status = 'success',  
n.parameter_field = 'pilot_weeks=10; stores=10; protect_tags=[destination,trip_driver]';
```

```
MERGE (n:AgentAnswer {id: 'N2024'})  
SET n.text = 'Shortlist: 12 SKUs recommended for pilot delisting; projected +6.8% category  
margin and +11% inventory turns with minimal revenue loss due to high substitutability.  
Guardrails: preserve 4 destination SKUs; monitor substitute stockouts and shopper feedback  
weekly.',  
n.conv_id = '2025-11-12_WMT_P1_C13',  
n.ingestion_time = '2025-11-12T15:07:03-05:00',  
n.update_time = '2025-11-12T15:07:03-05:00',  
n.embedding_id = 'emb_N2024',  
n.tags = ["recommendation", "sku_shortlist", "margin_lift", "inventory_turns", "pilot"],  
n.reasoning_pointer_ids = ['RB-P1-13C'],  
n.analysis_types = ["sku_scoring", "switching_simulation", "pilot_recommendation"],  
n.metrics = ["category_margin", "inventory_turns", "revenue_impact", "stockout_rate"];
```

```
MERGE (n:Event {id: 'N2025'})  
SET n.text = 'Pilot window set for Dec–Jan to avoid Black Friday confounds; includes weekly  
KPI readouts and halt conditions.'
```

```

n.conv_id = '2025-11-12_WMT_P1_C13',
n.ingestion_time = '2025-11-12T15:07:04-05:00',
n.update_time = '2025-11-12T15:07:04-05:00',
n.embedding_id = 'emb_N2025',
n.tags = ["pilot","readouts","halt_conditions","seasonality_control"],
n.reasoning_pointer_ids = ['RB-P1-13C'],
n.source_type = 'Calendar',
n.start_date = '2025-12-02',
n.end_date = '2026-01-31';

// Edges
MATCH (a {id: 'N2021'}), (b {id: 'N2022'})
MERGE (a)-[r:RELATES {id: 'E4021'}]->(b)
SET r.text = 'Rationalization request uses long-tail performance workbook as ground truth for scoring.',
r.weight = 0.89,
r.tags = ["evidence","sku","scoring_input"],
r.created_time = '2025-11-12T15:07:10-05:00';

MATCH (a {id: 'N2022'}), (b {id: 'N2023'})
MERGE (a)-[r:RELATES {id: 'E4022'}]->(b)
SET r.text = 'Composite score and switching analysis derived from margin, stability, cost, and substitutability.',
r.weight = 0.9,
r.tags = ["composite_score","switching","cross_elasticity"],
r.created_time = '2025-11-12T15:07:11-05:00';

MATCH (a {id: 'N2023'}), (b {id: 'N2024'})
MERGE (a)-[r:RELATES {id: 'E4023'}]->(b)
SET r.text = 'Simulation produces shortlist and KPI expectations for pilot delisting.',
r.weight = 0.94,
r.tags = ["simulation","pilot","recommendation"],
r.created_time = '2025-11-12T15:07:12-05:00';

MATCH (a {id: 'N2024'}), (b {id: 'N2025'})
MERGE (a)-[r:RELATES {id: 'E4024'}]->(b)
SET r.text = 'Recommendation bound to calendar pilot window with monitoring and halt criteria.',
r.weight = 0.82,
r.tags = ["calendar","governance","monitoring"],
r.created_time = '2025-11-12T15:07:13-05:00';

// Cross-conversation bridges (1.1/1.2 → 1.3)
MATCH (a {id: 'N2007'}), (b {id: 'N2023'})
MERGE (a)-[r:RELATES {id: 'E4025'}]->(b)

```

```

SET r.text = 'Insights on censored demand and safety stock inform risk controls in SKU
delisting pilots (avoid substitute stockouts).',
r.weight = 0.74,
r.tags = ["knowledge_transfer", "risk_control", "substitute_stockouts"],
r.created_time = '2025-11-12T15:07:14-05:00';

MATCH (a {id: 'N2014'}), (b {id: 'N2024'})
MERGE (a)-[r:RELATES {id: 'E4026'}]->(b)
SET r.text = 'Holiday allocation lessons shape timing and store selection for delisting pilots
(avoid confounded periods).',
r.weight = 0.7,
r.tags = ["timing", "pilot_design", "seasonality_control"],
r.created_time = '2025-11-12T15:07:15-05:00';

```

PARENT 2:

```

/* =====
REASONING BANK — Parent 2 (external registry, referenced only)
=====
```

RB-P2-21A: "Port disruptions propagate nonlinearly: the right heuristic is 'protect demand, protect flow.' First protect SKUs with (a) high holiday uplift, (b) low substitutability, (c) high penalty-of-lateness (lost GM\$, service-level SLAs). Then protect flow via multi-port routing (nearest uncongested port + rail/truck capacity check), not just 'fastest by ETA.'"

RB-P2-21B: "Delay ETA math: $\text{ETA}' = \text{Base_ETA} + \text{PortQueueDelay} + \text{Dwell} + \text{Drayage} + \text{Linehaul}$. Use percentiles from historical API feeds; prefer P75 or P90 in peak seasons. Re-run consolidation breaks (container mix) because partials inflate per-unit cost."

RB-P2-21C: "Before re-routing, verify downstream capacity constraints: DC receiving slots, yard space, carrier pickup windows, and labor shifts. A feasible slower route beats an infeasible 'fast' route that bottlenecks at the DC."

RB-P2-22A: "Reliability-adjusted landed cost beats unit cost alone. Score = UnitCost + RiskPremium(lead_time_variance, late_rate, quality_defect) - Rebate/TermBenefits. Optimize under service-level and capacity constraints."

RB-P2-22B: "Diversify by correlated risk. Two 'cheap' suppliers in the same risk basin (same port/weather/geo-politics) are not diversification. Penalize covariance of delay across suppliers."

RB-P2-22C: "When switching, audit tooling/qualification times and hidden NREs. Include ramp curves and first-article inspection failure probabilities; early scrap risk can erase nominal savings."

RB-P2-23A: "Crisis drills should be time-boxed and scored. KPIs: time-to-first-plan, fill-rate at top decile items, lane feasibility %, and human overrides required. Persist gaps as automation tasks."

RB-P2-23B: "During hurricanes, inventory reallocation favors inland hubs with resilient linehaul. Prefer hub-to-spoke repositioning with demand shielding; enforce fuel surcharge and driver HOS rules."

RB-P2-23C: "Write back learnings: encode playbooks as parameterized scenarios (facility_down, route_block, surge_demand). Tie them to monitors that auto-trigger the right playbook."

*/

```
// =====
// Parent 2 — Conversation 2.1
// "Port Congestion Alert" (Initiator: AI Agent)
// conv_id: 2025-11-12_WMT_P2_C21
// =====

// Event: Port congestion at Long Beach
MERGE (n:Event {id:'N2101'})
SET n.text = 'System-detected congestion at Port of Long Beach impacting inbound electronics and accessories; queue length up, berth delays + dwell inflation raising ETA uncertainty during pre-holiday peak.',
n.conv_id = '2025-11-12_WMT_P2_C21',
n.ingestion_time = '2025-11-12T16:05:00-05:00',
n.update_time = '2025-11-12T16:05:00-05:00',
n.embedding_id = 'emb_N2101',
n.tags =
['port','congestion','long_beach','eta','queue','holiday','electronics','dwell','linehaul','capacity'],
n.reasoning_pointer_ids = ['RB-P2-21A','RB-P2-21B','RB-P2-21C'],
n.source_type = 'System Incident',
n.start_date = '2025-11-10',
n.end_date = '2025-11-20';

// DataSource: External API feed
MERGE (n:DataSource {id:'N2102'})
SET n.text = 'Global_Shipping_Status_API.json — live port metrics (queue length, berth time), vessel ETAs, dwell statistics; historical percentiles for seasonal adjustment.',
n.conv_id = '2025-11-12_WMT_P2_C21',
n.ingestion_time = '2025-11-12T16:05:02-05:00',
n.update_time = '2025-11-12T16:05:02-05:00',
n.embedding_id = 'emb_N2102',
n.tags = ['api','port_status','eta','json','queue','dwell','historical'],
n.reasoning_pointer_ids = ['RB-P2-21B'],
n.source_type = 'json',
n.doc_pointer = 's3://feeds/Global_Shipping_Status_API.json',
n.relevant_parts = 'fields: port, queue_len, berth_delay_min, dwell_hours, vessel_id, eta_iso';
```

```

// DataSource: Internal supplier ETAs
MERGE (n:DataSource {id:'N2103'})
SET n.text = 'Supplier_ETA_Snapshot.parquet — PO line ETAs by supplier, origin, container, carrier; includes promised_date, incoterms, planned_port, and DC destination.',
    n.conv_id = '2025-11-12_WMT_P2_C21',
    n.ingestion_time = '2025-11-12T16:05:04-05:00',
    n.update_time = '2025-11-12T16:05:04-05:00',
    n.embedding_id = 'emb_N2103',
    n.tags = ['supplier', 'eta', 'po', 'carrier', 'parquet', 'incoterms', 'dc'],
    n.reasoning_pointer_ids = ['RB-P2-21A'],
    n.source_type = 'parquet',
    n.doc_pointer = 's3://erp/etl/Supplier_ETA_Snapshot.parquet',
    n.relevant_parts = 'columns: po_id, supplier_id, origin, port, container_id, carrier, promised_date, eta, dc_dest';

// AgentAction: Triage + re-routing feasibility
MERGE (n:AgentAction {id:'N2104'})
SET n.text = 'Triage congestion: join API port delays with supplier ETAs; recompute ETA with seasonal P75/P90; run multi-port re-routing feasibility (Oakland/Seattle) subject to DC receiving slots, yard space, linehaul capacity, and labor; recompute cost deltas per container.',
    n.conv_id = '2025-11-12_WMT_P2_C21',
    n.ingestion_time = '2025-11-12T16:05:08-05:00',
    n.update_time = '2025-11-12T16:05:08-05:00',
    n.embedding_id = 'emb_N2104',
    n.tags = ['triage', 'reroute', 'optimization', 'eta_model', 'capacity', 'linehaul', 'dc_slots', 'cost_delta'],
    n.reasoning_pointer_ids = ['RB-P2-21A', 'RB-P2-21B', 'RB-P2-21C'],
    n.status = 'complete',
    n.parameter_field = '{ "eta_quantile": "P75", "alt_ports": ["OAK", "SEA"], "constraints": ["DC_slots", "yard_space", "linehaul_capacity", "labor_shifts"] }';

// AgentAnswer: Recommendation summary
MERGE (n:AgentAnswer {id:'N2105'})
SET n.text = 'Recommendation: Re-route 28 containers (electronics, chargers, accessories) to OAK/SEA; protect low-substitutability SKUs with high holiday uplift; ETA improvement median +2.5 days vs staying at LGB during the window; capacity checks pass for DC-07/08 receiving and linehaul lanes. Estimated GM$ protected: $1.8M; incremental transport cost +$120k.',
    n.conv_id = '2025-11-12_WMT_P2_C21',
    n.ingestion_time = '2025-11-12T16:05:12-05:00',
    n.update_time = '2025-11-12T16:05:12-05:00',
    n.embedding_id = 'emb_N2105',
    n.tags =
['recommendation', 'reroute', 'holiday', 'low_substitutability', 'gm_protection', 'eta_gain', 'capacity_ok', 'cost_delta'],

```

```

n.reasoning_pointer_ids = ['RB-P2-21A','RB-P2-21C'],
n.analysis_types = ['routing_optimization','eta_recalculation','capacity_validation'],
n.metrics = ['eta_days_saved','gm_dollars_protected','incremental_transport_cost'];

// Edges (2.1)
MATCH (e:Event {id:'N2101'}),(d1:DataSource {id:'N2102'})
MERGE (e)-[r:RELATES {id:'E2401'}]->(d1)
SET r.text='Congestion event consumes external port/ETA feed as primary situational evidence.',
    r.weight=0.86, r.tags=['evidence','port','api'], r.created_time='2025-11-12T16:05:20-05:00';

MATCH (e:Event {id:'N2101'}),(d2:DataSource {id:'N2103'})
MERGE (e)-[r:RELATES {id:'E2402'}]->(d2)
SET r.text='Event links to internal supplier ETA snapshot to measure downstream impact by PO/container.',
    r.weight=0.84, r.tags=['impact','supplier','eta'], r.created_time='2025-11-12T16:05:22-05:00';

MATCH (d1:DataSource {id:'N2102'}),(a:AgentAction {id:'N2104'})
MERGE (d1)-[r:RELATES {id:'E2403'}]->(a)
SET r.text='External feed provides queue/dwell inputs to ETA recomputation and routing feasibility.',
    r.weight=0.88, r.tags=['input','eta_model','feasibility'],
    r.created_time='2025-11-12T16:05:25-05:00';

MATCH (d2:DataSource {id:'N2103'}),(a:AgentAction {id:'N2104'})
MERGE (d2)-[r:RELATES {id:'E2404'}]->(a)
SET r.text='PO-level ETAs and destinations join to capacity constraints (DC slots, linehaul).',
    r.weight=0.87, r.tags=['join','constraints','routing'],
    r.created_time='2025-11-12T16:05:27-05:00';

MATCH (e:Event {id:'N2101'}),(a:AgentAction {id:'N2104'})
MERGE (e)-[r:RELATES {id:'E2405'}]->(a)
SET r.text='Port congestion triggers triage and alternative-routing optimization.',
    r.weight=0.93, r.tags=['trigger','optimization','triage'],
    r.created_time='2025-11-12T16:05:30-05:00';

MATCH (a:AgentAction {id:'N2104'}),(ans:AgentAnswer {id:'N2105'})
MERGE (a)-[r:RELATES {id:'E2406'}]->(ans)
SET r.text='Feasibility + ETA model outputs roll up into re-routing recommendation with GM$ impact.',
    r.weight=0.94, r.tags=['solution','recommendation','gm'],
    r.created_time='2025-11-12T16:05:33-05:00';

MATCH (ans:AgentAnswer {id:'N2105'}),(e:Event {id:'N2101'})

```

```

MERGE (ans)-[r:RELATES {id:'E2407'}]->(e)
SET r.text='Recommendation explicitly addresses the congestion event timeframe with quantified impacts.',
r.weight=0.78, r.tags=['addresses','timebound','impact'],
r.created_time='2025-11-12T16:05:36-05:00';

// =====
// Parent 2 — Conversation 2.2
// "Alternate Supplier Simulation" (Initiator: Employee)
// conv_id: 2025-11-12_WMT_P2_C22
// =====

// UserRequest: Evaluate replacement suppliers
MERGE (n:UserRequest {id:'N2201'})
SET n.text = 'Employee requests reliability-adjusted cost simulation to replace chronically late suppliers; asks for risk score heatmap and service-level implications under DC capacity constraints.',
n.conv_id = '2025-11-12_WMT_P2_C22',
n.ingestion_time = '2025-11-12T16:12:00-05:00',
n.update_time = '2025-11-12T16:12:00-05:00',
n.embedding_id = 'emb_N2201',
n.tags = ['supplier','simulation','reliability','risk_heatmap','service_level','capacity'],
n.reasoning_pointer_ids = ['RB-P2-22A','RB-P2-22B','RB-P2-22C'],
n.user_role = 'Procurement Analyst',
n.user_id = 'u_3197';

// DataSource: Supplier scorecard extract
MERGE (n:DataSource {id:'N2202'})
SET n.text = 'Supplier_Performance_Q3.csv — Supplier_ID, Product_Line, Lead_Time, Unit_Cost, On_Time_Rate, Defect_Rate, Region; includes historical variance and rebate terms.',
n.conv_id = '2025-11-12_WMT_P2_C22',
n.ingestion_time = '2025-11-12T16:12:03-05:00',
n.update_time = '2025-11-12T16:12:03-05:00',
n.embedding_id = 'emb_N2202',
n.tags = ['supplier','scorecard','lead_time','unit_cost','on_time','defect','csv'],
n.reasoning_pointer_ids = ['RB-P2-22A'],
n.source_type = 'csv',
n.doc_pointer = 's3://procurement/Supplier_Performance_Q3.csv',
n.relevant_parts = 'fields: Supplier_ID, Product_Line, Lead_Time, Unit_Cost, On_Time_Rate, Lead_Time_SD, Defect_Rate, Region, RebateTerms';

// AgentAction: Optimization

```

```

MERGE (n:AgentAction {id:'N2203'})
SET n.text = 'Formulate reliability-adjusted landed cost optimization with service-level ≥ 95%, DC capacity, and regional risk covariance penalties; solve mixed-integer program for sourcing split.'
    n.conv_id = '2025-11-12_WMT_P2_C22',
    n.ingestion_time = '2025-11-12T16:12:08-05:00',
    n.update_time = '2025-11-12T16:12:08-05:00',
    n.embedding_id = 'emb_N2203',
    n.tags = ['optimization','mip','service_level','covariance_penalty','sourcing_split'],
    n.reasoning_pointer_ids = ['RB-P2-22A','RB-P2-22B'],
    n.status = 'complete',
    n.parameter_field = '{ "target_service_level":0.95, "risk_penalty":"cov(delay)", "cap_constraints":true, "min_alloc_per_supplier":0.1 }';

// AgentAnswer: Recommendation
MERGE (n:AgentAnswer {id:'N2204'})
SET n.text = 'Recommendation: Shift 35–45% of volume from Supplier #342 to #219 and #455; expected late_rate drops from 18%→9% with marginal landed cost +$0.04/unit; service-level improves to 96.2%. Risk heatmap shows #219 and #455 are uncorrelated on delay shocks (different ports/weather basins). Include 3-week ramp for tooling qualification.'
    n.conv_id = '2025-11-12_WMT_P2_C22',
    n.ingestion_time = '2025-11-12T16:12:12-05:00',
    n.update_time = '2025-11-12T16:12:12-05:00',
    n.embedding_id = 'emb_N2204',
    n.tags =
['recommendation','supplier_switch','late_rate','landed_cost','service_level','diversification','qualification'],
    n.reasoning_pointer_ids = ['RB-P2-22A','RB-P2-22B','RB-P2-22C'],
    n.analysis_types = ['mip_optimization','risk_heatmap','scenario_analysis'],
    n.metrics = ['late_rate','service_level','landed_cost_delta'];

// Edges (2.2)
MATCH (u:UserRequest {id:'N2201'}),(d:DataSource {id:'N2202'})
MERGE (u)-[r:RELATES {id:'E2411'}]->(d)
SET r.text='Request anchors to Q3 supplier scorecard as the quantitative evidence base.',
    r.weight=0.83, r.tags=['evidence','scorecard'], r.created_time='2025-11-12T16:12:20-05:00';

MATCH (d:DataSource {id:'N2202'}),(a:AgentAction {id:'N2203'})
MERGE (d)-[r:RELATES {id:'E2412'}]->(a)
SET r.text='Scorecard fields parameterize reliability-adjusted cost and delay variance inputs.',
    r.weight=0.87, r.tags=['input','variance','reliability'],
    r.created_time='2025-11-12T16:12:22-05:00';

MATCH (u:UserRequest {id:'N2201'}),(a:AgentAction {id:'N2203'})

```

```

MERGE (u)-[r:RELATES {id:'E2413'}]->(a)
SET r.text='User request triggers MIP solve with service-level and capacity constraints.',
r.weight=0.92, r.tags=['trigger','optimization'], r.created_time='2025-11-12T16:12:25-05:00';

MATCH (a:AgentAction {id:'N2203'}),(ans:AgentAnswer {id:'N2204'})
MERGE (a)-[r:RELATES {id:'E2414'}]->(ans)
SET r.text='Optimization produces sourcing split, risk diversification evidence, and ramp
guidance.',
r.weight=0.95, r.tags=['solution','sourcing','diversification'],
r.created_time='2025-11-12T16:12:28-05:00';

// Cross-link: 2.1 recommendation informs 2.2 simulation scope (topical bridge)
MATCH (p21:AgentAnswer {id:'N2105'}),(p22:UserRequest {id:'N2201'})
MERGE (p21)-[r:RELATES {id:'E2415'}]->(p22)
SET r.text='Re-routing plan and protected SKU set from congestion alert inform supplier
simulation priorities.',
r.weight=0.62, r.tags=['context_bridge','prioritization','cross_thread'],
r.created_time='2025-11-12T16:12:31-05:00';

// =====
// Parent 2 — Conversation 2.3
// "Disaster Response Playbook Drill" (Initiator: AI Agent)
// conv_id: 2025-11-12_WMT_P2_C23
// =====

// Event: Scheduled drill
MERGE (n:Event {id:'N2301'})
SET n.text = 'Quarterly disaster response drill: Gulf Coast hurricane scenario with simulated
facility downtime, road closures, fuel surcharges, and grid instability; objective is readiness
scoring and automation gap capture.',
n.conv_id = '2025-11-12_WMT_P2_C23',
n.ingestion_time = '2025-11-12T16:20:00-05:00',
n.update_time = '2025-11-12T16:20:00-05:00',
n.embedding_id = 'emb_N2301',
n.tags = ['drill','hurricane','downtime','closures','fuel_surcharge','readiness','automation_gap'],
n.reasoning_pointer_ids = ['RB-P2-23A','RB-P2-23B','RB-P2-23C'],
n.source_type = 'Calendar',
n.start_date = '2025-11-12',
n.end_date = '2025-11-12';

// DataSource: Synthetic scenario file
MERGE (n:DataSource {id:'N2302'})

```

```

SET n.text = 'Hurricane_Simulation_GulfCoast.yaml — facility_down% per DC, lane closures, energy price spikes, demand surge multipliers; includes time-phased parameters and recovery curves.',
n.conv_id = '2025-11-12_WMT_P2_C23',
n.ingestion_time = '2025-11-12T16:20:03-05:00',
n.update_time = '2025-11-12T16:20:03-05:00',
n.embedding_id = 'emb_N2302',
n.tags = ['yaml','simulation','facility_down','lane_closure','demand_surge','recovery'],
n.reasoning_pointer_ids = ['RB-P2-23B'],
n.source_type = 'yaml',
n.doc_pointer = 's3://sims/Hurricane_Simulation_GulfCoast.yaml',
n.relevant_parts = 'sections: facilities, lanes, energy, demand, recovery_curves';

// AgentAction: What-if chain + reallocation
MERGE (n:AgentAction {id:'N2303'})
SET n.text = 'Execute what-if chain: disable affected coastal DCs, re-route via inland hubs; recompute feasible lanes under HOS and fuel surcharge; shield top-decile demand; output playbook steps + automation tasks for gaps.',
n.conv_id = '2025-11-12_WMT_P2_C23',
n.ingestion_time = '2025-11-12T16:20:08-05:00',
n.update_time = '2025-11-12T16:20:08-05:00',
n.embedding_id = 'emb_N2303',
n.tags = ['what_if','reallocation','lane_feasibility','hos_rules','playbook','automation_tasks'],
n.reasoning_pointer_ids = ['RB-P2-23A','RB-P2-23B','RB-P2-23C'],
n.status = 'complete',
n.parameter_field = '{ "shield": "top_decile_items", "hos": true, "fuel_surcharge": true,
"hub": ["DC-12", "DC-14"] }';

// AgentAnswer: Drill outcomes + learnings
MERGE (n:AgentAnswer {id:'N2304'})
SET n.text = 'Drill results: maintain 93.4% fill-rate on protected SKUs; time-to-first-plan 11m; feasible lanes 78% with HOS & surcharge; flagged two automation gaps (dynamic yard slotting; carrier swap rules). Playbook updated and tasks logged.',
n.conv_id = '2025-11-12_WMT_P2_C23',
n.ingestion_time = '2025-11-12T16:20:12-05:00',
n.update_time = '2025-11-12T16:20:12-05:00',
n.embedding_id = 'emb_N2304',
n.tags = ['drill_results','fill_rate','time_to_plan','feasibility','automation_gap','playbook_update'],
n.reasoning_pointer_ids = ['RB-P2-23A','RB-P2-23C'],
n.analysis_types = ['scenario_planning','resilience_assessment','playbook_update'],
n.metrics = ['fill_rate','time_to_first_plan','lane_feasibility_pct'];

// Edges (2.3)
MATCH (ev:Event {id:'N2301'}),(ds:DataSource {id:'N2302'})

```

```

MERGE (ev)-[r:RELATES {id:'E2421'}]->(ds)
SET r.text='Scheduled drill consumes the synthetic hurricane scenario file as the driver of
conditions.',
    r.weight=0.85, r.tags=['scenario','driver','evidence'],
    r.created_time='2025-11-12T16:20:20-05:00';

MATCH (ds:DataSource {id:'N2302'}),(aa:AgentAction {id:'N2303'})
MERGE (ds)-[r:RELATES {id:'E2422'}]->(aa)
SET r.text='Scenario parameters feed what-if execution and hub reallocation optimization.',
    r.weight=0.88, r.tags=['input','what_if','reallocation'],
    r.created_time='2025-11-12T16:20:23-05:00';

MATCH (ev:Event {id:'N2301'}),(aa:AgentAction {id:'N2303'})
MERGE (ev)-[r:RELATES {id:'E2423'}]->(aa)
SET r.text='The drill event triggers the automated playbook run and gap capture.',
    r.weight=0.93, r.tags=['trigger','playbook','automation'],
    r.created_time='2025-11-12T16:20:26-05:00';

MATCH (aa:AgentAction {id:'N2303'}),(ans:AgentAnswer {id:'N2304'})
MERGE (aa)-[r:RELATES {id:'E2424'}]->(ans)
SET r.text='What-if outputs translate into quantified resilience KPIs and playbook updates.',
    r.weight=0.95, r.tags=['solution','kpi','update'], r.created_time='2025-11-12T16:20:29-05:00';

// Cross-links inside Parent 2 for temporal continuity
MATCH (c21:AgentAnswer {id:'N2105'}),(c23:AgentAction {id:'N2303'})
MERGE (c21)-[r:RELATES {id:'E2425'}]->(c23)
SET r.text='Routing preferences and capacity constraints learned during congestion alert inform
drill routing heuristics.',
    r.weight=0.66, r.tags=['heuristic','capacity','knowledge_transfer'],
    r.created_time='2025-11-12T16:20:33-05:00';

MATCH (c22:AgentAnswer {id:'N2204'}),(c23:AgentAnswer {id:'N2304'})
MERGE (c22)-[r:RELATES {id:'E2426'}]->(c23)
SET r.text='Supplier diversification plan improves drill resilience metrics by reducing correlated
delay exposure.',
    r.weight=0.61, r.tags=['resilience','diversification','cross_thread'],
    r.created_time='2025-11-12T16:20:36-05:00';

```

Parent 3:

```
/* =====  
REASONING BANK — Parent 3 (external registry, referenced only)  
=====
```

RB-P3-31A: "Elasticity drift detection: re-fit with rolling windows and stability diagnostics (Hausman-type test, sign consistency, leverage/influence). Track cross-price effects vs key rivals; guard against promo leakage and post-promo dip bias."

RB-P3-31B: "Prefer hierarchical shrinkage (SKU → subcategory → category) and partial pooling for sparse promo histories; add price floors/ceilings to avoid unstable slopes on low-volume tails."

RB-P3-31C: "Report elasticity with confidence bands and business interpretation: items with flipped sign or high variance require controlled pilots, not chain-wide price moves."

RB-P3-32A: "Targeted discounts must be revenue-neutral or accretive: simulate redemption rate ranges, halo/cannibalization, and stock constraints; compute EV[GM\$] under uncertain lift."

RB-P3-32B: "Segment by price sensitivity + income proxy + basket composition; avoid discounting trip-drivers where full-price integrity matters; couple with inventory sufficiency checks."

RB-P3-32C: "Pilot design: randomization at store-cluster level, 2–4 week window, holdout control, pre-trend checks, CUPED or fixed-effects regressions to tighten variance; predefined stop/scale rules."

RB-P3-33A: "Competitor shocks: react with bounded, time-limited actions (temporary discounts, bundles, loyalty credits) prioritized by cross-price elasticity and inventory headroom."

RB-P3-33B: "Detect asymmetric responses: if own-price elasticity steepened while cross-price rose, expect outsized share loss on leader SKUs; use guardrails to prevent margin collapse."

RB-P3-33C: "Comms & reversion: set clear end dates and automated rollback to avoid price shadowing spirals; monitor competitor reversion probability from historical patterns."

*/

```
// =====  
// Parent 3 — Conversation 3.1  
// "Price Elasticity Drift Check" (Initiator: Employee)  
// conv_id: 2025-11-12_WMT_P3_C31  
// =====
```

```
// UserRequest  
MERGE (n:UserRequest {id:'N3101'})  
SET n.text = 'Analyst suspects elasticity drift post-inflation; requests re-fit of elasticity curves, cross-price checks vs. competitor, and diagnostic report on sign flips/variance.',  
n.conv_id = '2025-11-12_WMT_P3_C31',  
n.ingestion_time = '2025-11-12T16:35:00-05:00',
```

```

n.update_time = '2025-11-12T16:35:00-05:00',
n.embedding_id = 'emb_N3101',
n.tags = ['elasticity','drift','post_inflation','diagnostics','cross_price','competitor'],
n.reasoning_pointer_ids = ['RB-P3-31A','RB-P3-31B','RB-P3-31C'],
n.user_role = 'Pricing Analyst',
n.user_id = 'u_2041';

// DataSource: Model object
MERGE (n:DataSource {id:'N3102'})
SET n.text = 'Price_Elasticity_Model_v5.pkl — prior fitted model artifacts (coefficients, priors, shrinkage parameters, SKU hierarchy).',
n.conv_id = '2025-11-12_WMT_P3_C31',
n.ingestion_time = '2025-11-12T16:35:03-05:00',
n.update_time = '2025-11-12T16:35:03-05:00',
n.embedding_id = 'emb_N3102',
n.tags = ['model','pickle','elasticity','hierarchical','priors'],
n.reasoning_pointer_ids = ['RB-P3-31B'],
n.source_type = 'pkl',
n.doc_pointer = 's3://pricing/models/Price_Elasticity_Model_v5.pkl',
n.relevant_parts = 'objects: coef_table, sku_map, shrinkage_priors, fit_summary';

// DataSource: Promo history
MERGE (n:DataSource {id:'N3103'})
SET n.text = 'Promo_History_Jan-Apr.csv — Date, SKU, DiscountPct, Units_Sold, BasePrice, CompetitorPrice, Channel; includes flags for promo overlap and post-promo dips.',
n.conv_id = '2025-11-12_WMT_P3_C31',
n.ingestion_time = '2025-11-12T16:35:05-05:00',
n.update_time = '2025-11-12T16:35:05-05:00',
n.embedding_id = 'emb_N3103',
n.tags = ['promo','history','csv','competitor_price','overlap','post_promo_dip'],
n.reasoning_pointer_ids = ['RB-P3-31A'],
n.source_type = 'csv',
n.doc_pointer = 's3://pricing/history/Promo_History_Jan-Apr.csv',
n.relevant_parts = 'columns: date, sku, price, discount_pct, competitor_price, units, channel, overlap_flag, post_dip_flag';

// AgentAction: Re-fit + diagnostics
MERGE (n:AgentAction {id:'N3104'})
SET n.text = 'Re-fit hierarchical elasticity with rolling windows; run sign-flip scan, influence diagnostics, cross-price stability vs. top competitor; produce SKU/subcategory variance report.',
n.conv_id = '2025-11-12_WMT_P3_C31',
n.ingestion_time = '2025-11-12T16:35:10-05:00',
n.update_time = '2025-11-12T16:35:10-05:00',
n.embedding_id = 'emb_N3104',

```

```

n.tags = ['refit','diagnostics','hierarchical','cross_price','variance'],
n.reasoning_pointer_ids = ['RB-P3-31A','RB-P3-31B'],
n.status = 'complete',
n.parameter_field = '{ "window":"8w_rolling", "hierarchy":["sku","subcategory","category"],
"diag":["sign_flip","influence","hausman"], "competitor":"TopRival" }';

// AgentAnswer: Findings
MERGE (n:AgentAnswer {id:'N3105'})
SET n.text = 'Findings: 37 SKUs show steepened own-price elasticity; 11 SKUs flipped cross-price sign vs. TopRival. High-variance tails clustered in low-volume SKUs with sparse promo history. Recommendation: hold prices on trip-drivers; schedule controlled pilots for 12 flagged SKUs before broad repricing.',
n.conv_id = '2025-11-12_WMT_P3_C31',
n.ingestion_time = '2025-11-12T16:35:14-05:00',
n.update_time = '2025-11-12T16:35:14-05:00',
n.embedding_id = 'emb_N3105',
n.tags = ['findings','elasticity_drift','sign_flip','pilot_recommendation','trip_drivers'],
n.reasoning_pointer_ids = ['RB-P3-31C'],
n.analysis_types = ['hierarchical_regression','drift_detection','competitor_analysis'],
n.metrics = ['num_skus_steeper','num_sign_flips','variance_cluster_count'];

// Edges (3.1)
MATCH (u:UserRequest {id:'N3101'}),(d1:DataSource {id:'N3102'})
MERGE (u)-[r:RELATES {id:'E3411'}]->(d1)
SET r.text='Request references prior elasticity model artifacts as baseline for comparison.',
r.weight=0.82, r.tags=['baseline','model_artifact'],
r.created_time='2025-11-12T16:35:20-05:00';

MATCH (u:UserRequest {id:'N3101'}),(d2:DataSource {id:'N3103'})
MERGE (u)-[r:RELATES {id:'E3412'}]->(d2)
SET r.text='Promo history provides event-level signals for re-fit and drift attribution.',
r.weight=0.86, r.tags=['evidence','promo_events'],
r.created_time='2025-11-12T16:35:22-05:00';

MATCH (d1:DataSource {id:'N3102'}),(a:AgentAction {id:'N3104'})
MERGE (d1)-[r:RELATES {id:'E3413'}]->(a)
SET r.text='Model priors and hierarchy inform the re-fit and shrinkage behavior.',
r.weight=0.88, r.tags=['input','priors','hierarchy'], r.created_time='2025-11-12T16:35:25-05:00';

MATCH (d2:DataSource {id:'N3103'}),(a:AgentAction {id:'N3104'})
MERGE (d2)-[r:RELATES {id:'E3414'}]->(a)
SET r.text='Promo events and competitor prices parameterize drift detection diagnostics.',
r.weight=0.89, r.tags=['input','diagnostics'], r.created_time='2025-11-12T16:35:27-05:00';

```

```

MATCH (a:AgentAction {id:'N3104'}),(ans:AgentAnswer {id:'N3105'})
MERGE (a)-[r:RELATES {id:'E3415'}]->(ans)
SET r.text='Diagnostic outputs roll up into a pilot-first pricing recommendation.',
r.weight=0.94, r.tags=['solution','pilot'], r.created_time='2025-11-12T16:35:30-05:00';

// =====
// Parent 3 — Conversation 3.2
// "Targeted Discount Pilot" (Initiator: AI Agent)
// conv_id: 2025-11-12_WMT_P3_C32
// =====

// Event: Pilot window
MERGE (n:Event {id:'N3201'})
SET n.text = 'Targeted discount A/B pilot window with cluster-level randomization; objective: revenue-neutral lift in low-income geographies without stockouts.',
n.conv_id = '2025-11-12_WMT_P3_C32',
n.ingestion_time = '2025-11-12T16:43:00-05:00',
n.update_time = '2025-11-12T16:43:00-05:00',
n.embedding_id = 'emb_N3201',
n.tags = ['pilot','ab_test','randomization','low_income_geo','revenue_neutral'],
n.reasoning_pointer_ids = ['RB-P3-32C','RB-P3-32B'],
n.source_type = 'Calendar',
n.start_date = '2025-11-20',
n.end_date = '2025-12-04';

// DataSource: Segmentation table
MERGE (n:DataSource {id:'N3202'})
SET n.text = 'Customer_Segmentation_2025Q3.parquet — segments by inferred income proxy, price sensitivity, basket composition, trip frequency; store-cluster mapping.',
n.conv_id = '2025-11-12_WMT_P3_C32',
n.ingestion_time = '2025-11-12T16:43:03-05:00',
n.update_time = '2025-11-12T16:43:03-05:00',
n.embedding_id = 'emb_N3202',
n.tags = ['segmentation','price_sensitivity','basket','trip_frequency','parquet'],
n.reasoning_pointer_ids = ['RB-P3-32B'],
n.source_type = 'parquet',
n.doc_pointer = 's3://marketing/segmentation/Customer_Segmentation_2025Q3.parquet',
n.relevant_parts = 'columns: household_id, segment, income_proxy, price_sens, basket_mix, store_cluster';

// DataSource: Elasticity outputs (from 3.1)
MERGE (n:DataSource {id:'N3203'})

```

```

SET n.text = 'Elasticity_Refit_Results_C31.csv — per-SKU own- and cross-price elasticities with
confidence intervals; flags for sign flips and high-variance tails.',
    n.conv_id = '2025-11-12_WMT_P3_C32',
    n.ingestion_time = '2025-11-12T16:43:05-05:00',
    n.update_time = '2025-11-12T16:43:05-05:00',
    n.embedding_id = 'emb_N3203',
    n.tags = ['elasticity_outputs','confidence_bands','sign_flip','variance_flag','csv'],
    n.reasoning_pointer_ids = ['RB-P3-32A'],
    n.source_type = 'csv',
    n.doc_pointer = 's3://pricing/results/Elasticity_Refit_Results_C31.csv',
    n.relevant_parts = 'columns: sku, own_elasticity, cross_elasticity, ci_lower, ci_upper,
sign_flip_flag';

// AgentAction: Pilot design & sim
MERGE (n:AgentAction {id:'N3204'})
SET n.text = 'Design targeted 5% discount pilot on high-sensitivity segments; simulate
redemption/lift (P10–P90), halo/cannibalization, inventory sufficiency; predefined
CUPED/fixed-effects analysis and stop/scale rules.',
    n.conv_id = '2025-11-12_WMT_P3_C32',
    n.ingestion_time = '2025-11-12T16:43:10-05:00',
    n.update_time = '2025-11-12T16:43:10-05:00',
    n.embedding_id = 'emb_N3204',
    n.tags = ['pilot_design','simulation','revenue_neutrality','cuped','fixed_effects','guardrails'],
    n.reasoning_pointer_ids = ['RB-P3-32A','RB-P3-32C','RB-P3-32B'],
    n.status = 'complete',
    n.parameter_field = '{ "discount_pct":0.05, "analysis":["CUPED","FE"], "clusters":64,
"duration_days":14, "redemption_scenarios":["P10","P50","P90"] }';

// AgentAnswer: Pilot plan
MERGE (n:AgentAnswer {id:'N3205'})
SET n.text = 'Pilot plan: 5% discount in 32 treatment clusters (matched 32 control); expected lift
+6.8% (P50) with cannibalization <1.2%; revenue-neutral to +0.3% GM$ under inventory
guardrails. Stop if GM$ < -0.5% or stockouts >2%; scale if GM$ ≥ +0.5% and no service risk.',
    n.conv_id = '2025-11-12_WMT_P3_C32',
    n.ingestion_time = '2025-11-12T16:43:14-05:00',
    n.update_time = '2025-11-12T16:43:14-05:00',
    n.embedding_id = 'emb_N3205',
    n.tags = ['pilot_plan','guardrails','revenue_neutral','scale_rules','stockout_risk'],
    n.reasoning_pointer_ids = ['RB-P3-32A','RB-P3-32C'],
    n.analysis_types = ['ab_testing','uplift_simulation','guardrail_policy'],
    n.metrics = ['expected_lift','gm_delta','cannibalization_rate','stockout_threshold'];

// Edges (3.2)
MATCH (ev:Event {id:'N3201'}),(ds1:DataSource {id:'N3202'})

```

```

MERGE (ev)-[r:RELATES {id:'E3421'}]->(ds1)
SET r.text='Pilot timing aligns with segmentation table to target high-sensitivity clusters.',
    r.weight=0.83, r.tags=['targeting','timing'], r.created_time='2025-11-12T16:43:20-05:00';

MATCH (ds2:DataSource {id:'N3203'}),(aa:AgentAction {id:'N3204'})
MERGE (ds2)-[r:RELATES {id:'E3422'}]->(aa)
SET r.text='Elasticity outputs parameterize lift priors and cannibalization expectations.',
    r.weight=0.89, r.tags=['input','lift_priors'], r.created_time='2025-11-12T16:43:22-05:00';

MATCH (ev:Event {id:'N3201'}),(aa:AgentAction {id:'N3204'})
MERGE (ev)-[r:RELATES {id:'E3423'}]->(aa)
SET r.text='Pilot window triggers experimental design and simulation.',
    r.weight=0.91, r.tags=['trigger','experiment'], r.created_time='2025-11-12T16:43:25-05:00';

MATCH (aa:AgentAction {id:'N3204'}),(ans:AgentAnswer {id:'N3205'})
MERGE (aa)-[r:RELATES {id:'E3424'}]->(ans)
SET r.text='Design and simulation synthesize into a guardrailed pilot plan.',
    r.weight=0.95, r.tags=['solution','plan'], r.created_time='2025-11-12T16:43:28-05:00';

// Cross-link: 3.1 → 3.2
MATCH (c31:AgentAnswer {id:'N3105'}),(c32:AgentAction {id:'N3204'})
MERGE (c31)-[r:RELATES {id:'E3425'}]->(c32)
SET r.text='Elasticity drift findings determine which SKUs/segments enter the pilot.',
    r.weight=0.72, r.tags=['context_bridge','selection'],
    r.created_time='2025-11-12T16:43:31-05:00';

// =====
// Parent 3 — Conversation 3.3
// "Competitor Price Shock Review" (Initiator: Employee)
// conv_id: 2025-11-12_WMT_P3_C33
// =====

// Event: Competitor price drop
MERGE (n:Event {id:'N3301'})
SET n.text = 'Competitor abruptly drops prices on select leader SKUs (-7% median); risk of
share loss given recent elasticity drift.',
    n.conv_id = '2025-11-12_WMT_P3_C33',
    n.ingestion_time = '2025-11-12T16:51:00-05:00',
    n.update_time = '2025-11-12T16:51:00-05:00',
    n.embedding_id = 'emb_N3301',
    n.tags = ['competitor_shock','price_drop','leader_sku','share_loss_risk'],
    n.reasoning_pointer_ids = ['RB-P3-33A','RB-P3-33B','RB-P3-33C'],
    n.source_type = 'Market Intel',

```

```

n.start_date = '2025-11-12',
n.end_date = '2025-11-12';

// DataSource: Market intel feed
MERGE (n:DataSource {id:'N3302'})
SET n.text = 'Market_Intel_DailyFeed.json — competitor scraped prices with SKU mapping, promo flags, and historical reversion stats.',
n.conv_id = '2025-11-12_WMT_P3_C33',
n.ingestion_time = '2025-11-12T16:51:03-05:00',
n.update_time = '2025-11-12T16:51:03-05:00',
n.embedding_id = 'emb_N3302',
n.tags = ['market_intel','scrape','competitor','json','reversion'],
n.reasoning_pointer_ids = ['RB-P3-33C','RB-P3-33A'],
n.source_type = 'json',
n.doc_pointer = 's3://market/Market_Intel_DailyFeed.json',
n.relevant_parts = 'fields: competitor_sku, our_sku, price, promo_flag, asof, reversion_prob';

// AgentAction: Shock response modeling
MERGE (n:AgentAction {id:'N3303'})
SET n.text = 'Estimate demand impact using cross-price elasticities from C31; simulate response options (temporary discount, bundle, loyalty credit) under inventory headroom and margin guardrails; define rollback criteria.',
n.conv_id = '2025-11-12_WMT_P3_C33',
n.ingestion_time = '2025-11-12T16:51:08-05:00',
n.update_time = '2025-11-12T16:51:08-05:00',
n.embedding_id = 'emb_N3303',
n.tags = ['shock_model','cross_price','bundle','loyalty','rollback','margin_guardrail'],
n.reasoning_pointer_ids = ['RB-P3-33A','RB-P3-33B','RB-P3-33C'],
n.status = 'complete',
n.parameter_field = '{ "options": ["temp_discount", "bundle", "loyalty_credit"], "guardrails": {"gm_margin_min": 0.28, "inventory_headroom": true}, "rollback": "date_or_reversion_trigger" }';

// AgentAnswer: Action recommendation
MERGE (n:AgentAnswer {id:'N3304'})
SET n.text = 'Recommended play: 2-week bundle offer on leaders + loyalty credit ($3) in high-cross-price clusters; avoid broad price match. Expected share loss cut by ~45% vs. no action, GM% down only -20 bps; auto-rollback if competitor reverts or on 2025-11-26.',
n.conv_id = '2025-11-12_WMT_P3_C33',
n.ingestion_time = '2025-11-12T16:51:12-05:00',
n.update_time = '2025-11-12T16:51:12-05:00',
n.embedding_id = 'emb_N3304',
n.tags = ['recommendation','bundle','loyalty','rollback','share_loss','gm_impact'],
n.reasoning_pointer_ids = ['RB-P3-33A','RB-P3-33C'],

```

```

n.analysis_types = ['scenario_simulation','margin_impact','rollback_policy'],
n.metrics = ['share_loss_reduction','gm_bps_change','action_window_days'];

// Edges (3.3)
MATCH (ev:Event {id:'N3301'}),(ds:DataSource {id:'N3302'})
MERGE (ev)-[r:RELATES {id:'E3431'}]->(ds)
SET r.text='Competitor shock grounded in daily market intel feed.',
    r.weight=0.84, r.tags=['evidence','competitor'], r.created_time='2025-11-12T16:51:20-05:00';

MATCH (ds:DataSource {id:'N3302'}),(aa:AgentAction {id:'N3303'})
MERGE (ds)-[r:RELATES {id:'E3432'}]->(aa)
SET r.text='Scraped prices map to our SKUs to parameterize cross-price response simulation.',
    r.weight=0.88, r.tags=['input','mapping','simulation'],
    r.created_time='2025-11-12T16:51:22-05:00';

MATCH (ev:Event {id:'N3301'}),(aa:AgentAction {id:'N3303'})
MERGE (ev)-[r:RELATES {id:'E3433'}]->(aa)
SET r.text='Shock event triggers response modeling with guardrails.',
    r.weight=0.92, r.tags=['trigger','guardrails'], r.created_time='2025-11-12T16:51:25-05:00';

MATCH (aa:AgentAction {id:'N3303'}),(ans:AgentAnswer {id:'N3304'})
MERGE (aa)-[r:RELATES {id:'E3434'}]->(ans)
SET r.text='Modeled options converge to bundled + loyalty strategy with explicit rollback.',
    r.weight=0.95, r.tags=['solution','rollback'], r.created_time='2025-11-12T16:51:28-05:00';

// Cross-links inside Parent 3
MATCH (c31:AgentAnswer {id:'N3105'}),(c33:AgentAction {id:'N3303'})
MERGE (c31)-[r:RELATES {id:'E3435'}]->(c33)
SET r.text='Elasticity drift and cross-price flips from C31 shape shock response options.',
    r.weight=0.71, r.tags=['context_bridge','elasticity'],
    r.created_time='2025-11-12T16:51:31-05:00';

MATCH (c32:AgentAnswer {id:'N3205'}),(c33:AgentAnswer {id:'N3304'})
MERGE (c32)-[r:RELATES {id:'E3436'}]->(c33)
SET r.text='Pilot guardrails and cluster logic from C32 inform bounded competitor response.',
    r.weight=0.64, r.tags=['guardrails','bounded_response'],
    r.created_time='2025-11-12T16:51:34-05:00';

```

Parent 5:

```
/* ======  
Reasoning Bank (external registry; referenced by reasoning_pointer_ids)  
Parent 5 — Vendor & Supplier Performance Monitoring  
=====
```

RB-P5-51A:

"Chronic vendor delay triage: decompose delay variance into (a) supplier lead-time drift, (b) lane/route exogenous shocks (weather, port, driver shortage), (c) internal handling latency (dock-to-stock). Always align delay % to volume-weighted units so outlier micro-POs don't dominate."

RB-P5-51B:

"When correlating delays with external signals, prefer lagged weather/severity indices (e.g., 7/14-day moving averages of storm alerts) and driver availability proxies. Use partial correlation or fixed effects to avoid confounding from seasonality and product mix."

RB-P5-51C:

"Before penalizing a vendor, simulate split-lane sourcing. If 20–40% can be re-routed to a secondary supplier without breaching cost-to-serve thresholds, you often reduce tail risk more than tightening SLAs."

RB-P5-52A:

"Automated scorecards must be reproducible: persist the metric definitions (OTIF, defect rate, ASN compliance) with versioning; attach data lineage pointers so procurement can audit any monthly swing."

RB-P5-52B:

"Executives skim—lead with a 1-slide executive summary: traffic-light status, top movers, root-cause hypotheses, and next actions. Place diagnostics and cohort breakdowns in the appendix."

RB-P5-52C:

"Presentation style matters for adoption: if a stakeholder prefers clean tables, avoid pie charts; use bar/line with clear baselines. Encode decisions with owner + due date on each slide to convert insights into actions."

RB-P5-53A:

"Supplier CO₂ auditing: compute CO₂ per SKU-mile = (mode emission factor × distance) + packaging-adjusted scope-3 adders. Normalize by revenue and units to reveal green-washing vs. real intensity changes."

RB-P5-53B:

"When identifying greener substitutes, constrain by service level and lead-time variance. A 15% lower CO₂ supplier that adds 5 days of uncertainty can increase safety stock and offset sustainability gains."

RB-P5-53C:

"Report dual currency: environmental (tCO₂e) and financial (GM\$, logistics cost). Sustainability recommendations land better when tied to P&L and risk exposure."

*/

```
/* =====
```

Parent 5 — Conversation 5.1

"Chronic Delay Investigation" (Includes calendar booking flow)

Initiator: Employee

conv_id: 2025-11-12_ACME_P5_C51

```
===== */
```

```
/* Nodes */
```

```
MERGE (n:UserRequest {id: 'N5001'})
```

```
SET n.text = 'Employee flags persistent late shipments from Vendor #342; requests root-cause analysis correlating delay% with regional weather severity, driver shortage indices, and internal dock-to-stock latency; asks for recommended split-lane sourcing and penalty simulation; then asks the agent to book a review meeting with vendor and internal ops.',
```

```
n.conv_id = '2025-11-12_ACME_P5_C51',
```

```
n.ingestion_time = '2025-11-12T17:05:00-05:00',
```

```
n.update_time = '2025-11-12T17:05:00-05:00',
```

```
n.embedding_id = 'emb_N5001',
```

```
n.tags =
```

```
["vendor_delay","lead_time","weather","driver_shortage","dock_to_stock","split_lane","penalty","meeting_request","calendar"],
```

```
n.reasoning_pointer_ids = ['RB-P5-51A','RB-P5-51B','RB-P5-51C'],
```

```
n.user_role = 'Procurement Analyst',
```

```
n.user_id = 'u_2197';
```

```
MERGE (n:DataSource {id: 'N5002'})
```

```
SET n.text = 'Vendor_Scorecard_Annual.xlsx — annual vendor performance scorecard with Vendor_ID, Avg_Lead_Time, Delay%, Return_Rate, Quality_Score; includes OTIF by month and product line.',
```

```
n.conv_id = '2025-11-12_ACME_P5_C51',
```

```
n.ingestion_time = '2025-11-12T17:05:05-05:00',
```

```
n.update_time = '2025-11-12T17:05:05-05:00',
```

```
n.embedding_id = 'emb_N5002',
```

```
n.tags = ["scorecard","OTIF","lead_time","quality","xlsx","vendor_342"],
```

```
n.reasoning_pointer_ids = ['RB-P5-51A'],
```

```
n.source_type = 'xlsx',
```

```

n.doc_pointer = 's3://acme/proc/Vendor_Scorecard_Annual.xlsx',
n.relevant_parts = 'Sheets: Summary, OTIF_by_Month, SKU_Mix';

MERGE (n:DataSource {id: 'N5003'})
SET n.text = 'External feeds: NOAA_Weather_Alerts.csv (storm severity, region, date), Driver_Availability_Index.csv (region-week), and PortCongestion_API.json.',
n.conv_id = '2025-11-12_ACME_P5_C51',
n.ingestion_time = '2025-11-12T17:05:08-05:00',
n.update_time = '2025-11-12T17:05:08-05:00',
n.embedding_id = 'emb_N5003',
n.tags = ["weather","storm","driver_index","port_congestion","external_feed"],
n.reasoning_pointer_ids = ['RB-P5-51B'],
n.source_type = 'mixed',
n.doc_pointer = 's3://acme/external/ops_feeds/',
n.relevant_parts = 'Fields: region, week, severity_index, driver_index';

MERGE (n:AgentAction {id: 'N5004'})
SET n.text = 'Methodology: join Vendor_Scorecard_Annual with external lagged indices; compute partial correlations controlling for product mix and seasonality; simulate split-lane sourcing (20–40%) to secondary supplier; compute penalty scenarios vs. cost-to-serve.',
n.conv_id = '2025-11-12_ACME_P5_C51',
n.ingestion_time = '2025-11-12T17:05:15-05:00',
n.update_time = '2025-11-12T17:05:15-05:00',
n.embedding_id = 'emb_N5004',
n.tags = ["join","regression","partial_correlation","simulation","split_lane","penalty_model"],
n.reasoning_pointer_ids = ['RB-P5-51A','RB-P5-51B','RB-P5-51C'],
n.status = 'complete',
n.parameter_field = 'SQL/Python pipeline; lags=7,14 days; FE: month, product_line; sim: secondary_share in {0.2,0.3,0.4}';

MERGE (n:AgentAnswer {id: 'N5005'})
SET n.text = 'Conclusion: Delay% for Vendor 342 is strongly associated with storm severity (lag 7–14 days) on the Southeast lane; driver availability negatively correlates with delay spikes. Recommended split-lane: re-route 30% of volume to Vendor 511 for SKUs A/C, yielding -22% delay variance and +$18k/month cost-to-serve improvement after penalty offsets.',
n.conv_id = '2025-11-12_ACME_P5_C51',
n.ingestion_time = '2025-11-12T17:05:22-05:00',
n.update_time = '2025-11-12T17:05:22-05:00',
n.embedding_id = 'emb_N5005',
n.tags =
["recommendation","split_lane","delay_variance","cost_to_serve","vendor_342","vendor_511"],
n.reasoning_pointer_ids = ['RB-P5-51C'],
n.analysis_types = ['correlation_analysis','what_if_simulation','sourcing_optimization'],
n.metrics = ['Delay%','OTIF','Cost_to_Serve','Penalty$'];

```

```

MERGE (n:AgentAction {id: 'N5006'})
SET n.text = 'Create calendar event for Vendor 342 review and internal ops sync; invite
procurement, logistics, and vendor contact; attach summary deck.'
    n.conv_id = '2025-11-12_ACME_P5_C51',
    n.ingestion_time = '2025-11-12T17:05:28-05:00',
    n.update_time = '2025-11-12T17:05:28-05:00',
    n.embedding_id = 'emb_N5006',
    n.tags = ["calendar","meeting","invite","vendor_review"],
    n.reasoning_pointer_ids = [],
    n.status = 'complete',
    n.parameter_field = 'CalendarAPI.create(title="Vendor 342 Review",
start="2025-11-14T10:00-05:00", end="2025-11-14T11:00-05:00",
attendees=["proc@acme.com","logistics@acme.com","v342@vendor.com"], location="Zoom");
attach=deck_v342.pdf';

```

```

MERGE (n:Event {id: 'N5007'})
SET n.text = 'Calendar event: Vendor 342 Review Meeting (Zoom). Objective: align on delay
drivers, agree split-lane pilot, and penalty framework.'

```

```

    n.conv_id = '2025-11-12_ACME_P5_C51',
    n.ingestion_time = '2025-11-12T17:05:31-05:00',
    n.update_time = '2025-11-12T17:05:31-05:00',
    n.embedding_id = 'emb_N5007',
    n.tags = ["calendar","meeting","vendor_342","split_lane","penalty"],
    n.reasoning_pointer_ids = [],
    n.source_type = 'Calendar',
    n.start_date = '2025-11-14T10:00:00-05:00',
    n.end_date = '2025-11-14T11:00:00-05:00';

```

```

/* Edges */
MATCH (a {id:'N5001'}),(b {id:'N5002'})
MERGE (a)-[r:RELATES {id:'E9001'}]->(b)
SET r.text='Request pulls annual vendor scorecard as primary evidence for lead-time and OTIF
baselines.',
    r.weight=0.92, r.tags=['trigger','evidence','scorecard'],
    r.created_time='2025-11-12T17:05:40-05:00';

```

```

MATCH (a {id:'N5001'}),(b {id:'N5003'})
MERGE (a)-[r:RELATES {id:'E9002'}]->(b)
SET r.text='Request requires external exogenous signals (weather, driver index, port
congestion) to explain delay variance.',
    r.weight=0.90, r.tags=['exogenous','weather','driver'],
    r.created_time='2025-11-12T17:05:42-05:00';

```

```
MATCH (a {id:'N5002'}),(b {id:'N5004'})  
MERGE (a)-[r:RELATES {id:'E9003'}]->(b)  
SET r.text='Scorecard is joined to external feeds; methodology documents lags and fixed effects  
for unbiased attribution.',  
    r.weight=0.86, r.tags=['join','methodology'], r.created_time='2025-11-12T17:05:44-05:00';
```

```
MATCH (a {id:'N5004'}),(b {id:'N5005'})  
MERGE (a)-[r:RELATES {id:'E9004'}]->(b)  
SET r.text='Simulation and partial correlations produce the actionable recommendation.',  
    r.weight=0.94, r.tags=['solution','recommendation'],  
    r.created_time='2025-11-12T17:05:46-05:00';
```

```
MATCH (a {id:'N5001'}),(b {id:'N5006'})  
MERGE (a)-[r:RELATES {id:'E9005'}]->(b)  
SET r.text='Employee explicitly asks to book a meeting; agent executes calendar creation with  
attendees and attachment.',  
    r.weight=0.97, r.tags=['calendar','execution'], r.created_time='2025-11-12T17:05:48-05:00';
```

```
MATCH (a {id:'N5006'}),(b {id:'N5007'})  
MERGE (a)-[r:RELATES {id:'E9006'}]->(b)  
SET r.text='Calendar action instantiates a concrete meeting event with objective and time  
window.',  
    r.weight=0.95, r.tags=['event','meeting'], r.created_time='2025-11-12T17:05:50-05:00';
```

```
/* ======  
Parent 5 — Conversation 5.2  
“Automated Scorecard Generation” (Includes presentation + style preferences)  
Initiator: AI Agent (then employee asks for manager-ready deck)  
conv_id: 2025-11-12_ACME_P5_C52  
===== */
```

```
/* Nodes */  
MERGE (n:UserRequest {id: 'N5011'})  
SET n.text = 'Follow-up: Employee asks the agent to package the monthly vendor scorecard  
into a manager-ready deck with an executive summary and minimal charts; requests revisions  
to match manager's style (clean tables, dark theme, action owners/due dates).',  
    n.conv_id = '2025-11-12_ACME_P5_C52',  
    n.ingestion_time = '2025-11-12T17:06:10-05:00',  
    n.update_time = '2025-11-12T17:06:10-05:00',  
    n.embedding_id = 'emb_N5011',  
    n.tags =  
    ["scorecard","presentation","executive_summary","style_preference","manager_readout"],  
    n.reasoning_pointer_ids = ['RB-P5-52B','RB-P5-52C'],
```

```
n.user_role = 'Procurement Analyst',
n.user_id = 'u_2197';
```

```
MERGE (n:DataSource {id: 'N5012'})
SET n.text = 'Warehouse telemetry + ERP extracts — automated monthly roll-up for top
improving/declining vendors; includes ASN compliance and defect logs.',
n.conv_id = '2025-11-12_ACME_P5_C52',
n.ingestion_time = '2025-11-12T17:06:14-05:00',
n.update_time = '2025-11-12T17:06:14-05:00',
n.embedding_id = 'emb_N5012',
n.tags = ["telemetry","ERP","ASN","defects","monthly_rollup"],
n.reasoning_pointer_ids = ['RB-P5-52A'],
n.source_type = 'parquet',
n.doc_pointer = 's3://acme/ops/monthly_rollup/2025-10/',
n.relevant_parts = 'Tables: vendor_monthly, asn_compliance, defect_log';
```

```
MERGE (n:AgentAction {id: 'N5013'})
```

```
SET n.text = 'Generate scorecard deck v1: 1-slide exec summary (traffic-light status, top
movers, root-causes, actions), detailed vendor pages with OTIF trendlines and defect cohorts,
appendix with lineage/definitions.',
n.conv_id = '2025-11-12_ACME_P5_C52',
n.ingestion_time = '2025-11-12T17:06:20-05:00',
n.update_time = '2025-11-12T17:06:20-05:00',
n.embedding_id = 'emb_N5013',
n.tags = ["deck","pptx","automation","definitions","appendix"],
n.reasoning_pointer_ids = ['RB-P5-52A','RB-P5-52B'],
n.status = 'complete',
n.parameter_field = 'DeckBuilder.create(template="standard_light",
sections=["ExecSummary","VendorDeepDives","Appendix"]);
out=/reports/vendor_scorecard_v1.pptx';
```

```
MERGE (n:UserPreference {id: 'N5014'})
```

```
SET n.text = 'Manager visual style: prefers dark theme, clean tables over pie charts, concise
bar/line charts with baselines; wants action owners and due dates on each slide.',
n.conv_id = '2025-11-12_ACME_P5_C52',
n.ingestion_time = '2025-11-12T17:06:25-05:00',
n.update_time = '2025-11-12T17:06:25-05:00',
n.embedding_id = 'emb_N5014',
n.tags = ["report_style", "tables_not_pie", "dark_theme", "actions_on_slides"],
n.reasoning_pointer_ids = ['RB-P5-52C'],
n.preference_type = 'report_style';
```

```
MERGE (n:AgentAction {id: 'N5015'})
```

```
SET n.text = 'Revise deck to dark theme; replace pie charts with tables and bars; add "Owner / Due Date" callouts; re-order slides so exec summary is first.',  
n.conv_id = '2025-11-12_ACME_P5_C52',  
n.ingestion_time = '2025-11-12T17:06:31-05:00',  
n.update_time = '2025-11-12T17:06:31-05:00',  
n.embedding_id = 'emb_N5015',  
n.tags = ["revision", "style_alignment", "deck_update"],  
n.reasoning_pointer_ids = ['RB-P5-52C'],  
n.status = 'complete',  
n.parameter_field = 'DeckBuilder.update(in="/reports/vendor_scorecard_v1.pptx",  
theme="dark", charts=["bar", "line"], tables=true, actions=true,  
out="/reports/vendor_scorecard_mgr_v2.pptx");
```

```
MERGE (n:AgentAnswer {id: 'N5016'})
```

```
SET n.text = 'Final deliverable: manager-ready vendor scorecard deck (dark theme) with traffic-light status, top 5 improving/declining vendors, root-cause notes (ASN defects, packaging), explicit owners and deadlines.',
```

```
n.conv_id = '2025-11-12_ACME_P5_C52',  
n.ingestion_time = '2025-11-12T17:06:36-05:00',  
n.update_time = '2025-11-12T17:06:36-05:00',  
n.embedding_id = 'emb_N5016',  
n.tags = ["deliverable", "deck", "executive_summary", "actions"],  
n.reasoning_pointer_ids = ['RB-P5-52B', 'RB-P5-52C'],  
n.analysis_types = ['report_generation', 'kpi_rollup'],  
n.metrics = ['OTIF', 'ASN_Compliance', 'Defect_Rate'];
```

```
/* Edges */
```

```
MATCH (a {id:'N5012'}),(b {id:'N5013'})  
MERGE (a)-[r:RELATES {id:'E9011'}]->(b)  
SET r.text='Monthly telemetry and ERP extracts feed the automated deck generation pipeline.',  
r.weight=0.88, r.tags=['data_lineage', 'automation'],  
r.created_time='2025-11-12T17:06:45-05:00';
```

```
MATCH (a {id:'N5011'}),(b {id:'N5013'})
```

```
MERGE (a)-[r:RELATES {id:'E9012'}]->(b)  
SET r.text='Employee request triggers building a manager-ready deck with executive summary first.',  
r.weight=0.93, r.tags=['trigger', 'presentation'], r.created_time='2025-11-12T17:06:47-05:00';
```

```
MATCH (a {id:'N5014'}),(b {id:'N5015'})
```

```
MERGE (a)-[r:RELATES {id:'E9013'}]->(b)  
SET r.text='Captured style preference guides the revision: dark theme, tables > pie charts, action owners.',  
r.weight=0.96, r.tags=['preference', 'styling'], r.created_time='2025-11-12T17:06:49-05:00';
```

```
MATCH (a {id:'N5015'}),(b {id:'N5016'})  
MERGE (a)-[r:RELATES {id:'E9014'}]->(b)  
SET r.text='Revised deck becomes the final deliverable aligned to the manager's expectations.',  
r.weight=0.95, r.tags=['deliverable','alignment'], r.created_time='2025-11-12T17:06:51-05:00';
```

```
/* ======  
Parent 5 — Conversation 5.3  
“Supplier Carbon Footprint Audit”  
Initiator: Employee  
conv_id: 2025-11-12_ACME_P5_C53  
===== */
```

```
/* Nodes */  
MERGE (n:UserRequest {id: 'N5021'})  
SET n.text = 'Sustainability team requests CO2 audit across suppliers: compute tCO2e per  
SKU-mile using mode emission factors and packaging; normalize by revenue and units; identify  
non-compliant partners and propose greener regional substitutes without jeopardizing service  
levels.' ,
```

```
    n.conv_id = '2025-11-12_ACME_P5_C53',  
    n.ingestion_time = '2025-11-12T17:07:05-05:00',  
    n.update_time = '2025-11-12T17:07:05-05:00',  
    n.embedding_id = 'emb_N5021',  
    n.tags = ["sustainability","co2","scope3","sku_mile","green_substitutes","service_level"],  
    n.reasoning_pointer_ids = ['RB-P5-53A','RB-P5-53B','RB-P5-53C'],  
    n.user_role = 'Sustainability Analyst',  
    n.user_id = 'u_3011';
```

```
MERGE (n:DataSource {id: 'N5022'})  
SET n.text = 'Sustainability_2025_Tests.docx — corporate goals by business unit; thresholds  
for CO2 intensity and compliance windows.',  
    n.conv_id = '2025-11-12_ACME_P5_C53',  
    n.ingestion_time = '2025-11-12T17:07:09-05:00',  
    n.update_time = '2025-11-12T17:07:09-05:00',  
    n.embedding_id = 'emb_N5022',  
    n.tags = ["policy","targets","thresholds","docx"],  
    n.reasoning_pointer_ids = [],  
    n.source_type = 'docx',  
    n.doc_pointer = 's3://acme/policy/Sustainability_2025_Tests.docx',  
    n.relevant_parts = 'Sections: CO2 Intensity Targets, Compliance Timing';
```

```
MERGE (n:DataSource {id: 'N5023'})
```

```

SET n.text = 'Supplier_CO2_Emissions.csv — route, mode, distance, emission_factor,
packaging_weight, SKU, units, revenue.',
    n.conv_id = '2025-11-12_ACME_P5_C53',
    n.ingestion_time = '2025-11-12T17:07:12-05:00',
    n.update_time = '2025-11-12T17:07:12-05:00',
    n.embedding_id = 'emb_N5023',
    n.tags = ["emissions","routes","mode","distance","units","revenue","csv"],
    n.reasoning_pointer_ids = ['RB-P5-53A'],
    n.source_type = 'csv',
    n.doc_pointer = 's3://acme/sustainability/Supplier_CO2_Emissions.csv',
    n.relevant_parts = 'Columns: route_id, mode, distance_km, ef_kg_per_km, packaging_kg,
sku, units, revenue_usd';

```

MERGE (n:AgentAction {id: 'N5024'})

```

SET n.text = 'Compute CO2 per SKU-mile (tCO2e) = (mode EF × distance) + packaging scope-3
adders; normalize by revenue and units; rank by intensity; constrain greener substitutes by
service level and lead-time variance.'

```

```

    n.conv_id = '2025-11-12_ACME_P5_C53',
    n.ingestion_time = '2025-11-12T17:07:18-05:00',
    n.update_time = '2025-11-12T17:07:18-05:00',
    n.embedding_id = 'emb_N5024',
    n.tags = ["calculation","normalization","ranking","constraints","substitution"],
    n.reasoning_pointer_ids = ['RB-P5-53A','RB-P5-53B'],
    n.status = 'complete',
    n.parameter_field = 'calc_version=1.2; service_level>=95%; lead_time_var<=2.0 days';

```

MERGE (n:AgentAnswer {id: 'N5025'})

```

SET n.text = 'Findings: 9 suppliers exceed CO2 intensity thresholds; recommend shifting 25% of
volume on two corridors to regional rail + greener packaging, reducing tCO2e by 17% with
neutral P&L after logistics re-optimization.'

```

```

    n.conv_id = '2025-11-12_ACME_P5_C53',
    n.ingestion_time = '2025-11-12T17:07:24-05:00',
    n.update_time = '2025-11-12T17:07:24-05:00',
    n.embedding_id = 'emb_N5025',
    n.tags = ["recommendation","green_substitute","rail","packaging","pnl_neutral"],
    n.reasoning_pointer_ids = ['RB-P5-53C'],
    n.analysis_types = ['sustainability_audit','routing_optimization'],
    n.metrics = ['tCO2e','Service_Level','GM$','Logistics_Cost'];

```

/ Edges */*

```

MATCH (a {id:'N5022'}),(b {id:'N5024'})
MERGE (a)-[r:RELATES {id:'E9021'}]->(b)

```

```

SET r.text='Corporate targets define thresholds used in emissions computations and compliance
checks.'

```

```
r.weight=0.84, r.tags=['policy','thresholds'], r.created_time='2025-11-12T17:07:31-05:00';

MATCH (a {id:'N5023'}),(b {id:'N5024'})
MERGE (a)-[r:RELATES {id:'E9022'}]->(b)
SET r.text='Route/mode/distance and packaging data parameterize SKU-mile CO2 calculations.',
    r.weight=0.88, r.tags=['data_input','calculation'], r.created_time='2025-11-12T17:07:33-05:00';

MATCH (a {id:'N5024'}),(b {id:'N5025'})
MERGE (a)-[r:RELATES {id:'E9023'}]->(b)
SET r.text='Computed intensities and constrained substitution analysis yield actionable
sustainability recommendations.',
    r.weight=0.93, r.tags=['recommendation','sustainability'],
    r.created_time='2025-11-12T17:07:35-05:00';
```