GFX Threshold Deviation Dashboard - Business Logic (Detailed & Readable)

This section provides the precise business logic used by the dashboard, written for readability and direct implementation.

A. Definitions

- **Trade**: A record containing at minimum: Trade ID, Currency Pair (e.g., CADAUD), Notional, Trade/COB Date, and any exception flags. - **Currency Group**: A group (G1..Gn) that a currency belongs to, each group having a numeric threshold (e.g., G1=0.40, G2=0.60). - **Threshold Types**: *Original* (prod), *Proposed* (from analysis), *Adjusted* (user-edited, used for impact). - **Deviation** (per trade): A non-negative value representing how far a trade's metric deviates from its group threshold. - **Bucket**: A closed-open interval [L, U) of deviation values for summarization (e.g., [0.0, 0.2), [0.2, 0.4), ...).

B. Inputs

1) **Trade dataset** from UAT/PROD APIs for selected Product Type, Legal Entity, Source System, Date Range. 2) **Threshold file** providing group -> threshold mapping per currency for Original, Proposed; Adjusted is derived in-UI. 3) **User selections**: threshold set to apply (Adjusted by default), filters, and edits within the grid.

C. Currency Group Assignment Rule

For each trade with currency pair XY (e.g., CADAUD): 1) Look up group thresholds for **X** and for **Y** from the active threshold set (Adjusted > Proposed > Original). 2) Compare the two group thresholds. 3) **Assign the trade to the currency whose group has the higher threshold** (i.e., the maximum). This becomes the trade's *evaluation group*. 4) Persist both currencies for distribution calculations later.

Pair	CAD Group/Thr	AUD Group/Thr	Assigned Group (Max Thr)
CADAUD	G2 / 0.60	G1 / 0.40	CAD → G2 (0.60)

D. Deviation Computation

Let *T* be the trade metric used for deviation (e.g., |ΔPrice|, |ΔNotional|, or a precomputed risk score). Let *Thr_g* be the threshold value of the assigned evaluation group.
Formula: *Deviation = max(0, T – Thr_g)* If the trade metric is itself a deviation-like number, use it directly against *Thr_g*.

E. Bucketing Rules

1) Build monotonic bucket edges starting at 0.0 up to at least the 99th percentile of deviations (UI-configurable width, e.g., 0.2). 2) Place each trade's *Deviation* into a single bucket [L, U). 3) **Highlight Rule**: In group summaries, highlight any bucket row where **L >= MaxThreshold(group)**. This visually flags regions beyond tolerated risk.

F. Distribution & Counting

- **Evaluation**: A trade contributes to alert counts using its assigned evaluation group only. - **Distribution**: The same trade contributes to deviation distribution of *both currencies* in its pair (helps currency-level views).

G. Impact Analysis (on Adjusted Edits)

When a user edits any threshold (Adjusted): 1) Recompute the evaluation group for every trade (per Rule C). 2) Recalculate each trade's deviation (per Section D). 3) Re-bucket deviations (per Section E). 4) Recount alerts by severity and group; regenerate summaries/pivots. 5) Update **delta metrics** vs Original/Proposed: ΔAlertCount, % change, counts per severity.

H. Alert Categorization (Severity Mapping)

Severity	Deviation Condition (example)	Usage
Info	Deviation < 0.2	Monitor only
Low	0.2 ≤ Deviation < 0.4	Triage later
Medium	0.4 ≤ Deviation < 0.6	Actionable if persistent
High	Deviation ≥ 0.6	Immediate review

I. Rounding Policy (Ops Reality)

- Ops apply rounding when **setting** thresholds (e.g., 0.43 → 0.4), not during impact analysis. - The dashboard respects this: rounding is applied to *Adjusted* values when saved; computations use the stored rounded value to mirror Ops.

J. Edge Cases & Tie-Breakers

1) **Equal thresholds**: If both currencies have the same threshold, default to the alphabetically first currency for evaluation group (deterministic). 2) **Missing threshold**: Fall back to Proposed → Original; if still missing, drop the trade from alerts and flag in an "Integrity" panel. 3) **Zero/negative values**: Clamp thresholds and deviations at ≥ 0. 4) **Outliers**: Display but allow user to exclude from recalibration via a toggle; still counted in a separate "Outliers" summary.

K. Reference Pseudocode

```
for trade in trades: x, y = trade.ccy1, trade.ccy2 thr_x = get_threshold(x) thr_y =
get_threshold(y) if thr_x is None or thr_y is None: mark_integrity_issue(trade); continue eval_thr
= max(thr_x, thr_y) eval_ccy = x if thr_x >= thr_y else y # tie -> alphabetic rule if equal
deviation = max(0, trade.metric - eval_thr) bucket = place_in_bucket(deviation)
counts.eval[eval_ccy_group][bucket] += 1 dist[x][bucket] += 1 dist[y][bucket] += 1
```

L. Key KPIs surfaced

- Total Alerts (by severity / group / currency) - ΔAlerts vs Original / Proposed (% change) - Bucketed distributions with highlight beyond MaxThreshold - Top Drivers (currencies / entities) and Outliers count

Architecture Context (Reference)