Proposal: BEAM 2.0—The Automated Governance Contract for the AI Era

To: Lawrence Corr, Co-Creator of the BEAM Methodology **From:** [Your Organization/Team] **Date:** October 29, 2025 **Subject:** Operationalizing BEAM as the Direct Automation Input for the Semantic Layer

---

Executive Summary: BEAM as the Zero-Translation Artifact

The rise of **Agentic Systems** and **Generative AI** is accelerating the demand for consistent, high-governance data. These systems rely entirely on being fed **reliable, version-controlled metrics**—a capability often undermined by traditional, fragmented BI semantic layers.

We propose a strategic evolution of the BEAM methodology that transforms the collaborative **BEAM Table** from a descriptive design tool into an **executable automation contract.** By enhancing the BEAM artifact with the specific technical metadata required by a modern, code-based Semantic Layer (e.g., dbt MetricFlow), we can eliminate three major layers of traditional complexity:

1. **The Data Mart Layer:** Replaced by MetricFlow's dynamic joins.
2. **The BI Semantic Layer:** Replaced by a centralized, version-controlled MetricFlow YAML.
3. **The Business Analyst Translation Layer:** Replaced by a deterministic automation script.

This enhancement locks **business requirements** directly to **governed code**, providing the necessary confidence and agility for the AI era.

---

1. The Streamlined Architecture

This approach maintains the integrity of the integrated **Inmon 3NF** core while connecting it directly to consumption layers via metadata, creating a **Zero-Manual-Translation, Zero-Redundancy** model.

The AI-Era Data Flow

The traditional pipeline is replaced with a streamlined, high-governance process:

1. **Ingestion/EL:** Handled by managed services (eliminating custom Data Engineering code).
2. **Transformation (T): Analytics Engineer** builds integrated, clean dbt models (3NF structure) as the source of truth.

3. **Requirements: BA Architect** conducts **Modelstorming** to output the **Enhanced BEAM Table**.
4. **Semantic Generation:** An **Automation Script** consumes the BEAM Table and generates MetricFlow YAML files.
5. **Consumption: Agentic Systems** and APIs query the Semantic Layer directly.

The Evolved Roles

This structure redefines core data roles for maximum automation and focus:

- **Analytics Engineer (AE): The Full-Stack Data Professional.**
  - Focuses solely on data modeling and transformation (the T in ELT). The **traditional Data Engineer role effectively disappears** as E and L are managed.
- **Business Analyst (BA): The BEAM Architect.**
  - Shifts from a manual **translator** to a **requirements architect**.
  - Their job is to ensure the BEAM output is 100% accurate, including the **technical metadata** required by the automation script.

---

2. The BEAM 2.0 Specification: A Contract for Automation

The BEAM artifact must be extended to capture the specific metadata that allows the automation script to generate the MetricFlow YAML without human intervention. This proposal formalizes two linked tables.

A. The Enhanced Business Event Table (Defining Semantic Models and Simple Measures)

This table defines the **Semantic Model** (dimensions and entities) and all **simple metrics** derived from a single column.

Role in Event	Enhanced Column Name	Required Value Type	MetricFlow Component Target
How (Key)	`event_key_column`	String (e.g., `order_key`)	**Primary Entity**
Who, What, Where, Why	`dimension_column_name`	String (e.g., `customer_region`)	**Dimension** or **Entity**
NEW	`entity_role`	`primary`, `foreign`, `unique`, `none`	**Entity Type** (Crucial for Dynamic Joins)
When	`time_column_name`	String (e.g., `order_date`)	**agg_time_dimension**

Role in Event	Enhanced Column Name	Required Value Type	MetricFlow Component Target
How Many	`measure_source_column`	String (e.g., `sale_amount`)	**Measure Expression (expr)**
NEW	`simple_metric_name`	String (e.g., `total_revenue`)	**Simple Metric Name**
NEW	`aggregation_type`	`sum`, `count_distinct`, `avg`, `min`	**Measure Aggregation (agg)**

B. The Complex Metric Definition Table (Defining Derived Metrics)

This separate table allows the BA Architect to define **Ratios, Cumulative Metrics, and Derived Metrics** by referencing the **simple metrics** defined in Table A.

MetricFlow Component	Enhanced Column Name	Required Value Type	Notes
Metric Name	`metric_name`	String	Must be unique.
Metric Type	`metric_type`	`ratio`, `cumulative`, `derived`	Specifies the calculation structure.
Ratio Definition	`numerator_metric`	Existing Simple Metric Name	Required if `metric_type` is `ratio`.
Ratio Definition	`denominator_metric`	Existing Simple Metric Name	Required if `metric_type` is `ratio`.
Derived Definition	`derived_expression`	String (referencing `[other_metrics]`)	Required if `metric_type` is `derived`.
Cumulative Definition	`time_window`	e.g., P30D, all time	Required if `metric_type` is `cumulative`.

---

3. Value Proposition

By formalizing the BEAM output in this manner, the methodology evolves from a design guide to a powerful **governance and automation contract** fit for

the AI era:

1. **Governed AI Inputs:** Ensures Agentic Systems and LLM interfaces (Text-to-Metric) are fed with a single, governed definition, eliminating metric drift and hallucination.
2. **Model Efficiency:** The Analytics Engineer focuses on a lean, integrated EDW, eliminating the need for redundant Data Marts.
3. **Scalability:** The BEAM Table becomes the only human-managed artifact required to scale the semantic layer, drastically reducing maintenance overhead.