

NDS CONFERENCE · TECHNICAL DAY

Joseph Birkner
Klebert Engineering
14 October 2026 · Volvo Cars, Gothenburg

Using the NDS MapViewer as an MCP Server

Who is using what?

Show of hands

AGENT / HARNESS

Codex
Claude Code
Copilot
Cursor

Something else? Not yet?

MODEL / PROVIDER

OpenAI
Anthropic
Google

Company-managed, local, or not sure?

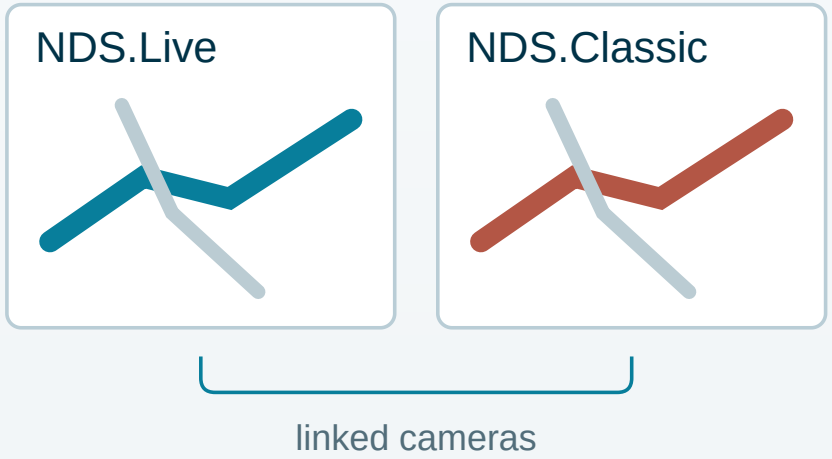
WHERE DO YOU USE IT?

Desktop app / Terminal
IDE / Browser

And who has already connected an MCP server?

Viewer control

“Zoom to Munich. Live on the left, Classic on the right.”



Illustrative view arrangement

01

Find Munich

Resolve the intended place.

02

Choose the maps

Check available coverage.

03

Arrange two views

Set layers and camera sync.

04

Check the result

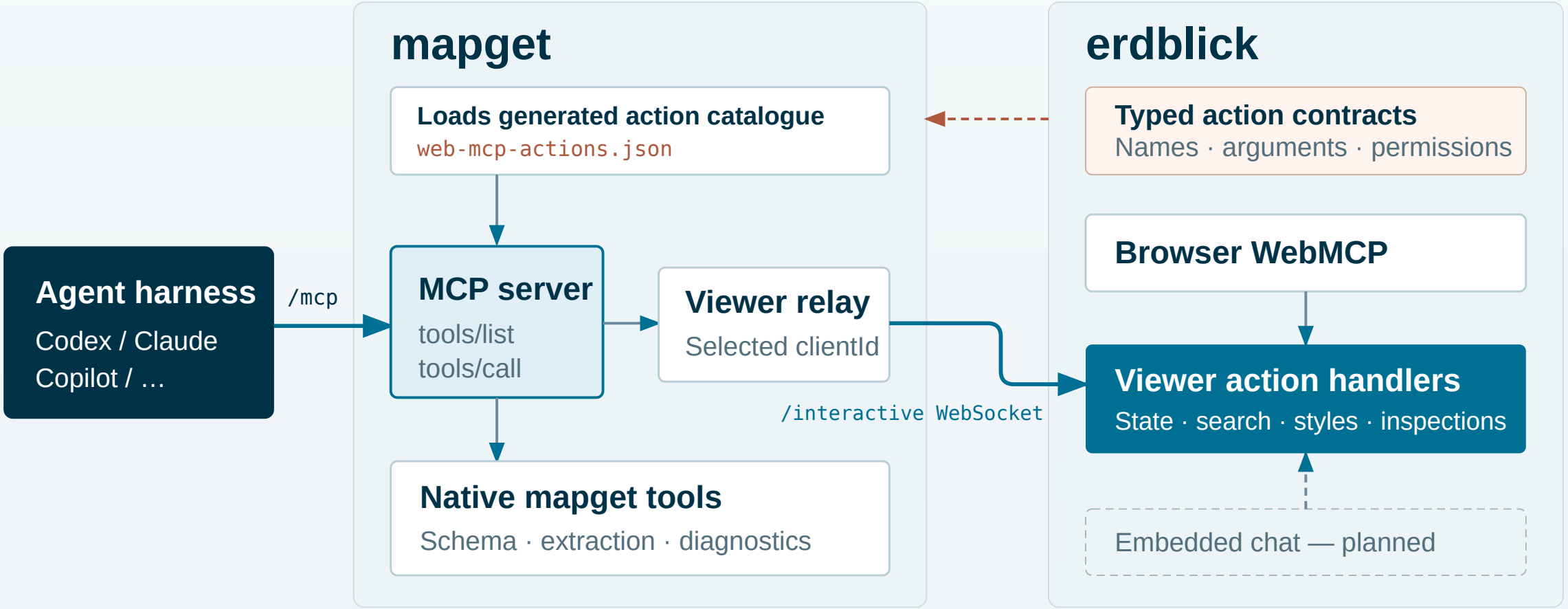
Read state and inspect the map.

Linked cameras can show different layers.

Open the viewer ↗

How erdblick tools become mapget tools

One remote MCP endpoint



Native tools run without an open tab. Remote viewer calls and browser WebMCP use the same action contracts.

The tool catalogue

Browse a group or select a tool

Group: All groups | Both | erdblick | mapget | All | Read | Write

42 tools

Discover & understand 06

- mapget_docs
- mapget_list_sources
- mapget_get_coverage
- mapget_query_schema
- viewer_get_catalog
- viewer_describe_app_state

Sessions, maps & location 09

- viewer_list_sessions
- viewer_get_app_state
- viewer_set_app_state
- viewer_manage_view
- viewer_navigate
- mapget_lookup_place
- mapget_get_place_geometry
- mapget_convert_coordinates
- mapget_convert_tile_id

Search & extract 09

- mapget_validate_expression
- mapget_extract_features
- mapget_extract_source_data
- viewer_start_search
- viewer_control_search
- viewer_get_search
- viewer_set_search
- viewer_get_search_results
- viewer_export_search

Author styles 03

- viewer_get_style
- viewer_validate_style
- viewer_edit_style

Inspect & share 05

- viewer_inspect
- viewer_close_inspection
- viewer_open_source_data
- viewer_screenshot
- viewer_get_share_link

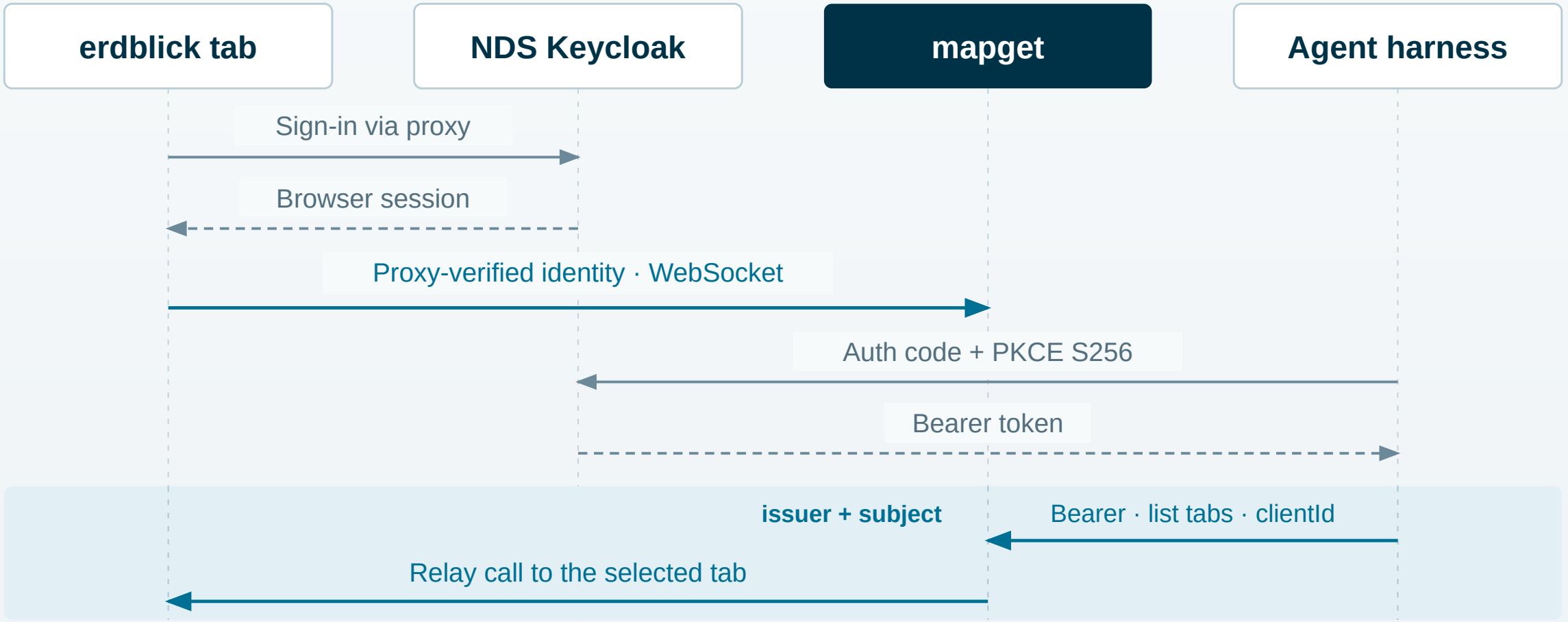
Operate the UI 06

- viewer_take_snapshot
- viewer_get_element
- viewer_click
- viewer_fill
- viewer_scroll
- viewer_resize

Diagnose & administer 04

- viewer_get_diagnostics
- mapget_get_diagnostics *
- mapget_get_config *
- mapget_set_config *

Which tab can the agent control?



Identity comes from verified claims, not an email in the URL. After reconnection, discover the current clientId again.

Munich. Live left, Classic right.

Example tool sequence

Tools for this question

Discover & understand

- `viewer_get_catalog`
- `viewer_describe_app_state`

Sessions, maps & location

- `viewer_list_sessions`
- `viewer_get_app_state`
- `mapget_lookup_place`
- `viewer_manage_view`
- `viewer_set_app_state`
- `viewer_navigate`

Inspect & share

- `viewer_screenshot`

01 Choose the tab

`viewer_list_sessions` `viewer_get_app_state`

02 Find Munich

`mapget_lookup_place`

03 Find both datasets

`viewer_get_catalog`

04 Create the second view

`viewer_manage_view` `viewer_get_app_state`

05 Assign layers and sync

`viewer_describe_app_state` `viewer_set_app_state`

06 Fit the comparison

`viewer_navigate`

07 Check both views

`viewer_get_app_state` `viewer_screenshot`

Check the result against the original question.

Synchronize the cameras, not the layers. Verify that both selected maps cover Munich.

Colour by speed limit

Example tool sequence

Tools for this question

Discover & understand

- `mapget_list_sources`
- `mapget_query_schema`
- `mapget_docs`

Sessions, maps & location

- `viewer_get_app_state`

Search & extract

- `mapget_extract_features`
- `mapget_validate_expression`
- `viewer_start_search`
- `viewer_get_search`
- `viewer_set_search`
- `viewer_get_search_results`

Inspect & share

- `viewer_inspect`

01 **Read the map context**
`viewer_get_app_state` `mapget_list_sources`

02 **Discover the semantics**
`mapget_query_schema` `mapget_docs`

03 **Inspect an assignment**
`mapget_extract_features`

04 **Validate attribute scope**
`mapget_validate_expression`

05 **Search and observe**
`viewer_start_search` `viewer_get_search`

06 **Style values and exceptions**
`viewer_set_search`

07 **Inspect a plotted case**
`viewer_get_search_results` `viewer_inspect`

Check the result against the original question.

The NDS.Live specification MCP supplies definitions. MapViewer supplies the actual map records.

Resolving references and matching traffic lights

Stored relation

Feature —Target ID→ **Referenced feature**

Resolve IDs in expressions

2026.5.0 planned: ERD-433

TRAFFIC_LIGHT without a stored relation

Road / Lane attribute
`TRAFFIC_LIGHT`
Direction + geometry validity

?
No stored link

Traffic-light landmark
Physical geometry
Object properties

Which landmark matches this attribute validity?

Spatial and semantic matching required. The nearest light may control a different lane.

Resolving a target ID does not establish an association that the data does not store.

Three entry points

WORKSHOP ROUTE

Remote MCP

`https://…/mcp`

Connect your agent.
Sign in through OAuth.
Select your viewer tab.

Native data tools also work without a tab.

CURRENT BROWSER DOCUMENT

Browser WebMCP

A compatible browser agent discovers the tools in the open viewer.

Browser setup and availability ↗

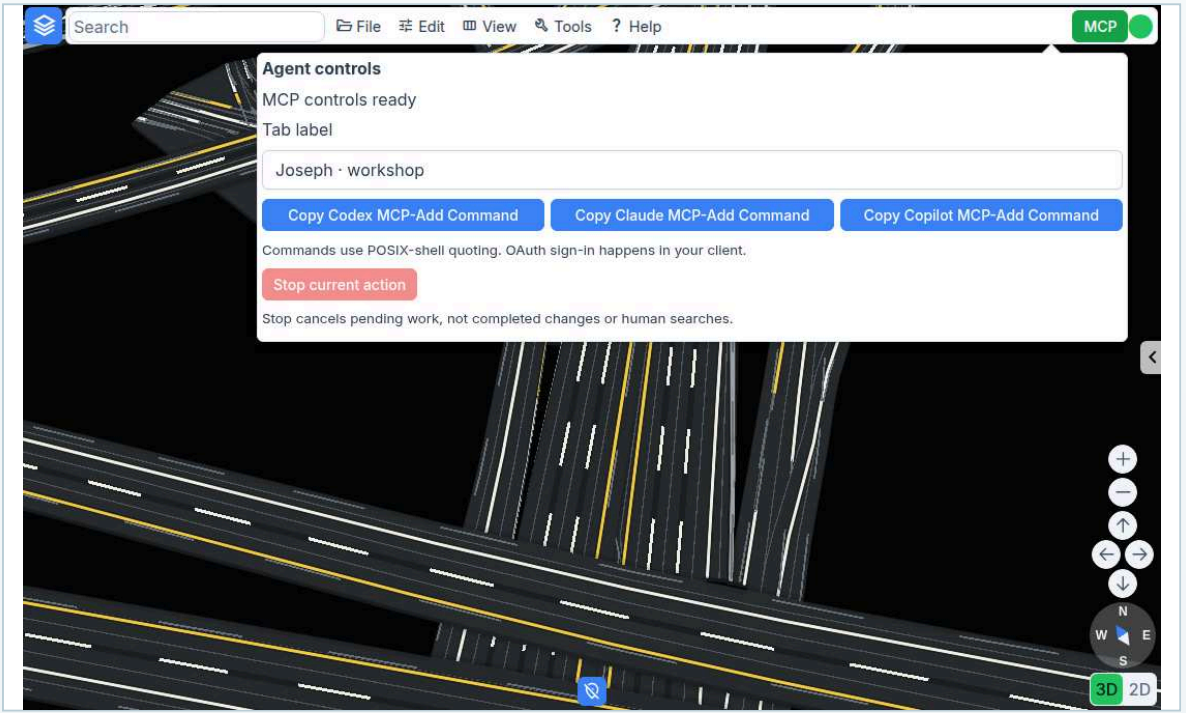
PLANNED

Embedded chat

The agent interface lives inside erdblick and reuses its action handlers.

NDS access and permission to send map data to a model provider are separate questions.

Connect your agent



1. Open viewer.nds.live and sign in.
2. Open MCP and give your tab a recognisable name.
3. Connect using the same NDS identity.

Codex CLI | Claude Code | Copilot CLI | VS Code | WebMCP

Codex CLI

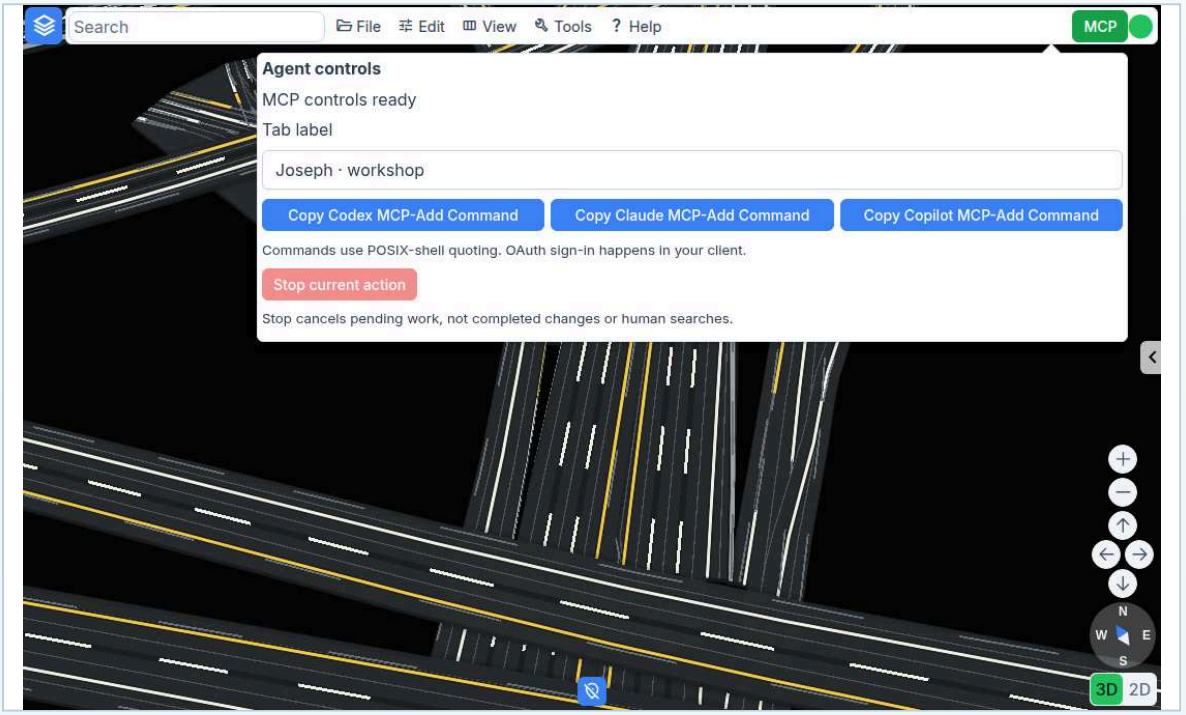
1. Use “Copy Codex MCP-Add Command” in the popup.
2. Run the copied add/login commands in your terminal.
3. Complete NDS sign-in, then verify the server with /mcp.

> List my viewer sessions. Select the tab with the name I give you and tell me which maps and layers are visible. Do not change anything.

The copied command contains the deployment’s endpoint, public client ID and requested scopes.

[Setup instructions and prompts ↗](#) Already connected? [Choose an experiment](#).

Connect your agent



1. Open viewer.nds.live and sign in.
2. Open MCP and give your tab a recognisable name.
3. Connect using the same NDS identity.

Codex CLI | **Claude Code** | Copilot CLI | VS Code | WebMCP

Claude Code

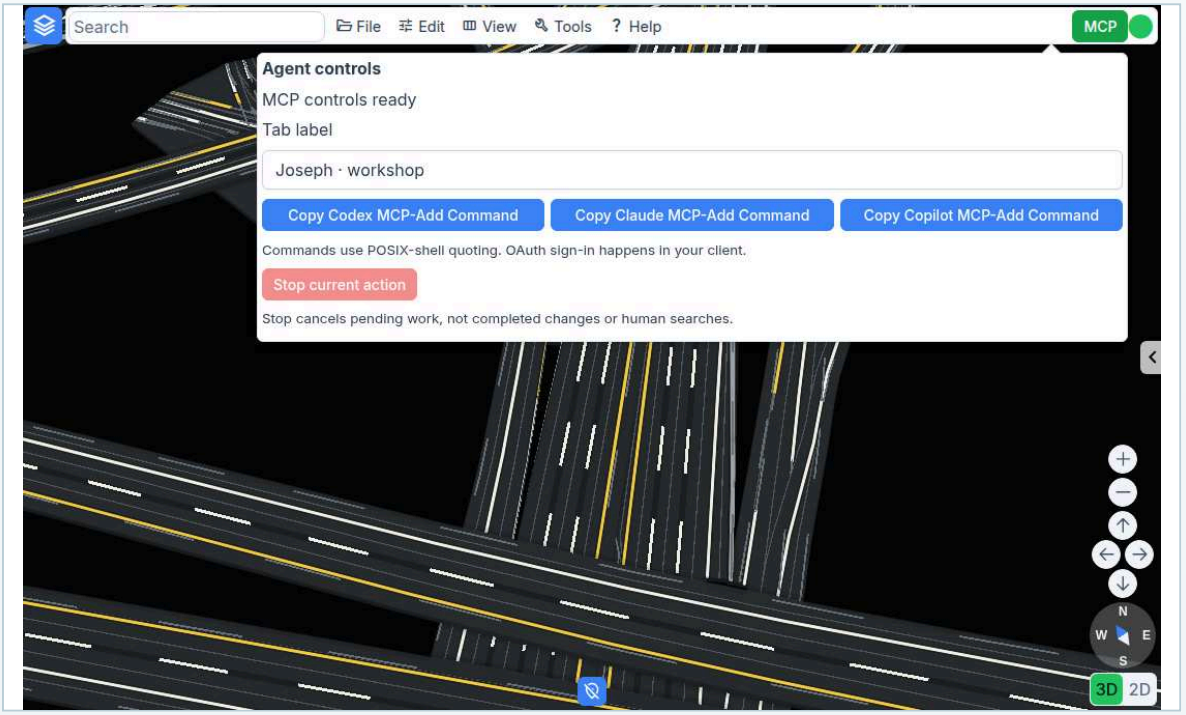
1. Use “Copy Claude MCP-Add Command” in the popup.
2. Run the copied commands and complete NDS sign-in.
3. Check /mcp in Claude Code before trying the prompt.

> List my viewer sessions. Select the tab with the name I give you and tell me which maps and layers are visible. Do not change anything.

Claude Code and a desktop/web connector can have different setup and callback requirements.

Setup instructions and prompts ↗ Already connected? Choose an experiment.

Connect your agent



1. Open viewer.nds.live and sign in.
2. Open MCP and give your tab a recognisable name.
3. Connect using the same NDS identity.

Codex CLI | Claude Code | Copilot CLI | VS Code | WebMCP

Copilot CLI

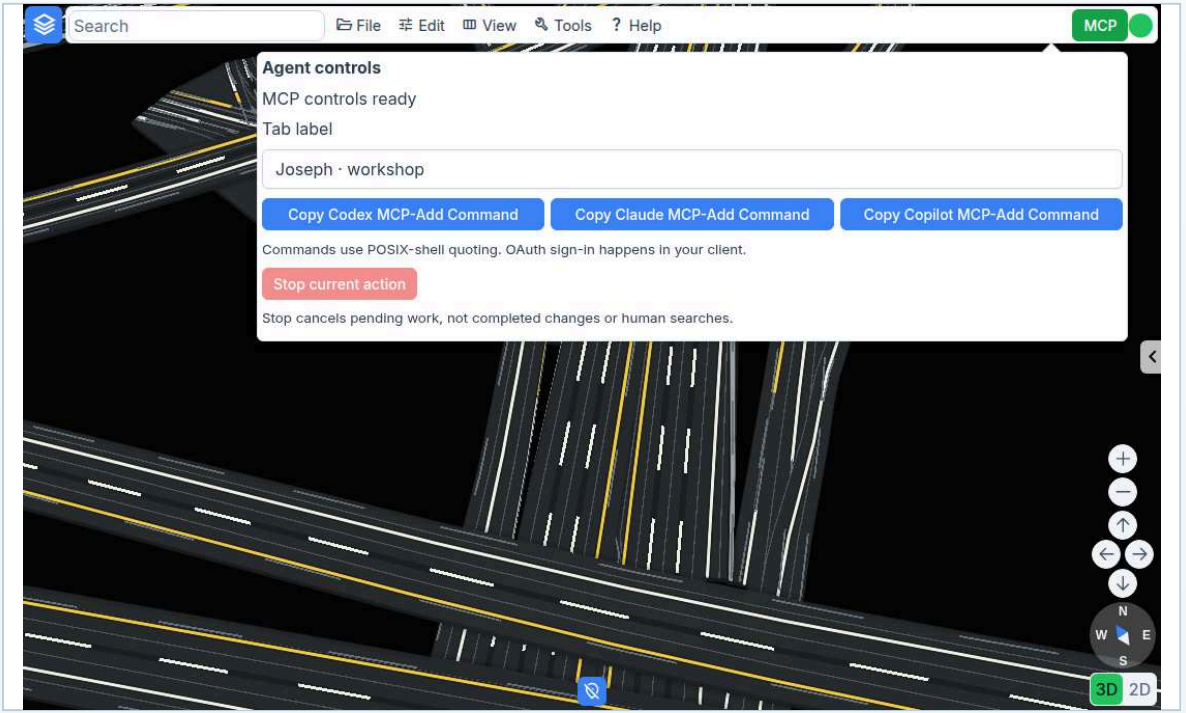
1. Use “Copy Copilot MCP-Add Command” in the popup.
2. Run it to start Copilot with this session’s MCP configuration.
3. Inside Copilot, run /mcp auth mapviewer and complete NDS sign-in.

> List my viewer sessions. Select the tab with the name I give you and tell me which maps and layers are visible. Do not change anything.

The hosted OAuth command is session-only. Copilot CLI and Copilot in VS Code have different setup.

Setup instructions and prompts ↗ Already connected? Choose an experiment.

Connect your agent



1. Open viewer.nds.live and sign in.
2. Open MCP and give your tab a recognisable name.
3. Connect using the same NDS identity.

Codex CLI | Claude Code | Copilot CLI | **VS Code** | WebMCP

VS Code

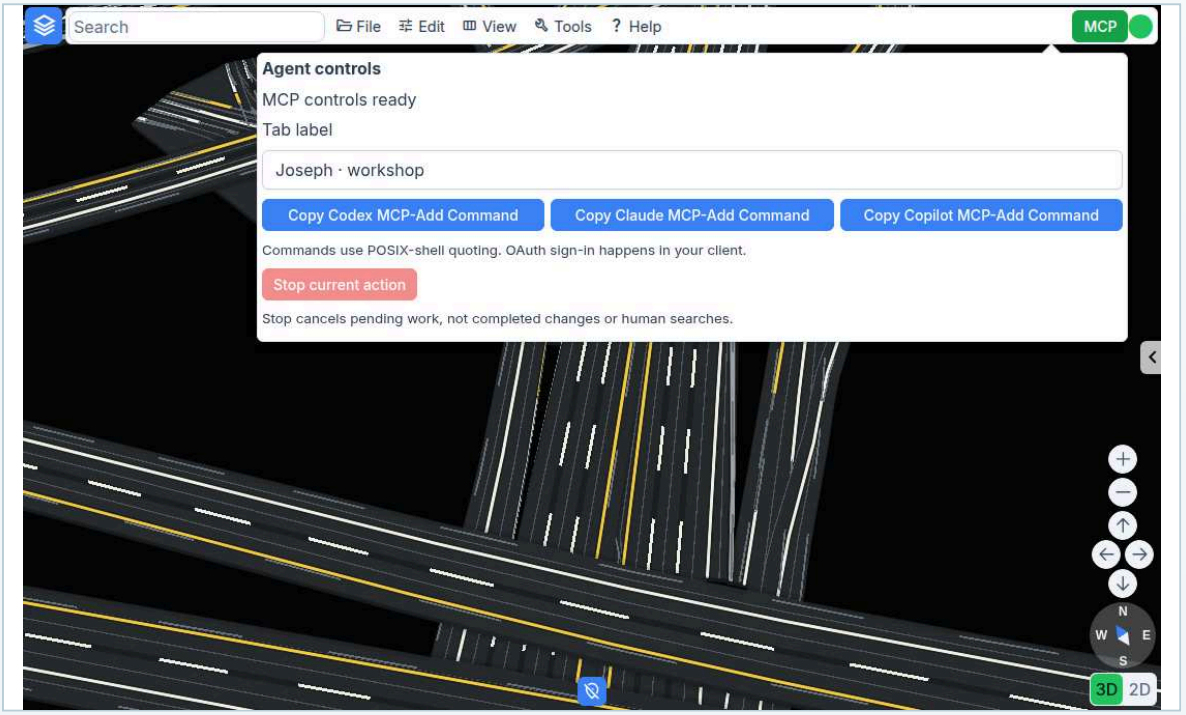
1. Run “MCP: Add Server”, choose HTTP and enter the MCP endpoint.
2. Use the deployment’s registered OAuth client ID, then sign in.
3. Trust the server, use Agent mode and check the available tools.

> List my viewer sessions. Select the tab with the name I give you and tell me which maps and layers are visible. Do not change anything.

Use the handout’s VS Code configuration, not the popup’s Copilot CLI command. The client callback must be registered.

Setup instructions and prompts ↗ Already connected? Choose an experiment.

Connect your agent



1. Open viewer.nds.live and sign in.
2. Open MCP and give your tab a recognisable name.
3. Connect using the same NDS identity.

Codex CLI | Claude Code | Copilot CLI | VS Code | WebMCP

WebMCP

1. Use a browser and browser-agent integration that support WebMCP.
2. Enable the WebMCP testing feature if your browser requires it.
3. Open the signed-in viewer and discover the tools in that tab.

> Tell me which maps and layers are visible in this tab. Do not change anything.

This is the optional browser route. Pasting a page URL into an arbitrary agent does not connect WebMCP.

Setup instructions and prompts ↗ Already connected? Choose an experiment.

What would you like to try?

Workshop · choose any question

VIEW AND STYLE

01 **Munich, Live and Classic**

02 **Functional road class**

03 **Animal warning signs**

06 **Road travel direction**

READ THE DATA

04 **Speed limits and applicability**

05 **Missing speed-limit attributes**

07 **The longest lane in view**

11 **Inline-skating candidates**

INVESTIGATE

08 **Why is the viewer slow?**

09 **Advertised extent and available tiles**

10 **What differs between these maps?**

12 **Redundant intersections?**

Connection help Tool atlas All prompts on your machine ↗

Munich, Live and Classic

01

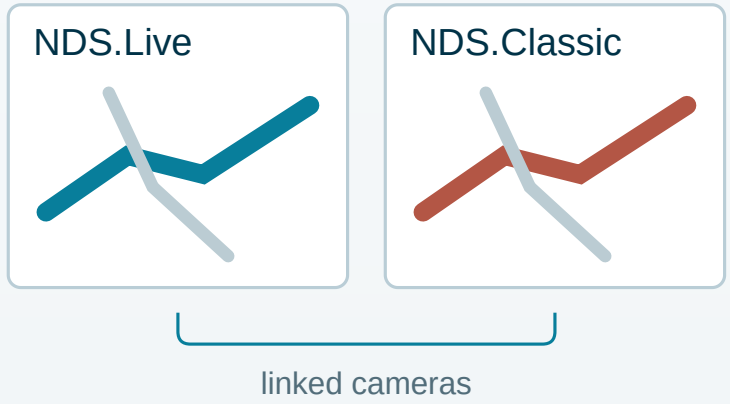
Munich. Live on the left, Classic on the right.

Same location. Independent layers. Linked cameras.

Copy prompt

▶ Full prompt

Schematic



A USEFUL RESULT

Two views show the same Munich area with the intended maps and synchronized movement.

▶ Data, tools and boundaries

All experiments Connection help Tool atlas Prompt sheet ↗

Functional road class

02

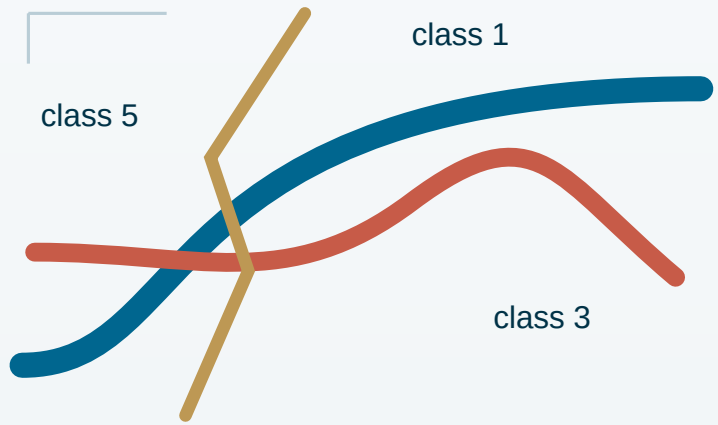
Colour the roads by functional road class.

Discover the field and categories in this map.

Copy prompt

▶ Full prompt

Schematic



Separate category colours

A USEFUL RESULT

Road classes have distinct colours, with an explanation of the categories and missing values.

▶ Data, tools and boundaries

Animal warning signs

03

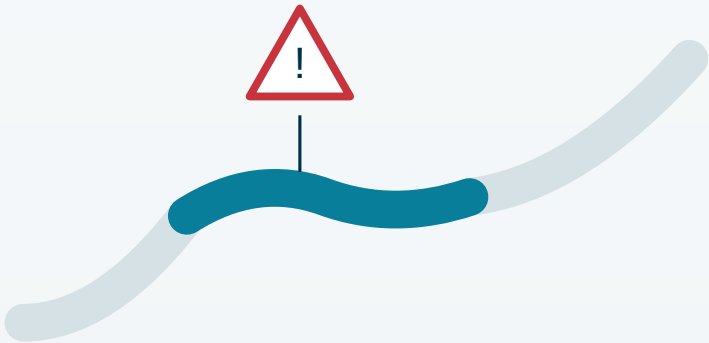
Where are the animal warning signs?

Search the visible area and inspect one assignment.

Copy prompt

▶ Full prompt

Schematic



Animal warning assignment

Highlight its spatial validity

A USEFUL RESULT

Matching warning-sign validities appear on the map, or a completed scoped search reports no matches.

▶ Data, tools and boundaries

Speed limits and applicability

Schematic

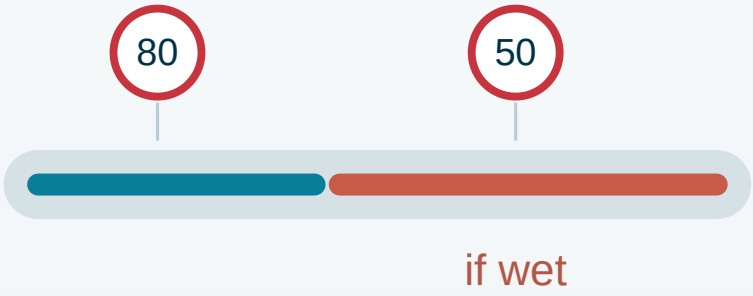
04

Which speed limit applies here?

Keep direction, spatial validity and conditions together.

Copy prompt

▶ Full prompt



Value + condition + validity

A USEFUL RESULT

The display distinguishes actual limit values and exposes the validity or conditions of an inspected example.

▶ Data, tools and boundaries

All experiments Connection help Tool atlas Prompt sheet ↗

Missing speed-limit attributes

Schematic

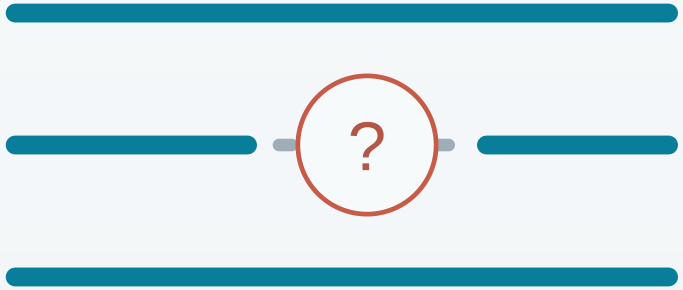
05

Which roads have no explicit speed limit?

Define what “missing” means before searching.

Copy prompt

▶ Full prompt



No explicit assignment

A USEFUL RESULT

A completed search identifies features without the defined explicit assignments and states its scope.

▶ Data, tools and boundaries

Road travel direction

06

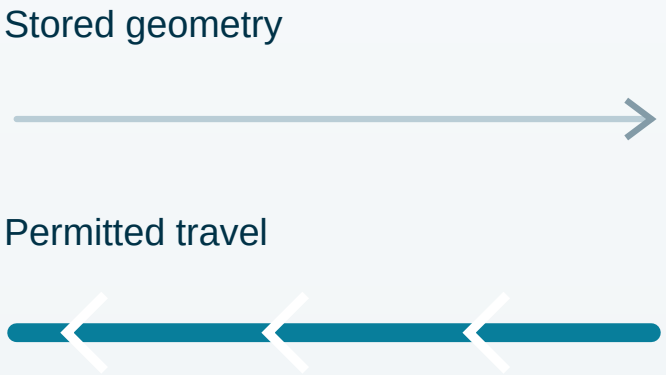
Show permitted travel direction.

Compare an arrow with the actual source record.

Copy prompt

▶ Full prompt

Schematic



A USEFUL RESULT

An arrow overlay explains permitted direction and agrees with an inspected data example.

▶ Data, tools and boundaries

All experiments Connection help Tool atlas Prompt sheet ↗

The longest lane in view

07

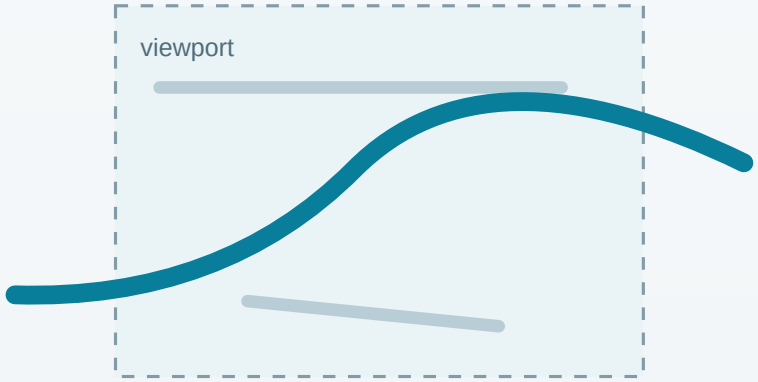
Find the longest lane in this viewport.

Use recorded length. Finish the bounded candidate scan.

Copy prompt

▶ Full prompt

Schematic



Full feature length may extend outside

A USEFUL RESULT

One lane is highlighted with its verified length, unit and the candidate coverage used for the comparison.

▶ Data, tools and boundaries

Why is the viewer slow?

08

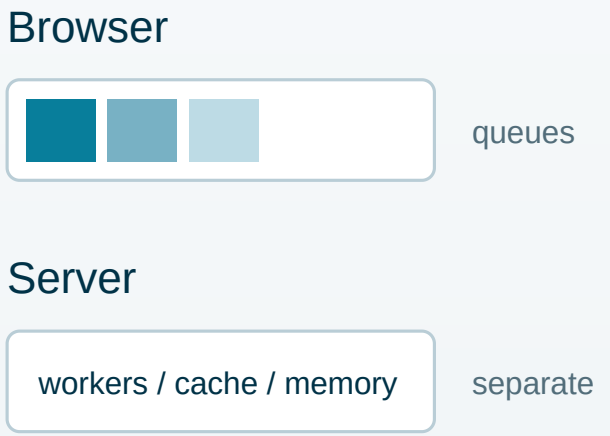
Why does my viewer feel slow?

Separate browser observations from server evidence.

Copy prompt

▶ Full prompt

Schematic



A USEFUL RESULT

A diagnosis separates observed browser and server evidence and proposes a testable hypothesis.

▶ Data, tools and boundaries

All experiments Connection help Tool atlas Prompt sheet ↗

Advertised extent and available tiles

09

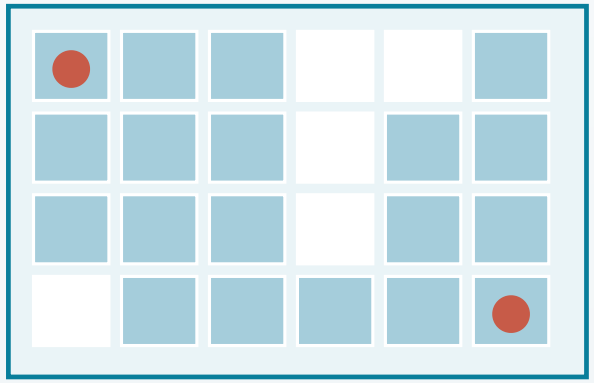
Does the tile coverage match the metadata?

Inspect sparse coverage. Probe at most eight tiles.

Copy prompt

▶ Full prompt

Schematic



Bounds, sparse coverage, sample probes

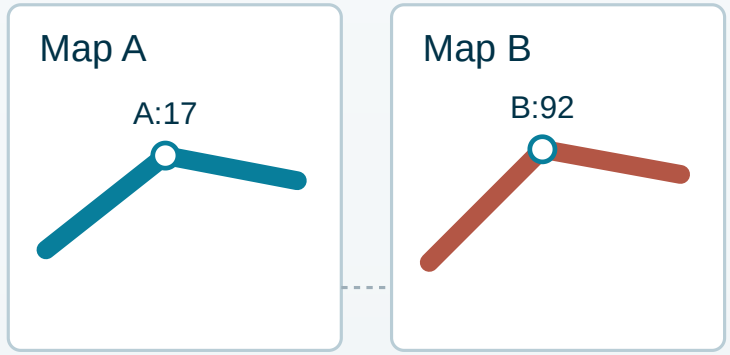
A USEFUL RESULT

A small evidence table compares the extent, coverage and outcomes of the selected tile probes.

▶ Data, tools and boundaries

What differs between these maps?

Schematic



Correspondence before comparison

10

What differs between these two maps?

Agree on correspondence. Compare up to ten examples.

Copy prompt

▶ Full prompt

A USEFUL RESULT

A bounded comparison names concrete differences and makes uncertain feature matches visible.

▶ Data, tools and boundaries

All experiments Connection help Tool atlas Prompt sheet ↗

Inline-skating candidates

11

Which roads might suit inline skating?

Agree on a rule. Leave missing evidence visible.

Copy prompt

▶ Full prompt

Schematic



A USEFUL RESULT

An agreed heuristic highlights candidates and distinguishes missing evidence from a negative match.

▶ Data, tools and boundaries

All experiments Connection help Tool atlas Prompt sheet ↗

Redundant intersections?

12

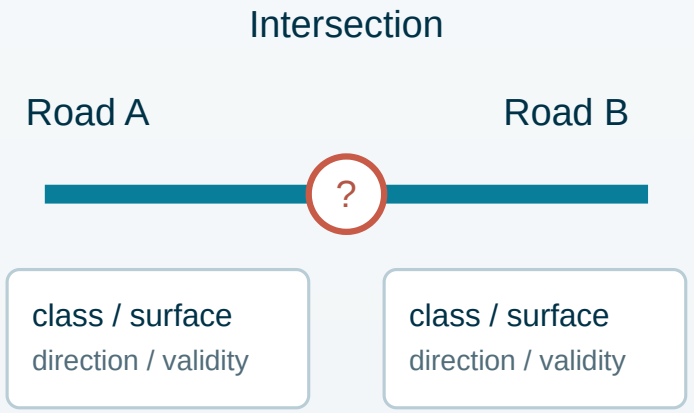
Two connected roads. Is this junction redundant?

Fetch the targets. Compare up to five candidates.

Copy prompt

▶ Full prompt

Schematic



A USEFUL RESULT

A few inspected candidates show both road targets and an explicit attribute comparison.

▶ Data, tools and boundaries

Discussion

Which question should we be able to answer next?

Show us a useful result.

Or an assumption the agent got wrong.

Back to the experiments Tool atlas Setup and prompts ↗