From Technical Data Model to Logical Business Data Model

Research Summary

Robert Blok

June 25, 2026

From Technical Data Model to Logical/Business Data Model: A Research Summary

Background and Problem Statement

Organizations frequently inherit technical databases — relational schemas shaped by implementation constraints, surrogate keys, normalization decisions, and years of drift — with little or no surviving documentation of what the data actually *means*. Recovering a logical or business data model from such a technical artifact is the central problem addressed by two overlapping research traditions: **database reverse engineering (DBRE)** and **NLP/AI-driven semantic extraction from documentation**.

Part 1: Database Reverse Engineering (DBRE)

The Core Two-Step Framework

The foundational conceptual model, developed by Hainaut and colleagues at Namur University in the late 1990s, describes DBRE as two sequential processes:

1. **Data structure extraction** — reconstructing the DBMS-level logical schema from physical files, constraints, and application code.
2. **Data structure conceptualization** — recovering domain-meaningful semantics from that logical schema.

The second step is what produces a logical or business data model. The hard part is that physical schemas routinely lose semantic information during forward engineering: naming conventions collapse, business rules become obscure integrity constraints, and the distinction between natural and surrogate keys disappears.

Extracting Entity-Relationship and Conceptual Models

The most common target is an ER or Extended ER (EER) diagram. Key approaches include:

- **Andersson (1994)** — analyzes SQL application code (join conditions, view definitions) to infer implicit foreign-key relationships and reconstruct an ER schema. This remains one of the more-cited works in this space.
- **Chiang (1993)** — built the Knowledge Extraction System (KES), combining schema analysis with data-instance analysis and expert consultation to generate EER schemas, explicitly flagging cases requiring human judgment.
- **Harsh et al. (2025)** — a recent automated approach that reads relational metadata (primary keys, foreign keys, nullability) to generate ER elements without needing application code.
- **RDB2 Model (Widad & Bahaj, 2017)** — a two-step retro-design method that extracts data structures and then enriches the resulting conceptual model with relational constraints to improve semantic understanding.

Ontology as Target Representation

A parallel line of work uses OWL ontologies rather than ER diagrams as the output. Tools like **DB2OWL** map relational constructs (tables → classes, columns → properties, foreign keys → object properties) according to formal transformation rules. Lubyte & Tessaris (2007) proved formally that for schemas in third normal form (3NF), their extraction procedure preserves all relational constraints without data loss — a useful formal guarantee for high-stakes migrations.

Model-Driven Approaches (MDA)

The Model-Driven Architecture community has formalized these transformations using metamodels and transformation languages (ATL, QVT). Ristic et al. at Novi Sad implemented a model-driven pipeline in IIS*Studio that takes a generic relational schema through a chain of model-to-model transformations to produce a domain-specific conceptual model. Mazon & Trujillo (2008) applied the same MDA approach to data warehouse reverse engineering, deriving a

conceptual multidimensional model from operational source schemas.

The Persistent Challenge: Semantic Gap

Physical schemas encode business meaning in ways that are ambiguous, lossy, or wholly absent — especially in legacy systems. No fully automatic approach works without some human input, particularly for naming and business-rule recovery.

Part 2: AI/NLP Mining of Customer Documentation

Classical NLP: Requirements and Use Cases → ER/UML

A well-established research thread uses NLP to extract conceptual models from natural language requirements specifications and use cases. The core idea is consistent: identify **nouns** as candidate entities, **verbs** as relationships, and **adjectives/modifiers** as attributes, then apply heuristic rules to filter and assemble them.

- **Btoush & Hammad (2015)** — maps Part-of-Speech tags to ER elements using predefined syntactic rules, applied to written requirements documents.
- **Sagar & Abirami (2014)** (*Journal of Systems and Software*) — uses typed grammatical dependencies combined with OO design principles to extract EER-notated class models from functional specifications, validated against human-generated models.
- **Arora et al. (2016)** (ACM/IEEE MODELS, 125 citations) — the closest to a real industrial test: combines software engineering heuristics with information-retrieval rules and modern dependency parsers, applied to four industrial requirements documents with an expert accuracy study. Key finding: the extractor produced useful candidate elements but required expert review.
- **Javed & Lin (2020)** (*Information and Software Technology*) — handles heterogeneous input formats (general requirements, use case specs, user stories) and extracts both ER models and business process models simultaneously.

LLMs for Ontology Learning and Concept Extraction

LLMs significantly raise the capability ceiling for this problem:

- **LLMs4OL (Giglou et al., 2023)** — benchmarked nine LLM families zero-shot on three ontology learning tasks (term typing, taxonomy discovery, non-taxonomic relation extraction) across WordNet, GeoNames, and UMLS. LLMs performed well on term typing but struggled with complex relational structures.
- **OLLM (Lo et al., NeurIPS 2024)** — fine-tunes an LLM to generate entire ontology subcomponents rather than individual relations, outperforming subtask composition approaches on both semantic accuracy and structural integrity.
- **OmEGa (Shim et al., 2025)** (*Advanced Engineering Informatics*, Q1) — uses LLMs with a predefined Task-Centric Ontology to perform instance recognition and relation classification over multimodal manufacturing documents (text, images, tables). Strong proof of concept for mining heterogeneous technical documentation.
- **Multi-agent LLM framework (Pan et al., 2026)** — applied to Ethernet switch configuration manuals, achieving extraction correctness of 0.97–0.99 across three extraction tasks using an iterative Extract-Evaluate-Improve loop.

The Consistent Finding: Human-in-the-Loop is Necessary

Both classical NLP and LLM approaches degrade on hard cases — exactly where human judgment matters most. Classical systems require relatively structured input; inconsistent writing or domain synonyms cause recall to drop sharply. LLMs handle variability better but introduce hallucinated relationships, conflation of similar concepts, and inconsistent entity naming across document sections.

The practical architecture most researchers converge on is **semi-automatic / human-in-the-loop**: AI generates candidate entities, attributes, and relationships from the corpus; domain experts validate and correct.

Part 3: Combining Both Approaches

The natural synthesis — running DBRE from the schema *and* NLP extraction from documentation in parallel — is compelling precisely because the two pipelines provide independent evidence:

- **Agreement** between what the schema implies and what the documentation names is a high-confidence signal for a correct semantic mapping.
- **Divergence** (schema entities with no documentation trace, or documented concepts with no schema counterpart) flags exactly what needs human attention.

Using mention frequency across documents as a confidence weight (an entity mentioned 20 times is higher confidence than one mentioned once) allows expert effort to be directed efficiently.

Key Takeaways

Aspect	Status
DBRE as a research field	Well-established since the 1990s; standardized frameworks exist
Ontology extraction from schemas	Formally proven to be lossless for 3NF schemas; tooling available
NLP from requirements / use cases	Active research; industrial validation limited but exists
LLMs for ontology/concept extraction	Rapidly maturing; strong results in specialized domains (2023–2025)
Fully automatic pipelines	Not yet reliable; human-in-the-loop remains best practice
Combined schema + documentation mining	Logically sound; not yet studied as an integrated approach in the literature found

---

Based on an initial search across database reverse engineering, NLP-driven model extraction, and LLM-based ontology learning literature. Deeper searches on RAG-augmented schema enrichment and recent LLM-assisted data modeling tools would likely surface additional relevant work.