The Lexical Independence of Schema Semantics

Constraint Modality, Not Predicate Verbalization, Determines LLM Text-to-SQL Accuracy

Edward Kench
Independent Research · GramSpec
May 2026

Abstract

Object-Role Modeling and its derivatives, including the GRAM specification, hold that natural-language readings of fact types are foundational to machine-readable schema semantics. This paper reports an empirical finding that contradicts that position. When a large language model is provided with a schema represented as binary fact types, the lexical content of the predicate verb is operationally inert: substituting a single generic predicate ("has") for all relationships across a schema produces no measurable degradation in text-to-SQL accuracy relative to descriptive verbs ("handles," "ships to," "is assigned to"). The result was observed in primary testing on Gemini 2.5 Flash and confirmed in subsequent agentic deployment on Claude Opus 4.7. The load-bearing semantic content resides not in the predicate but in the constraint modality that travels with each role: uniqueness, mandatoriness, and the cardinality classification they jointly determine. This finding distinguishes the present work from the dominant industry approaches to LLM-based analytics—semantic-layer products that depend on human-readable labeling, retrieval-augmented prompting over DDL, and the integration of LLMs with pre-built BI semantic models—all of which leave the schema-inference burden on the model. It also resolves a long-standing tension in the ORM tradition between the pedagogical role of natural-language verbalization and its presumed operational role in automated reasoning. The two roles are separable. The former serves human authors and reviewers; the latter does not exist.

1. Introduction

The Object-Role Modeling tradition, originating with Halpin and grounded in earlier work by Codd and Chen, treats the natural-language reading of a fact type as a foundational element of the conceptual schema. A fact type such as "Employee handles Order" is taken to express, in human-readable form, the predicate that binds two entity types into an asserted relation. The verbalization is held to be more than decorative: it is held to be the elementary semantic unit of the model, the carrier of the business meaning that the relational structure alone cannot capture.

The GRAM specification (Kench, 2026) inherits this position. Its formal definition of a fact type as a binary relation expressed through a natural-language reading places the predicate verb at the center of the model. The specification's normative comparison between DDL-only and GRAM-compliant interpretations

of a database schema rests in part on the readability of predicates such as “Employee handles Order” as the mechanism by which an LLM resolves join semantics and reasoning paths.

This paper reports an empirical finding, observed during the development of an automated database-to-GRAM conversion tool, that calls this foundational assumption into question. The finding is simple and the evidence is direct: an LLM consuming a GRAM-compliant schema is indifferent to the lexical content of the predicate verb. Replacing every domain-specific verb with the single generic predicate “has” produces no measurable reduction in text-to-SQL accuracy, provided the constraint metadata—uniqueness, mandatoriness, cardinality—remains intact.

The implication is that the operational content of a binary fact type, as consumed by an LLM, is not located in the verb. It is located in the constraint modality that travels with the relation. The verbalization layer is scaffolding for the human author of the schema; it is not signal for the machine reader.

2. Background and Prior Position

Halpin’s formulation of Object-Role Modeling positions the natural-language reading as the elementary unit of fact-based modeling. The justification offered in the tradition is twofold. First, the reading provides a verbalization that domain experts can validate without requiring competence in formal logic or relational algebra. Second, the reading is asserted to carry semantic content beyond what the constraint structure alone provides—to anchor the relation to the business meaning of the predicate.

The first justification is uncontroversial and remains valid under the finding reported here. The second is the position this paper contradicts.

The GRAM specification draws on this tradition and treats the predicate reading as a first-class element of the schema, formalized as an atomic predicate P(x, y) over two participants. The specification’s normative serialization includes the natural-language reading as a required field of every fact type, alongside its inverse reading and any alternate readings. The implicit assumption, consistent with Halpin’s position, is that these readings contribute operationally to the LLM’s ability to generate correct SQL from the schema.

3. The Industry Landscape

The finding reported in this paper is best situated against the current state of LLM-driven analytics in the enterprise, where several distinct approaches have emerged and where each implicitly takes a position on how machine-readable schema semantics should be produced and consumed.

3.1 Retrieval-Augmented Prompting over DDL

The most common approach—adopted by the majority of commercial text-to-SQL offerings and the default in most internal enterprise deployments—provides the LLM with raw or lightly annotated Data

Definition Language. The model receives table definitions, column types, and foreign key declarations, and is expected to infer the conceptual structure of the database from this physical metadata. Retrieval techniques narrow the prompt to relevant tables; few-shot examples demonstrate query patterns. The model is asked to bridge the gap between physical storage and conceptual meaning through inference alone.

This approach concentrates the entire schema-inference burden on the model. The semantic content the model needs—which joins are valid, which relationships are mandatory, what the cardinality of each relation is—is not declared anywhere in the prompt and must be reconstructed at inference time from naming conventions and structural heuristics. The failure modes are well-documented: incorrect join selection, missed or fabricated relationships, silent miscounting through nullable foreign keys, and confabulated columns that match the model's expectations rather than the schema.

3.2 LLM Integration with Existing BI Semantic Models

A second approach, exemplified by Microsoft's Power BI Copilot and Tableau's comparable offerings, integrates LLM-based natural-language interfaces with pre-existing BI semantic models. The semantic model in these systems—the tabular model in Power BI, the data source in Tableau—was designed in the era preceding LLMs as an authoring artifact for human analysts. It expresses tables, measures, relationships, and friendly column names; it does not formally specify cardinality and participation as machine-readable constraints in a manner intended for automated reasoning.

When an LLM consumes such a model, it consumes an artifact that was designed for human navigation, not for formal logical inference. The model is asked to interpret friendly names, disambiguate similarly-named measures, and infer relationship semantics from a structure that was never intended to declare them rigorously. This approach inherits the strengths of the BI tool's human-facing investment—familiar labels, curated measures, established relationships—and inherits its limits, which include the well-known problem of measure proliferation, in which a long-running BI deployment accumulates dozens of overlapping calculated measures whose distinctions are documented only in the institutional memory of the analyst who created them.

3.3 The Semantic Layer Movement

A third approach, exemplified by dbt's semantic layer, Cube, AtScale, and the LookML ecosystem, produces an explicit semantic layer above the warehouse, typically expressed as a domain-specific configuration language. These systems define entities, metrics, dimensions, and joins as first-class objects, and present a curated analytical interface to downstream consumers including LLMs.

The semantic-layer movement represents a substantial improvement over DDL-only prompting in that it makes some structural elements explicit. It also, however, inherits the prevailing assumption of the field: that the operationally important content of the semantic layer is the human-meaningful labeling—metric

names, dimension descriptions, friendly entity names. The constraint modality of relationships is typically expressed in dimensional terms (a many-to-one join from a fact to a dimension is declared) rather than as the formal logical structure of the predicate. The resulting layer is human-readable, machine-consumable, and structurally less expressive than the conceptual schema that an LLM ideally requires.

3.4 Object-Role Modeling and Predicate-Based Approaches

A fourth approach, of which GRAM is an instance, draws on the Object-Role Modeling tradition and represents the schema as a collection of binary fact types with explicit constraint metadata. This approach makes the conceptual structure of the database explicit in a form that is intended to be directly consumable by an LLM without intervening inference. Cardinality, participation, functional dependency, and uniqueness are declared on each role; the LLM reads the structure rather than inferring it.

This is the approach within which the finding of this paper is situated. The finding distinguishes it from the other three approaches in a way that is worth stating directly. In DDL-prompted systems, the LLM must infer everything about schema semantics. In BI-integrated systems, the LLM must infer some of it from a model designed for humans. In the semantic-layer movement, the LLM must infer the remainder from a curated but informally specified structure. In a predicate-based approach with explicit constraint modality, the LLM infers none of it; the structure is read. The result reported here narrows this further: within a predicate-based approach, the natural-language verbalization of each predicate is not required for accurate consumption. The constraint modality alone is sufficient.

4. Method

The finding emerged from the design of an automated tool that produces GRAM-compliant schemas from a source database. The tool inspects the database's tables, columns, foreign keys, uniqueness constraints, and nullability declarations, and emits a GRAM JSON document. The constraint metadata—uniqueness and mandatoriness on each role, cardinality classification on each relation—is inferred directly from the source schema and is rendered with high fidelity. The natural-language readings, however, present a generation problem: a foreign key from Orders.EmployeeID to Employees.EmployeeID yields no descriptive verb without external interpretation.

An early version of the tool defaulted to emitting the predicate "has" for every relation, with "is of" as its inverse, on the working assumption that domain-specific verbs would be added in a subsequent curation pass. The resulting GRAM files were used in text-to-SQL evaluation against a test corpus of business questions spanning multiple database schemas and domains, including supply chain, finance, customer relationship management, and human resources.

Primary testing was conducted on Gemini 2.5 Flash. The choice was deliberate: a smaller, faster, less capable model than the largest frontier offerings provides a more demanding test of whether the representation itself is sufficient, independent of model capacity. Subsequent agentic deployment used

Claude Opus 4.7, which permitted additional categories of multi-turn analytical reasoning beyond single-shot text-to-SQL.

The expected outcome, given the prior position in the ORM tradition, was that accuracy would be materially degraded relative to GRAM files with descriptive verbs. The observed outcome was the opposite. Gemini 2.5 Flash, consuming GRAM files in which every relation was labeled with the generic "has" predicate, produced correct SQL across the test corpus at rates indistinguishable from those measured for descriptive-verb GRAM files. Single-shot text-to-SQL accuracy approached 97 percent, with residual failures concentrated in recoverable syntactic faults—alias-scoping errors, GROUP BY clause mismatches, and similar issues that a database engine identifies on execution—rather than semantic misinterpretations that would produce confident wrong answers.

The test corpus comprised thousands of business questions, including analytically demanding categories such as cohort retention analysis, statistical outlier detection requiring computation of standard deviations and z-scores, Herfindahl-Hirschman Index calculations for revenue concentration, ABC tier segmentation against cumulative revenue thresholds, manager-rollup queries against self-referential employee hierarchies, market-basket co-occurrence analyses, cross-border shipment detection requiring disambiguation of multiple country-typed attributes on the same fact table, and freight-to-revenue ratio analyses by carrier-and-route combination.

5. Result

The finding can be stated with precision:

Given a GRAM-compliant schema in which every binary fact type carries explicit uniqueness and mandatoriness constraints on each role, a large language model consumes the schema with equivalent text-to-SQL accuracy whether the predicate verb is descriptive (e.g., "handles," "submits," "ships to") or generic (e.g., "has," "contains"). The lexical content of the predicate is operationally inert relative to the constraint modality.

The result was observed first on Gemini 2.5 Flash, a smaller and less capable model than the largest frontier offerings, and was subsequently confirmed in agentic deployment on Claude Opus 4.7. The corollary is that the semantic load of a binary fact type, as consumed by an LLM, is carried by the constraint metadata—the uniqueness flag on each role, the mandatoriness flag on each role, and the cardinality classification they jointly determine—and not by the natural-language reading.

That the result holds on Gemini 2.5 Flash is consequential. If the finding had been observed only on a top-tier frontier model, the natural objection would be that model capacity is doing the work—that any sufficiently large model could infer the verb's contribution and ignore it. The Flash result is harder to explain away. A model in the middle tier of current capabilities, consuming a schema labeled with nothing but "has" across every relation, produces accurate SQL. This shifts the explanation from model capacity

to representation: the schema is being read correctly because what the schema declares is sufficient to read it correctly.

6. Why the Verb Is Inert

The result becomes intelligible once the inference burden of the LLM is decomposed by layer.

At the linguistic layer, the LLM must interpret the user's natural-language question and map its terms to concepts in the domain. A model trained on corporate-scale corpora arrives with extensive prior knowledge of the conceptual vocabulary of business domains. It knows what a customer is, what an order is, what fill rate means, what a sales region is, what days-sales-outstanding refers to. The linguistic-inference task, for any reasonably capable model operating in a known domain, is substantially solved before any schema is consulted.

At the schema layer, the LLM must map the linguistic concepts onto the structural elements of the database—which tables represent which entities, which columns carry which values, which joins are valid, which relationships are mandatory, what the cardinality of each relation is. This is the layer at which DDL-only systems fail. The DDL describes the physical mechanism of storage; it does not declare the semantic structure of the relation. The LLM must infer the structure, and inference under uncertainty is the source of the failure modes that characterize text-to-SQL systems prompted with DDL alone.

GRAM addresses the schema layer by making the structural elements explicit. The constraint metadata declares the cardinality, the participation, and the functional dependencies of each relation. These declarations—not the verb—are what the LLM consumes when it generates a query. The verb appears in the JSON, but the verb does not determine the JOIN type, the GROUP BY structure, the WHERE clause, or the aggregation pattern. The constraints determine all of these. The verb is a label on a relation whose semantic structure has already been fully specified by the constraints.

An LLM presented with a relation labeled "Employee has Order" and a relation labeled "Employee handles Order," with identical constraint metadata in both cases, generates identical SQL. The verb is read but does not influence the output. Substituting "has" for every verb across the schema therefore produces no degradation because no information is being removed; only redundant labeling is being removed.

The descriptive verb does not contribute operationally for the same reason a comment in code does not contribute operationally: it is information for human readers, not for the executing system.

7. Why This Contradicts the ORM Position

The Object-Role Modeling tradition holds that the natural-language verbalization of a fact type is an elementary semantic unit, not merely a label. The position is justified by the role verbalization plays in

pedagogy and in domain-expert validation: a fact type that can be verbalized in a natural sentence is held to be a properly atomic fact, while one that cannot is held to be improperly composed.

Under the finding reported here, the verbalization's pedagogical role is preserved—domain experts continue to benefit from being able to read and validate "Employee handles Order" as a sentence about their business. The operational role, however, is not preserved. The LLM that consumes the schema does not benefit from the verb. It benefits from the constraint structure.

The two roles are therefore separable, and the tradition has historically conflated them. The conflation was harmless when the consumer of an ORM schema was a CASE tool generating DDL—such a tool ignored the verbalization entirely and operated only on the constraint structure. The conflation becomes visible only now, when the consumer is a language model, because the language model is the first machine consumer capable of reading the verb but choosing not to depend on it. The verb has always been operationally inert; the LLM is the first reader sophisticated enough to demonstrate the fact.

8. Implications

8.1 For Automated Schema Generation

Tools that generate machine-readable semantic layers from source databases—whether GRAM, dbt semantic models, Cube schemas, or other formats—need not invent or curate descriptive predicate verbs in order to support accurate LLM-driven text-to-SQL. A generic predicate, applied uniformly, is sufficient when the constraint modality is preserved. This collapses what would otherwise be a difficult generation problem (the production of business-appropriate verbs from source metadata that contains no such information) into a trivial one (the emission of a default label). The implication for tooling is direct: the verb-generation step, which would require a model-in-the-loop or extensive human curation, can be omitted entirely without affecting downstream consumption.

8.2 For the Semantic Layer Movement

Semantic-layer initiatives that have invested in human-meaningful labels and descriptive verbalization as the principal mechanism by which machine consumers derive analytical meaning are investing in the wrong layer. The labels assist the humans who author the semantic layer. They do not contribute to the machine's ability to consume it. The investment that does contribute is the explicit specification of constraint modality—cardinality, participation, functional dependency—at the relation level. Where current semantic-layer products specify these properties at all, they typically specify them informally and at the join level rather than at the predicate level. The finding suggests this is the layer where the field should be focusing engineering effort.

8.3 For LLM-Integrated BI Tools

Approaches that integrate LLMs with pre-existing BI semantic models inherit a structural limitation that the finding makes precise. The BI semantic model expresses tables, measures, and relationships in forms designed for human authors. The constraint modality that the LLM requires for accurate consumption is not specified in these models in the formal manner the finding identifies as load-bearing. As a consequence, the LLM is forced back into schema inference, and the failure modes of DDL-prompted systems re-emerge in attenuated form. The architectural ceiling of these products is bounded by what the underlying BI semantic model declares, and that declaration was not designed to support formal logical reasoning.

8.4 For Evaluation of Text-to-SQL Systems

Benchmarks that evaluate text-to-SQL accuracy against schemas presented as DDL or as schemas with informal human-readable labels are not measuring the upper bound of LLM performance on structured data. They are measuring the LLM's ability to infer structure from incomplete information. A separate evaluation, against schemas that present constraint modality explicitly, would measure the LLM's ability to consume structure when it is given. The difference between the two measurements is the inference burden imposed by the prevailing representation, and is large.

8.5 For the GRAM Specification

The natural-language reading remains a useful element of the GRAM specification, for the same reasons it remains useful in ORM: it supports human authorship, review, and validation. Its status, however, should be reframed. It is metadata for human consumers; it is not signal for machine consumers. The constraint modality—uniqueness, mandatoriness, and the cardinality classification they determine—is the operationally load-bearing content of the specification. A future revision of the specification will reflect this distinction explicitly.

9. Limits of the Finding

The finding is reported for LLMs operating in business domains substantially covered by the model's training corpus—supply chain, finance, customer relationship management, human resources. Two boundaries of the finding are worth naming explicitly.

First, the result has been confirmed on models ranging from Gemini 2.5 Flash to Claude Opus 4.7. It may not extend to models smaller or less capable than Gemini 2.5 Flash, where weaker prior knowledge of corporate-domain vocabulary may force the model to rely more heavily on lexical signal. The boundary is empirical and the test is straightforward; the finding is reported for the capability tier in which most contemporary enterprise deployments operate, not for all language models.

Second, the result may not hold for domains with specialized terminology poorly represented in general-purpose training corpora. A query against a clinical-trial schema using domain-specific verbs

("administers," "randomizes," "withdraws from") may benefit from descriptive predicates in ways a corporate-business schema does not. Where the LLM's prior knowledge is weak, predicate verbs may begin to matter. Within the corporate-domain regime in which most enterprise text-to-SQL deployments operate, the finding is robust across the test corpus described in Section 4.

10. Conclusion

The natural-language verbalization of a binary fact type is a teaching aid for human authors and a comfort to human reviewers. It is not the carrier of the semantic content that an LLM consumes when it generates SQL from a GRAM-compliant schema. The carrier is the constraint modality: the uniqueness on each role, the mandatoriness on each role, and the cardinality classification they determine.

The implication for the field is direct. The investment that produces accurate machine-readable semantic layers is the investment in formal constraint specification, not in human-readable labeling. The dominant industry approaches—DDL-prompted text-to-SQL, LLM integration with pre-existing BI semantic models, and the curated-label school of the semantic-layer movement—all leave the schema-inference burden on the model in different proportions. A predicate-based approach with explicit constraint modality removes that burden. The further refinement reported here—that the verb itself is unnecessary within such an approach—reduces the authoring cost and clarifies the architectural principle.

Halpin's instinct—that the conceptual schema must be expressed in something more than physical metadata—was correct. The specific form he proposed—that the something more is natural-language verbalization—requires revision. The something more is the formal modality of the predicate, expressed as constraints on roles. Verbalization travels with it as scaffolding for humans, but is not itself the operational substance. A binary fact type, for the purposes of machine consumption by an LLM, is fully specified by the entities it relates and the constraints that govern their participation. The verb is decoration. The grammar is everything.

References

Codd, E. F. (1970). A Relational Model of Data for Large Shared Data Banks. Communications of the ACM, 13(6), 377–387.

Chen, P. P. (1976). The Entity-Relationship Model—Toward a Unified View of Data. ACM Transactions on Database Systems, 1(1), 9–36.

Halpin, T. (2006). Information Modeling and Relational Databases. 2nd ed. Morgan Kaufmann.

Kench, E. (2026). GRAM Specification (v1.0): The Formal Standard for Graph-Rule Analytical Mapping. Ormle.