

Development of MBD Method for Global Design Optimization Using Cross-Domain Model

- (3rd Report) A Method for Building LLM Agents through a Consolidated Plaintext and Progressive Writing-down of Tacit Knowledge -

Shigemitsu Takahashi¹⁾ Aoto Utsumi¹⁾ Masayuki Kiyono¹⁾

1) Nissan Motor Co., Ltd. 560-2 Okatsukoku, Atsugi-shi, Kanagawa 243-0126, Japan

KEY WORDS: Vehicle development, LLM agents, Tacit knowledge, Human-AI collaboration, Robustness [B2]

Engineering knowledge for operating CAE systems lives largely as the tacit judgment of skilled engineers, and is lost when they leave. Large language models (LLMs) can help transfer this knowledge, but they are probabilistic and cannot simply replace engineers. Conventional agent architectures also scatter what must be inherited across tool servers, retrieval stores and separate documents, and presuppose that knowledge is written down in advance - which tacit knowledge does not allow.

This report proposes a method for building LLM agents that take on work relying on such tacit knowledge, and applies it to the operation of the cross-domain model CAEM of the earlier report. Knowledge, procedures and tools are folded into a consolidated plaintext, FUMI (Flat Unified Markdown Interface): a single collection of Markdown documents with reference links, which humans and the LLM read and write in the same way (Fig. 1). The state of the work resides in FUMI, and the LLM is stateless and loosely coupled, so work continues even when the LLM is unavailable. Agent roles are also documents on FUMI; one LLM composes different agents by changing what it reads.

Tacit knowledge is written down progressively during operation rather than up front. Legacy materials are ingested as Markdown or SQL-queryable tables, and work is delegated as it is. When premises are missing, the agent stops and asks (human-in-the-loop). When engineers point out flaws - far easier than writing tacit knowledge down from scratch - the LLM reflects on why it erred and updates FUMI so the error does not recur. Routinized processing is lowered into deterministic scripts written by the LLM, progressively peeling roles away from the LLM. Hooks that fire before execution refuse operations that cannot be undone.

Standard RAG fails here structurally, not for want of tuning. It splits documents into chunks, embeds them, and passes the closest to the question: a procedure is cut at a chunk boundary, a table is flattened so rows and columns no longer correspond, and distance says nothing about which document is current. FUMI is read instead by starting at the index, following reference links, and matching part numbers and abbreviations lexically; a document that matches is read whole. The index and the links are text that engineers and the LLM both read, so replacing the model does not mean rebuilding them.

The method was applied to months of CAEM operation. Bootstrapping took days of feedback in which missing premises and misread figures were written back. An engineer unfamiliar with both CAEM and CAE completed a towing-capacity study in about three days without consulting experts - a task of about one hour for an expert - showing that accumulated procedures and recorded rationale can stand in for direct consultation. Work continued whether the human or the LLM was absent. Remaining issues include dependence on frontier-class models and keeping LLM-written text readable to humans.

Corrections ran in both directions over FUMI. In a fuel-economy sensitivity study the LLM misread a motor parameter definition; an engineer pointed it out, and the LLM surveyed the model and wrote the missing definition back. In the other direction it caught an engineer confusing rows with columns against what FUMI recorded. Debugging an algebraic loop from coupling two models, it found the cause and inserted a corrective block after two or three pieces of feedback, without the engineer opening the model.

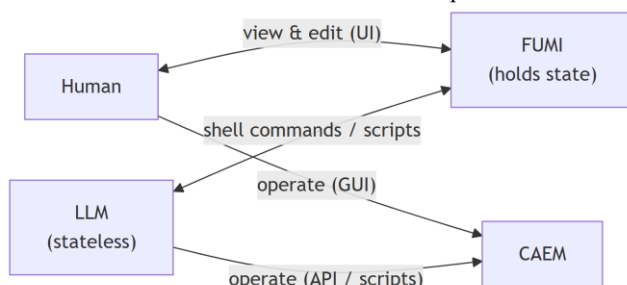


Fig. 1 Architecture of the proposed system

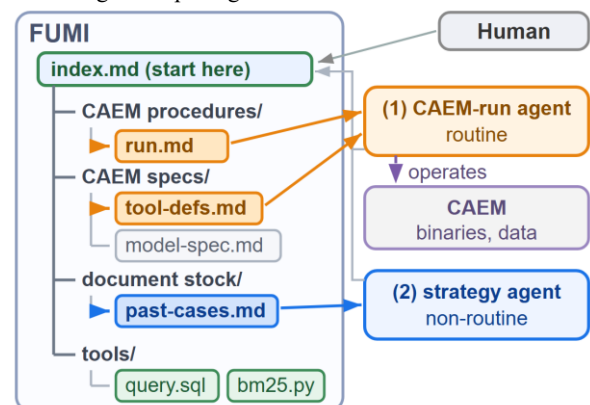


Fig. 2 Agents defined on FUMI