# Formula Split with Explicit Mathematical Expressions

|  |  |  |  |
| --- | --- | --- | --- |
| Formula / Logic | Goes In | Formula Definition (example) | Explicit Mathematical Formula |
| Categorical Scoring Map (e.g. Payment Terms) | Policy Engine | "30 days" → 4, "60 days" → 7, "90 days" → 10 | Scoreᵢ = scoring\_map(xᵢ) |
| Range-Based Scoring Map (e.g. Contract Term) | Policy Engine | "3 years" → 10, "5-7 years" → 8, "7+ years" → 5 | Scoreᵢ = range\_map(xᵢ) = {5 if 3 ≤ xᵢ < 5; 8 if 5 ≤ xᵢ < 7; 10 if xᵢ ≥ 7} |
| Clause Weight Lookup | Policy Engine | Clause-level importance | Weightᵢ = policy\_matrix[i] |
| Normalization (Inverse Min-Max, e.g. Unit Price) | Supplier Ranking Agent | Lower = better | Scoreᵢ = ((max(x) - xᵢ) / (max(x) - min(x))) × 10 |
| Normalization (Standard Min-Max, e.g. Warranty Years) | Supplier Ranking Agent | Higher = better | Scoreᵢ = ((xᵢ - min(x)) / (max(x) - min(x))) × 10 |
| TCO per Product | Supplier Ranking Agent | Unit price × volume | TCOᵢⱼ = UnitPriceᵢⱼ × Volumeᵢⱼ |
| TCO Aggregated per Supplier | Supplier Ranking Agent | Sum across all products | TCO\_Supplierⱼ = Σ (TCOᵢⱼ) |
| Weighted Score per Clause | Supplier Ranking Agent | Score × (Weight / 100) | WeightedScoreᵢ = Scoreᵢ × (Weightᵢ / 100) |
| Total Supplier Score | Supplier Ranking Agent | Sum of all weighted clause scores | TotalScoreⱼ = Σ (WeightedScoreᵢⱼ) |
| Supplier Ranking | Supplier Ranking Agent | Order by descending score | Rankⱼ = DescendingRank(TotalScoreⱼ) |
| Boosted Score (if applicable) | Policy Engine + Agent | Bonus application | BoostedScoreᵢ = Scoreᵢ + BoostValueᵢ |
| Boosted Weight (if applicable) | Policy Engine + Agent | Bonus weight | BoostedWeightᵢ = Weightᵢ × (1 + BoostFactor) |
| Conditional Weight Override (Policy Logic) | Policy Engine | Adaptive logic by deal type | If TCOⱼ > 1,000,000 ⇒ Weight\_Price = 60 |