# Natural Language AI Interfaces for Tunnel Monitoring Data

By Rich Laver and Angus Maxwell of Maxwell Geosystems Ltd.

**As instruments** become more sophisticated, more automated and the cost of installation falls engineers and geologists are now able to monitor the tunnelling process like never before. On the London and Birmingham sections of the HS2 project in the UK, tunnelling close to key national infrastructure has required over 150,000 instruments to be deployed resulting in 2.5 billion records growing at 65 million per month. Whilst this undoubtably reduces risk, the sheer quantity of data presents a significant challenge to the end user to assimilate, identify trends and make reliable judgements.

On the Tung Chung Extension part of the Lantau Portfolio Projects for the Hong Kong MTRC, Maxwell Geosystems and Engineering Surveys have been appointed as the Independent Monitoring Consultant (IMC). The company's role is to provide an independent review of monitoring data through the collection of independent readings and through the provision of an independent software platform for the collection, processing and interpretation of the data. With a view to enhancing this role Maxwell Geosystems has developed a proof of concept natural language AI interface empowering stakeholders to access data on the fly.

## Rapid exploration of site data is now a reality

Big data platforms typically require users to navigate through a user interface to find what they need. Users often miss crucial insights because visualisations are predefined and capture only one particular perspective of the complete picture. Users may be isolated from their workstations often down the tunnel and need rapid curated feedback on what is going on.

The multi-agent AI gives users direct access to the data and the visualisations they need. It furnishes a natural language interface through which users can interact with site data, extract data and generate visualisations on demand to precisely address their request without the need for prior training in the larger software systems..

The development initiates a move towards more direct and immediate access to data than is possible through standard graphical user interfaces, and opens the door to fully-automated data exploration and reporting.

The AI generates a response by chaining multiple agents together. Each agent invokes a powerful Large-Language Model (LLM).

We overcome two fundamental challenges faced when building human language "chat" apps:

1. Feeding each LLM all that it needs to perform its task
2. Optimising performance in generating the response, namely in its accuracy and latency

## Teaching what we take for granted

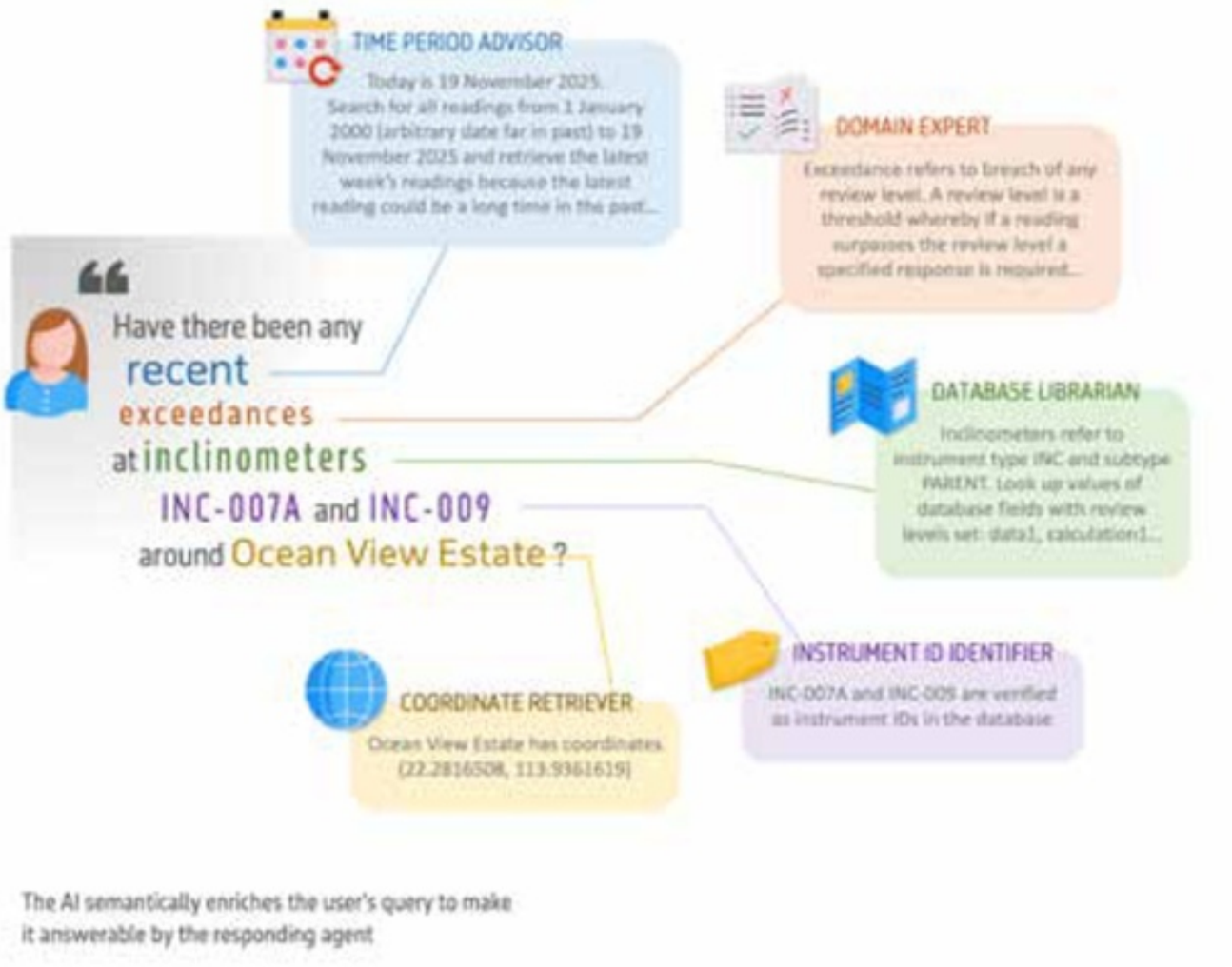
Foundation LLMs excel at reasoning, but they still require additional information called context to answer a data exploration query. The LLM might need to know many things before it can respond:

- Semantics on instrumentation monitoring
- The date today
- How to handle missing readings
- Where to find data in the database
- Which tokens in the query are instrument names
- The location of places named by the user

Feeding too much context will confuse the LLM so the AI intelligently selects only the context required to answer the query. Context selection is implemented through a multi-agent architecture comprising an orchestrator agent with access to worker agents. Each worker agent returns a different aspect of context when requested by the orchestrator. Worker agents are called in parallel where possible so that within a few seconds the AI retrieves the precise context needed to craft a response.

## How to navigate a database with no signposts

A significant breakthrough for the AI is its ability to enrich the database with labels and semantics. Many databases have fixed schemas ie their fields and nomenclature are predefined. Because they are fixed, they cannot adapt to changing requirements so




The AI semantically enriches the user's query to make it answerable by the responding agent

more powerful systems grant users freedom to define any instrument and any data field, robustly catering for the diverse spectrum of instruments and use cases we find in real world projects. The corollary of this flexibility is the inability for an AI to reliably identify instruments and data fields in the database. To illustrate, a query such as "How much did groundwater levels drop yesterday?" requires the responding agent to first find which instruments monitor groundwater level and then find which data field stores groundwater elevation. Without consistent labelling, the agent is left to guess.

The AI overcomes this by robustly generating database labels and semantics through an offline context enrichment tool. The tool orchestrates a workflow which employs LLMs to thoroughly comb the database, looking for hints giving meaning to user-defined instruments and data fields, for example naming patterns, measurement units, and formulae. The tool then generates a document comprehensively describing when and how to reference every single instrument and data field in the database.

When a user submits a query, an agent intelligently feeds only relevant snippets of the document to the responding agent, providing essential signposts on where to find data whilst keeping the context clean of irrelevant information.

An exciting byproduct of this is the ability of the AI to point to many different data bases and using these semantic tools extract data reliably without prior knowledge of the schema. This opens up opportunities for simpler data aggregation and warehousing giving increased democratisation of data amongst the community or business.

## No unauthorized entry!

Project users often require different access rights to data depending on their role. It is imperative that users are strictly blocked from querying data they are restricted from viewing.

One approach is to instruct the responding agent to enforce user permissions. Unfortunately LLMs are never 100% reliable. Instead the AI guarantees strict enforcement by consistently implementing a tooled guardrail. The guardrail is entirely rule-based and adds a permission filter consistently to any extraction query executed on the database with 0% failure rate. The AI thereby ensures that users can never access data they are not permitted to access.

## A picture is worth 1000 tokens

Site data is by nature spatiotemporal in that it has both a location and a timestamp. The human mind is accustomed to visualising spatiotemporal variations and so most site data platforms utilise high-impact graphics effectively to draw attention to patterns and anomalies.

The AI takes this one step further: the responding agent is equipped with highly configurable tools to generate plots. Once the agent has discerned the user's underlying need the agent judiciously creates the plots precisely conveying the insights the user is after. The AI is capable of supplementing answers with the visualisations built without need access to the underlying source application functionality, including: Time series data from multiple instruments with superposed review levels; map views of readings and their changes, map views of review levels and their changes.

## Users demand accurate responses in seconds

Users will judge the performance of any AI primarily on two conflicting metrics:

Accuracy: users expect to ground safety-critical decisions on the data they get.

Latency: users expect a response within tens of seconds at most.

In creating the AI we applied best-practice techniques to maximise response accuracy and quality without significantly impacting response generation time.

To optimise this the AI generates up to three candidate responses per query and intelligently selects the best response as the final answer. This approach cuts random error markedly. For example, for a single agent giving 30% random errors, combining three agents would reduce random error to just 3%. We execute candidate responses in parallel to minimally impact latency.

## The power of code execution

For anything but the simplest query we grant the responding agent power to write its response plan and execute it as code. This not only forces the agent to think methodically but also endows it with advanced computational capabilities to competently tackle complex or data-heavy queries.

Large volumes of extracted data would practically freeze a conventional reasoning agent as it painstakingly regenerates data token-by-token between thinking steps. In contrast executed code only outputs the final result, with all intermediate steps isolated from the agent.

We reserve the latest state-of-the-art reasoning and coding models for response generation whilst selecting smaller but faster models for more straightforward tasks.

## Ask and it will be received

Tools empower LLMs with specific abilities, for example to extract data they need or to plot it. A common problem is that an LLM does not always call the right tool with the right inputs. We overcame this by developing robust tools with near-100% success in calling.

Tools typically require the calling LLM to populate a precise input schema—a hit-and-miss process prone to error. We crafted our tools to instead accept a natural language input from the LLM—granting the LLM more freedom. A smaller, faster and dedicated LLM inside the tool then translates the input into the tool's input schema. We boosted performance with error feedback so that the LLM can self-correct, and we found this approach resulted in consistently correct tool calls.

## Just like FedEx

We allow users to extract an extensive volume of data by serving a downloadable CSV file along with the response summary. The CSV file together with any generated plots comprise the artefacts supplementing the response.

Left to its own devices our LLM would handle these large artefacts slowly and expensively—clogging its context window and on final output consuming the token budget along with the user's patience. We get around this by providing the AI with access to an external artefact store. Instead of returning the artefact to the LLM, any artefact generating tool will automatically deposit it in the store and return a unique identifier for the artefact along with a concise description. The agent drafting the final response reviews the description to accurately refer to the artefact in its answer. Meanwhile a retrieval tool in the user interface looks out for the identifier so that it can serve up the right artefact from the store. This artefact management process delivers near-immediate rendition of megabyte-sized plots and files.

## The first step to offloading man years

In this article we present a proof of concept natural language AI interface for interacting with large volume, high throughput site data. The development yields rapid and precise responses enriched with tailored visualisations. The AI leapfrogs the conventional user interface necessary to navigate current platforms and customises responses to align with the user's underlying need.

The work paves the way for broader queries such as "Anything unusual happened on site today?" or "What caused the sudden settlement near the diaphragm wall yesterday?". Such queries demand more immersive agent interactions with site data that would result in unprecedented time savings compared with manual data exploration.