

MCP Server for INTEND 5G4Data use-case

Introduction

The 5G4Data use-case in the INTEND project is described in more details [here](#). We will in this document explore if it is possible and useful to create an MCP server for the 5G4Data use-case that can be used by the conversational tools in the INTEND toolchain for the 5G4Data use-case (i.e, used by inChat and/or inSwitch). More details related to MCP servers can be found [here](#).

Background

Model Context Protocol

The Model Context Protocol (MCP) is an open, client–server standard that lets AI systems discover and use external capabilities in a uniform way. MCP organizes those capabilities into three primitives: **tools** (actions the model can call with typed inputs/outputs), **resources** (read-only artifacts the model can fetch as authoritative context), and **prompts** (server-hosted, parameterized templates that standardize reasoning patterns).

In the INTEND project’s 5G4Data use-case, we already possess the TM Forum Intent toolkit, including a full intent ontology (TIO), validation assets, and TMF921 schemas. An MCP server can expose these assets cleanly:

- As **resources**: versioned TIO RDF/OWL, SHACL shapes, TMF921 JSON Schemas, KPI catalogs, and worked examples that the agent can load on demand for grounding and traceability.
- As **tools**: callable functions such as ontology lookup and disambiguation, SHACL-based ontology validation, TMF921 schema validation, intent synthesis/generation, and negotiation/feasibility checks against intent handlers.
- As **prompts**: curated templates for NL intent → slot extraction, mapping to TIO IRIs, normalization (units/time windows), and negotiation summaries to ensure consistent transformations across environments.

Introducing an MCP server in this context would decouple model reasoning from backend specifics, improve governance (versioning and audit of ontology artifacts), and enable interoperable, repeatable conversion of natural language expressed business intents into TMF292 (intent ontology) and TMF921 (intent management API) compliant business/service/resource intent descriptions suitable for the 5G4Data scenario.

Scope limitation

For a Minimal Viable Scenario (MVS) implementation we could make some limitations to simplify developing a Proof of Concept (PoC). Some examples could be:

- Introduce restrictive dialogue templates (to simplify extraction of needed information given the limited scope)
- Create algorithm and templates for how to generate slice creation and deployment configurations intents in TM Forum format by inserting values in slots in predefined templates
- Generate SHACL shapes for validation purposes (for the limited structure of slice and deployment intents)

MCP Server outline

In this chapter we will outline how the three different assets (prompts, resources and tools) of an MCP server could be designed to help conversational INTEND tools (clients) to do their business.

Prompts

The Model Context Protocol (MCP) provides a standardized way for servers to expose prompt templates to clients. Prompts allow servers to provide structured messages and instructions for interacting with language models. Clients (like inChat and inSwitch) can discover available prompts, retrieve their contents, and provide arguments to customize them. In the context of the 5G4Data use-case, we could create prompts that:

- Guides the inChat dialogue with the user, e.g. “Greeting message” that asks the user about which use-case you want to explore or that tells the user which use-case it is currently connected to or which use-case it can support.
- Prompts that try to extract slice or deployment specific “slot” information for intent templates
- ...

More details about MCP Server Prompts can be found [here](#).

Resources

The Model Context Protocol (MCP) provides a standardized way for servers to expose resources to clients. Resources allow servers to share data that provides context to language models, such as files, database schemas, or application-specific information. Each

resource is uniquely identified by a URI. In the context of the 5G4Data use-case, we could provide resources like:

- Intent templates for slice creation and workload deployments
- The complete TM Forum intent ontology
- TM Forum intent related documents
- Our synthetic data (for the 5G4Data use-case)
- ...

More details about MCP Server Resources can be found [here](#).

Tools

The Model Context Protocol (MCP) allows servers to expose tools that can be invoked by language models. Tools enable models to interact with external systems, such as querying databases, calling APIs, or performing computations. Each tool is uniquely identified by a name and includes metadata describing its schema. For the 5G4Data use-case some feasible tools could be:

- Intent generation: Based on predefined “slots”, create a valid slice or workload deployment service or resource level intent based on predefined templates with open slots
- Validation: Based on SHACL shapes, the tool will validate the received TM Forum formatted intent
- TMF921 payload generation: Create a valid TMF 921 API payload
- Discovery: Based on intent handler meta data (i.e. capabilities of different intent managers), the discovery tool can guide the client to find the appropriate intent handler for generated intents.

More details about MCP Server Tools can be found [here](#).

Conclusion

Creation of a 5G4Data specific MCP server would represent a good “separation of concerns” and this strategy could be valid for other use-cases as well. Following this strategy, it will be easier to generate “generic” INTEND tools that implements the generic parts across all use cases and that will interact with use-case specific MCP servers to handle use-case specific functionality.

Future explorations

A 5G4Data use-case specific MCP server can be a good option to make life easier for the conversational tools in the INTEND project with relation to our use-case. The introduction of MCP servers for the other use-cases could also be of interest.

We have however also identified other needs for the composition of tools for the 5G4Data use-case, for example the problem of identifying/discovering feasible intent handlers for the 5G4Data use-case. This is also a general problem for the composition of tools for any use-case.

The INTEND tools can be regarded as “agents”, and we should probably investigate the A2A protocol to see how it can fit related to creating compositions of tools for a given use-case. The A2A protocol includes discovery mechanisms. We should also investigate how tools (agent) can negotiate how to interact. For our use-case, we have suggested to use TM Forum “Intent Management” 921 API. It could be that the A2A protocol could be used to negotiate an agreement on this, or that maybe even the A2A protocol itself could be used instead of the TMF921 API? Is it for example possible to negotiate about payload structure as part of the A2A protocol?