

Natural Language Query for Extracting OpenStreetMap Data Using MCP and QGIS

Student

Monika Lötscher

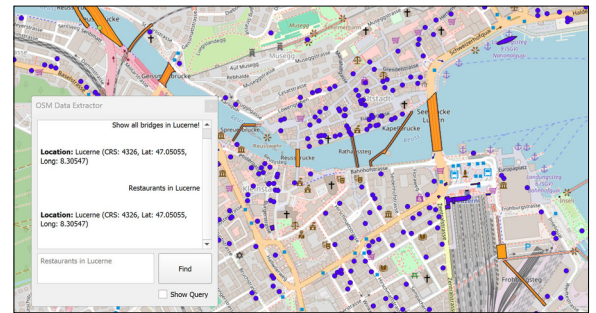
Objective: This project evaluated whether the Model Context Protocol (MCP) could provide QGIS users with a chat-style, natural language interface for creating simple, interactive maps visualizing data from OpenStreetMap (OSM). QGIS is an open-source cross-platform Geographic Information System (GIS) that allows users to create, visualize, analyze, and publish geospatial data. A proof of concept (PoC) of a QGIS plugin was implemented in Python which combines the former multi-step process of retrieving OSM data into a single query like "Show me all restaurants in Lucerne". This feature lowers the barrier of entry for novice QGIS users and offers experienced users a simple solution for the first steps of GIS analyses. The objectives can be met by an agentic, MCP-based AI application which employs an off-the-shelf Large Language Model (LLM) for the extraction of the necessary information from textual user input, but also for the orchestration of the processing steps like geocoding and composing queries to the OSM database.

Approach / Technology: Agentic AI applications are goal-driven systems which use an LLM to decide on actions and iteratively evaluate the outcome of each step before planning on the next steps. MCP is an open standard which defines how LLM-based applications connect to tools, resources, and prompts. It supports agentic AI by providing agents with a uniform way to discover and call tools and update context. Agentic AI applications can be categorized as AI agents or agentic workflows, with mixed-type applications in-between. The plugin can be considered an AI agent because of the dynamic planning of process steps, but it also has characteristics of an agentic workflow, because some steps, like retrieving and visualizing data, are unconditional. The MCP tools implemented are `get_location`, which is responsible for geocoding, `get_tags` to identify the appropriate OSM features, `get_query` to combine the results of the former two into a query to the OSM database, and `notify_user` to communicate errors and results. An iterative and incremental practice was selected for the project, which helped in gradually evolving functionality. For the implementation of MCP, the FastMCP library proved helpful because of its high-level API providing all required features.

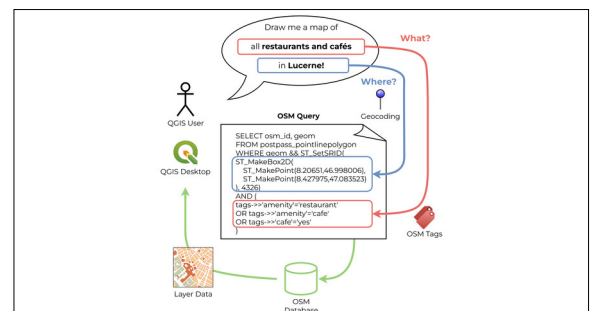
Result: A technology overview of GIS-oriented solutions involving LLMs revealed many products and projects concerned with natural language input methods. QGIS plugins featuring the download of OSM data without LLM involvement are "Quick OSM" and "OSMDownloader". The PoC QGIS Plugin, named «OSM Data Extractor», implements the specified use case, its main weakness being that the generated OSM queries still lack semantic precision, which can lead to irrelevant data being retrieved. On the technical side, MCP was found beneficial for this

type of application and functionality, since it allows expanding towards a more feature-rich agent or assistant. For the implementation of a production-ready QGIS plugin with natural-language input, concepts of the PoC like the architecture based on MCP can be re-used, whereas features like query generation and OSM tags identification require further research and conceptual work. Other desirable features that have yet to be implemented are better handling of locations, such as the current map extension or punctual entities plus radius, and a short-term memory of the conversation. This would allow implicit reference to previously mentioned locations or feature classes: see chat in fig. 1.

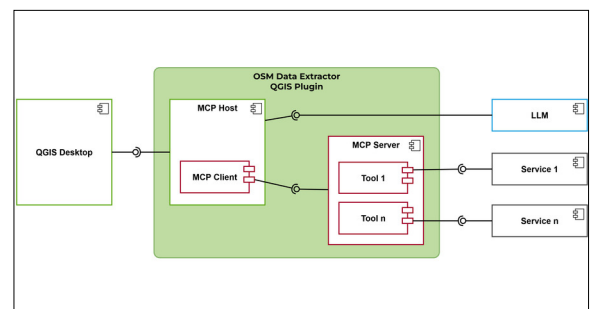
Chat of the QGIS Plugin "OSM Data Extractor" and a map in QGIS with retrieved OpenStreetMap features
Own presentation



Flow of information from the user through the QGIS plugin to the OSM database and back to the user as map features
Own presentation



Component design and integration with peripheral systems - MCP host and MCP tools are custom implementations
Own presentation



Advisor
Prof. Stefan F. Keller

Subject Area
Data Science