

TECHNICAL FIELD GUIDE

Contriver Node Atlas MCP

Configuration Reference

A cross-platform setup guide for Claude Desktop, Cursor, and VS Code.

MCP SETUP • ABSOLUTE PATHS • DIAGNOSTICS

Substitute all paths marked ← with your actual values.
Paths shown from the development machine are templates, not defaults.

Configuration Reference
Contriver Node Atlas

Two registration routes

Node Atlas registers its MCP server one of two ways. Pick one.

| Route | When to use | Python required |

A — Automatic: *Click AI in the Node Atlas toolbar.*

Default. Registers Claude Desktop, Cursor, and VS Code in one action.

No — a helper bundled with the add-in supplies its own

B — Manual: *Edit each client's config by hand.*

Non-standard interpreter, project-scoped config, or Route A unavailable

Yes — 3.10+ with `mcp[cli]`

The Prerequisites table below applies to Route B only.

Requirement	Check
Python 3.10+	<code>python3 --version</code>
<code>mcp[cli]</code> installed for that interpreter	<code>python3 -c "import mcp; print('ok')"</code>
<code>node_atlas_mcp.py</code> at a known absolute path	<code>ls -l /your/path/node_atlas_mcp.py</code>
Node Atlas palette run at least once	Produces <code>node_atlas_current.json</code>

Install `mcp[cli]` for a specific interpreter (use the absolute path):

BASH

```
# macOS / Linux
/Users/YourName/Desktop/ContriverNodeAtlas/CONTRIVER\ NODE\ ATLAS\ MCP\venv/bin/
python -m pip install "mcp[cli]"
# ↑ substitute your interpreter

# Windows (PowerShell)
C:\Users\YourName\envs\nodeatlas\Scripts\python.exe -m pip install "mcp[cli]"
# ↑ substitute your interpreter
```

The only runtime knob is `NODE\_ATLAS\_EXPORT` — the absolute path to `node_atlas_current.json`.

If the env var is absent, the server falls back to:

| `~/Desktop/ContriverNodeAtlas/ContriverAtlas/node_atlas_current.json`

Set it explicitly in every config block below to avoid the fallback silently pointing nowhere.

Route A — automatic registration

Open a design in Fusion, then click **AI** in the Node Atlas toolbar.

One entry named ``contriver-node-atlas`` is merged into each of three config files:

```
| Client | Scope written | Path |
| --- | --- | --- |
| Claude Desktop | global | macOS `~/Library/Application Support/Claude/
claude_desktop_config.json` · Windows `%APPDATA%\Claude\claude_desktop_config.json` |
| Cursor | global | macOS `~/.cursor/mcp.json` · Windows `%USERPROFILE%\cursor\mcp.json` |
| VS Code | user | macOS `~/Library/Application Support/Code/User/mcp.json` · Windows
`%APPDATA%\Code\User\mcp.json` |
```

Project-scoped files are never written. For ``.cursor/mcp.json`` or ``.vscode/mcp.json`` inside a project root, use Route B.

Merge Behaviour:

- Existing servers are preserved. Only the ``contriver-node-atlas`` key is added or replaced.
- A timestamped ``.bak`` copy is written before any existing config is modified.
- A config file that is not valid JSON is reported and left untouched.
- Missing files and missing parent directories are created.

A result panel reports one line per client — ``added``, ``updated``, ``skipped``, or ``failed`` — in the lower right of the palette. Click it to dismiss.

Restart each client after registering.

First launch.

The bundled helper fetches a Python 3.10+ runtime and ``mcp[cli]`` on its first run. This needs network access once, then it is cached.

On a slow connection the fetch can outlast the client's startup window and the server will report a failed start; see **Diagnosing a failed start**.

macOS.

The first launch is also subject to Gatekeeper.

See *macOS Gatekeeper — first launch of the bundled helper*

What the AI button writes

Reproduced for verification and manual repair. Substitute all paths marked ←.

Two independent paths appear in every block:

- `command` and `args` ← follow **wherever the add-in folder is installed**
- `NODE\_ATLAS\_EXPORT` ← always `~/Desktop/ContriverNodeAtlas/ContriverAtlas/node\_atlas\_current.json`, **regardless of where the add-in is installed**

The export path is fixed because the palette is the sole writer of the JSON and writes to that location unconditionally.

Do not repoint it to the add-in folder.

Claude Desktop and Cursor — macOS, Apple silicon

```
``json
{
  "mcpServers": {
    "contriver-node-atlas": {
      "command": "/Users/YourName/Desktop/ContriverNodeAtlas/mcp/bin/uv-macos-arm64",
      "args": [
        "run",
        "--script",
        "/Users/YourName/Desktop/ContriverNodeAtlas/mcp/node_atlas_mcp.py"
      ],
      "env": {
        "NODE_ATLAS_EXPORT": "/Users/YourName/Desktop/ContriverNodeAtlas/ContriverAtlas/
node_atlas_current.json"
      }
    }
  }
}
```

On Intel Macs the only difference is the helper filename: `uv-macos-x64`.

Claude Desktop and Cursor — Windows

```
``json
{
  "mcpServers": {
    "contriver-node-atlas": {
      "command": "C:\\Users\\YourName\\Desktop\\ContriverNodeAtlas\\mcp\\bin\\uv-windows-
x64.exe",
      "args": [
        "run",
        "--script",
        "C:\\Users\\YourName\\Desktop\\ContriverNodeAtlas\\mcp\\node_atlas_mcp.py"
      ],
      "env": {
        "NODE_ATLAS_EXPORT": "C:\\Users\\YourName\\Desktop\\ContriverNodeAtlas\\ContriverAtlas\\
node_atlas_current.json"
      }
    }
  }
}
```

VS Code — macOS

The top-level key is `servers`, and `"type": "stdio"` is required.

```

{
  "servers": {
    "contriver-node-atlas": {
      "type": "stdio",
      "command": "/Users/YourName/Desktop/ContriverNodeAtlas/mcp/bin/uv-macos-arm64",
      "args": [
        "run",
        "--script",
        "/Users/YourName/Desktop/ContriverNodeAtlas/mcp/node_atlas_mcp.py"
      ],
      "env": {
        "NODE_ATLAS_EXPORT": "/Users/YourName/Desktop/ContriverNodeAtlas/ContriverAtlas/
node_atlas_current.json"
      }
    }
  }
}

```

VS Code — Windows

```

{
  "servers": {
    "contriver-node-atlas": {
      "type": "stdio",
      "command": "C:\\Users\\YourName\\Desktop\\ContriverNodeAtlas\\mcp\\bin\\uv-windows-
x64.exe",
      "args": [
        "run",
        "--script",
        "C:\\Users\\YourName\\Desktop\\ContriverNodeAtlas\\mcp\\node_atlas_mcp.py"
      ],
      "env": {
        "NODE_ATLAS_EXPORT": "C:\\Users\\YourName\\Desktop\\ContriverNodeAtlas\\ContriverAtlas\\
node_atlas_current.json"
      }
    }
  }
}

```

Windows path note: use double backslashes `\\` or forward slashes `/` — both work in JSON. Do not use single `\'`.

Restart rules

Change made	Action required
Any config file edited	Restart the MCP client
<code>NODE_ATLAS_EXPORT</code> value changed	Restart the MCP client (env is read at spawn)
Fusion design changed	Re-run the Node Atlas palette (it is the sole writer)
Add-in source changed	Restart Fusion
Add-in folder moved or renamed	Re-click AI (registered paths are absolute)
Add-in folder replaced with fresh download	Re-click AI then re-clear Gatekeeper on macOS

1. Claude Desktop

macOS

Config file:

| ~/Library/Application Support/Claude/claude_desktop_config.json

JSON

```
{
  "mcpServers": {
    "contriver-node-atlas": {
      "command": "/Users/YourName/Desktop/ContriverNodeAtlas/CONTRIVER NODE ATLAS
MCP/venv/bin/python",
      "args": [
        "/Users/YourName/Desktop/ContriverNodeAtlas/CONTRIVER NODE ATLAS MCP/
node_atlas_mcp.py"
      ],
      "env": {
        "NODE_ATLAS_EXPORT": "/Users/YourName/Desktop/ContriverNodeAtlas/ContriverAtlas/
node_atlas_current.json"
      }
    }
  }
}
```

Substitute:

- `command` ← absolute path to your Python interpreter
- first `args` entry ← absolute path to `node_atlas_mcp.py`
- `NODE_ATLAS_EXPORT` ← absolute path to `node_atlas_current.json`

Restart Claude Desktop after saving.

Windows

Config file:

| %APPDATA%\Claude\claude_desktop_config.json

JSON

```
{
  "mcpServers": {
    "contriver-node-atlas": {
      "command": "C:\\Users\\YourName\\envs\\nodeatlas\\Scripts\\python.exe",
      "args": [
        "C:\\Users\\YourName\\ContriverNodeAtlas\\node_atlas_mcp.py"
      ],
      "env": {
        "NODE_ATLAS_EXPORT": "C:\\Users\\YourName\\Desktop\\ContriverNodeAtlas\\
ContriverAtlas\\node_atlas_current.json"
      }
    }
  }
}
```

Substitute:

- `command` ← absolute path to your Python interpreter (use `where.exe python` to find it)
- first `args` entry ← absolute path to `node_atlas_mcp.py`
- `NODE_ATLAS_EXPORT` ← absolute path to `node_atlas_current.json`

Windows path note: use double backslashes `\\` or forward slashes `/` — both work in JSON. Do not use single `\`.

Restart Claude Desktop after saving.

2. Cursor

Applies to all projects opened in Cursor on this machine.

macOS

Config file:

```
| ~/.cursor/mcp.json
```

JSON

```
{
  "mcpServers": {
    "contriver-node-atlas": {
      "command": "/Users/YourName/Desktop/ContriverNodeAtlas/CONTRIVER NODE ATLAS
MCP/venv/bin/python",
      "args": [
        "/Users/YourName/Desktop/ContriverNodeAtlas/CONTRIVER NODE ATLAS MCP/
node_atlas_mcp.py"
      ],
      "env": {
        "NODE_ATLAS_EXPORT": "/Users/YourName/Desktop/ContriverNodeAtlas/ContriverAtlas/
node_atlas_current.json"
      }
    }
  }
}
```

Substitute the same three values as Claude Desktop (macOS) above. Restart Cursor after saving.

Windows

Config file:

```
| %USERPROFILE%\.cursor\mcp.json
```

JSON

```
{
  "mcpServers": {
    "contriver-node-atlas": {
      "command": "C:\\Users\\YourName\\envs\\nodeatlas\\Scripts\\python.exe",
      "args": [
        "C:\\Users\\YourName\\ContriverNodeAtlas\\node_atlas_mcp.py"
      ],
      "env": {
        "NODE_ATLAS_EXPORT": "C:\\Users\\YourName\\Desktop\\ContriverNodeAtlas\\
ContriverAtlas\\node_atlas_current.json"
      }
    }
  }
}
```

Substitute the same three values as Claude Desktop (Windows) above. Restart Cursor after saving.

Cursor — Project-Local Config

Overrides the global config for one project. Checked in to the repo if desired; add to `.gitignore` if the paths are machine-specific.

macOS

Config file (create if absent):

```
| <project-root>/.cursor/mcp.json
```

JSON

```
{
  "mcpServers": {
    "contriver-node-atlas": {
      "command": "/Users/YourName/Desktop/ContriverNodeAtlas/CONTRIVER NODE ATLAS
MCP/venv/bin/python",
      "args": [
        "/Users/YourName/Desktop/ContriverNodeAtlas/CONTRIVER NODE ATLAS MCP/
node_atlas_mcp.py"
      ],
      "env": {
        "NODE_ATLAS_EXPORT": "/Users/YourName/Desktop/ContriverNodeAtlas/ContriverAtlas/
node_atlas_current.json"
      }
    }
  }
}
```

Windows

Config file (create if absent):

```
| <project-root>\.\cursor\mcp.json
```

JSON

```
{
  "mcpServers": {
    "contriver-node-atlas": {
      "command": "C:\\Users\\YourName\\envs\\nodeatlas\\Scripts\\python.exe",
      "args": [
        "C:\\Users\\YourName\\ContriverNodeAtlas\\node_atlas_mcp.py"
      ],
      "env": {
        "NODE_ATLAS_EXPORT": "C:\\Users\\YourName\\Desktop\\ContriverNodeAtlas\\
ContriverAtlas\\node_atlas_current.json"
      }
    }
  }
}
```

Restart Cursor after saving either file.

3. VS Code

VS Code MCP support requires **VS Code 1.99 or later**. The config format differs from Cursor and Claude Desktop: the top-level key is "servers" (not "mcpServers") and each server requires "type": "stdio" explicitly.

Workspace Config

Scoped to one project. Preferred for project-specific export paths.

Config file (create if absent):

```
| <project-root>/vscode/mcp.json
```

macOS:

JSON

```
{
  "servers": {
    "contriver-node-atlas": {
      "type": "stdio",
      "command": "/Users/YourName/Desktop/ContriverNodeAtlas/CONTRIVER NODE ATLAS
MCP/venv/bin/python",
      "args": [
        "/Users/YourName/Desktop/ContriverNodeAtlas/CONTRIVER NODE ATLAS MCP/
node_atlas_mcp.py"
      ],
      "env": {
        "NODE_ATLAS_EXPORT": "/Users/YourName/Desktop/ContriverNodeAtlas/ContriverAtlas/
node_atlas_current.json"
      }
    }
  }
}
```

Windows:

JSON

```
{
  "servers": {
    "contriver-node-atlas": {
      "type": "stdio",
      "command": "C:\\Users\\YourName\\envs\\nodeatlas\\Scripts\\python.exe",
      "args": [
        "C:\\Users\\YourName\\ContriverNodeAtlas\\node_atlas_mcp.py"
      ],
      "env": {
        "NODE_ATLAS_EXPORT": "C:\\Users\\YourName\\Desktop\\ContriverNodeAtlas\\
ContriverAtlas\\node_atlas_current.json"
      }
    }
  }
}
```

VS Code — User Config

Applies to all workspaces. Reached via **Command Palette** → **"MCP: Open User Configuration"**.

Config file locations:

Platform	Path
macOS	~/Library/Application Support/Code/User/mcp.json
Windows	%APPDATA%\Code\User\mcp.json

Contents are identical to the workspace configs above — same "servers" key, same three substitutions.

Reload: After saving either file, open the Command Palette → **"MCP: List Servers"** — `contriver-node-atlas` should appear with status **running**. If it shows **stopped**, open the MCP output panel (**View** → **Output** → **MCP**) for the startup log.

Quick-reference: config file locations

Agent	Scope	macOS	Windows
Claude Desktop	global	~/Library/Application Support/Claude/claude_desktop_config.json	%APPDATA%\Claude\claude_desktop_config.json
Cursor	global	~/.cursor/mcp.json	%USERPROFILE%\cursor\mcp.json
Cursor	project	.cursor/mcp.json in project root	same
VS Code	user	~/Library/Application Support/Code/User/mcp.json	%APPDATA%\Code\User\mcp.json
VS Code	workspace	.vscode/mcp.json in project root	same

The **AI** button writes the three global/user scopes only:

- **Claude Desktop Global**
- **Cursor Global**
- **VS Code User.**

The two project scopes — Cursor project and VS Code workspace — are Route B only.

Diagnosing a failed start (all agents)

Run the server bare — fastest real diagnostic:

BASH

```
# macOS / Linux
/your/python /your/node_atlas_mcp.py

# Windows (PowerShell)
C:\your\python.exe C:\your\node_atlas_mcp.py
```

Instant traceback → that traceback is your answer

- **Sits silently** → server started fine; the problem is in the client config
- **`ModuleNotFoundError: No module named 'mcp'`** → wrong interpreter; this Python does not have `mcp[cli]`
- **`Node Atlas export not found`** → `NODE_ATLAS_EXPORT` points nowhere, or the palette has not been run yet

Ctrl+C to exit.

> ****Route A (bundled helper).**** Run the helper instead of a system interpreter.

> Substitute both paths marked ←.

```
` ` `bash
# macOS / Linux
~/Desktop/ContriverNodeAtlas/mcp/bin/uv-macos-arm64 \
run --script ~/Desktop/ContriverNodeAtlas/mcp/node_atlas_mcp.py
```

Windows (PowerShell)

Route A (bundled helper)

Run the helper instead of a system interpreter. Substitute both paths marked ←.

```
# macOS / Linux
~/Desktop/ContriverNodeAtlas/mcp/bin/uv-macos-arm64 \
run --script ~/Desktop/ContriverNodeAtlas/mcp/node_atlas_mcp.py

# Windows (PowerShell)
C:\Users\YourName\Desktop\ContriverNodeAtlas\mcp\bin\uv-windows-x64.exe `
run --script C:\Users\YourName\Desktop\ContriverNodeAtlas\mcp\node_atlas_mcp.py
```

The first run of this command completes the Python and `mcp[cli]` fetch in the foreground with no time limit.

It is also the fastest way to confirm the helper is executable at all.

macOS Gatekeeper — first launch of the bundled helper

Applies to macOS only, and to Route A only. Skip on Windows and under Route B.

Symptom

The **AI** button reports success and writes the config, but the client shows the server as failed:

- Claude Desktop — Server disconnected, Could not attach to MCP server contriver-node-atlas
- Cursor — the server never reaches ready
- VS Code — MCP: List Servers shows stopped

Restarting the client does not clear it.

Cause

Route A launches a uv helper binary bundled with the add-in. macOS tags every file extracted from a downloaded archive with a quarantine flag, and the helper is not notarized under an Apple Developer ID. Gatekeeper blocks it. Because the client launches the helper in the background, no approval dialog is ever shown — the process is terminated silently and the client reports only that the connection dropped.

This is a one-time approval per install. Replacing the add-in folder with a freshly downloaded copy re-applies the flag.

Fix A — System Settings, no terminal

1. Quit and reopen the MCP client. This triggers a fresh block.
2. Open **System Settings** → **Privacy & Security** and scroll to **Security**.
3. Find the notice naming uv-macos-arm64 (Apple silicon) or uv-macos-x64 (Intel) and click **Open Anyway**. Authenticate when prompted.
4. Quit and reopen the client.

The notice appears only for a short period after the block. If it is absent, repeat step 1 and check again immediately.

Fix B — Terminal, definitive

Substitute the add-in folder path marked ←. The full path to the helper is shown in the client's own server entry, in the **Command** field.

```
bash
```

```
xattr -dr com.apple.quarantine ~/Desktop/ContriverNodeAtlas # ← your add-in folder
```

Then quit and reopen the client.

Verifying

Run the helper directly using the Route A command under *Diagnosing a failed start*.

- Sits silently → the helper runs and Gatekeeper is cleared. Ctrl+C to exit.
- zsh: killed, or an immediate exit with no output → still blocked; apply Fix B.