



Agentic AI Foundation

AGNTCon + MCPCon North America

Where the **agentic stack** is being built.

PyPI, The Registry, and The Rough Edges

Shipping an MCP Server Nobody Told You How To

Mesut Oezdil

DevOps Engineer, CNCF Tech Lead

Most guides stop at “here is a server”...

1 > **package**

pyproject,
PyPI,
uvx

2 > **prove it's yours**

namespace,
mcp-name,
Registry

3 > **get found**

aggregators,
clients,
configs

4 > **keep up**

SDK v2,
spec 2026-07-28

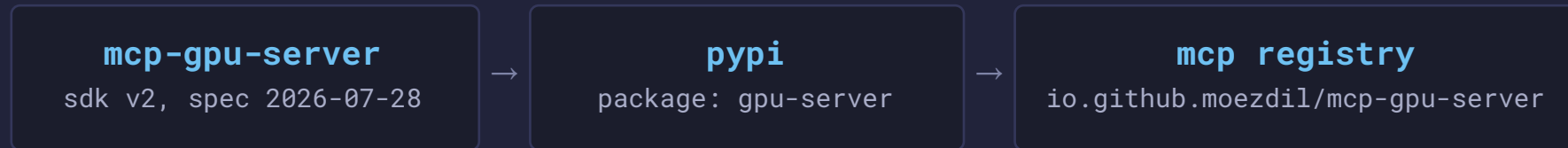
This is the second half

Where the **agentic stack** is being built.



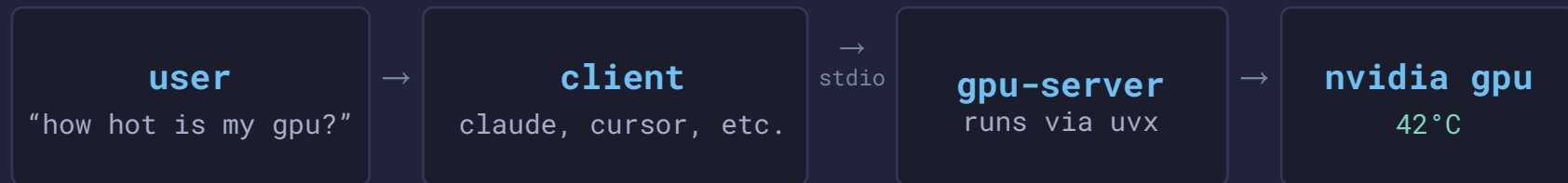
The big picture

publish: once



↓

use: every question



The gpu is just the vehicle. Everything here applies to any mcp server.

Meet mcp-gpu-server

```
from mcp.server import MCPServer

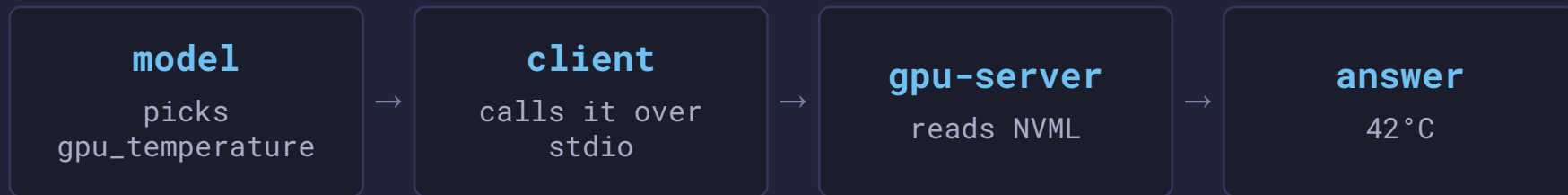
mcp = MCPServer("mcp-gpu-server")

@mcp.tool(annotations=READ_ONLY)
def gpu_utilization() -> dict:
    """Busy % of the GPU cores (gpu_pct) and memory
    controller
    (mem_pct) for each device.

    mem_pct is memory bandwidth activity, not how full
    VRAM is; use gpu_vram for that.
    """
    s = _all_stats() # NVML, else nvidia-smi
    ...
```

```
# 5 tools, all read-only
gpu_info
    name, driver, cuda
gpu_utilization
    core and memory-controller busy
    %
gpu_vram
    total, used, free
gpu_temperature
    °C
gpu_stats
    everything at once
```

One question, end to end



```
# stdin (client → server)
{"method":"tools/call","params":{"name":"gpu_temperature"}}
# stdout (server → client)
{"result":{"structuredContent":{"gpus":[{"index":0,"temp_c":42}]}}
```

SDK v2: 1 server, 2 protocols

	sdk v1	sdk v2 since jul 2026
server class	FastMCP the old entry point	MCPServer renamed, one import line changes
protocol	2025-11-25, stateful handshake, session	2026-07-28, stateless every request carries its own context
old clients	n/a	still supported falls back to 2025-11-25
discovery	initialize handshake capabilities only from the handshake	server/discover handled by the sdk
caching	none no freshness hint	ttlMs, cacheScope "keep this tool list for 24h"
pip install mcp	v1 code fails No module named 'mcp.server.fastmcp'	works pin mcp>=2.3,<3

Package it

```
# README.md, line 1
mcp-name:
io.github.moezdil/mcp-gpu-server

# pyproject.toml
name = "gpu-server"
dependencies = ["mcp>=2.3,<3", ...]
[project.scripts]
gpu-server = "mcp_gpu_server.server:main"
```

```
$ uvx gpu-server
```

```
name
  what users install
<3
  the next major can't break you
gpu-server script
  so uvx gpu-server just works
mcp-name
  the Registry checks this line
```

PyPI never forgets

*every
fix
costs
a version*

uploads can't be replaced:
delete one, and its name is gone
for good.
check before you upload.

```
$ uv build
$ tar -tzf dist/*.tar.gz
  # anything in the sdist you
  # didn't mean to ship?
$ twine check dist/*
$ mcp-publisher validate
$ uv publish
$ mcp-publisher publish
```

The Registry: a phone book, not a store

prove the namespace

```
io.github.moezdil/...  
= your GitHub login
```

prove the package

```
mcp-name: <server name>  
= first line of the README  
on PyPI
```

```
$ mcp-publisher login github  
✓ Successfully logged in
```

```
$ mcp-publisher publish  
✓ Successfully published  
✓ Server io.github.moezdil/  
mcp-gpu-server version 0.2.2
```

The Registry checks **who**, not what

checks

- ✓ package is on pypi.org
- ✓ that exact version exists
- ✓ README has the mcp-name line
- ✓ namespace matches your login
- ✓ description ≤ 100 chars

doesn't

- ✗ run your server
- ✗ read your tool descriptions
- ✗ scan your code
- ✗ look inside your repo
- ✗ remove abandoned servers

A listing is a pointer, not an attestation

Discovery: where it still breaks



Agentic AI Foundation

AGNTCon + MCPCon

North America

mcp registry

the source



aggregators

PulseMCP, Glama, GitHub



clients

where users are

VSCode

reads GitHub MCP
Registry, first listing
approved by a human

Claude Desktop

its own curated
directory

Cursor & others

you paste mcp.json by
hand

→ ship a copy-paste client config in your README

Where the **agentic stack** is being built.

The rough edges

1 > package

- ✗ floor without ceiling
- `pin mcp>=2.3,<3`
- ✗ package name ≠ command
- same-name script
- ✗ extra files in the sdist
- limit it, `tar -tzf`

2 > pypi

- ✗ README fixed after upload?
- no: check mcp-name first
- ✗ account hygiene
- recovery codes, 2nd owner

3 < mcp registry

- ✗ doesn't check your code
- a pointer, not proof
- ✗ username change
- reserve the old name

4 > discovery

- ✗ registry ≠ VSCode
- copy-paste config

None of these are in the getting-started guide.
Each is one command or one setting away from not hurting you.

Proof: real hardware

37/37

gpu checks passed

2026-07-28

negotiated by a v2 client

2025-11-25

negotiated by a v1 client

```
$ python e2e.py venv/bin/gpu-server
```

```
protocol: 2026-07-28
```

```
gpu_stats  NVIDIA GeForce RTX 3090  24576 MiB  42°C  CUDA 13.0
```

Before you publish

- [x] pin the major: `mcp>=2.3,<3`
- [x] `mcp-name`: on the first line of the README
- [x] a console script named like the package
- [x] `tar -tzf` the sdist before uploading
- [x] `twine check + mcp-publisher validate`
- [x] one run on real hardware
- [x] one v1 client, one v2 client
- [x] 2FA recovery codes saved, a second owner



Agentic AI Foundation

AGNTCon + MCPCon North America

Thanks!

Let's connect!

// linkedin.com/in/mesut-oezdil

// github.com/moezdil

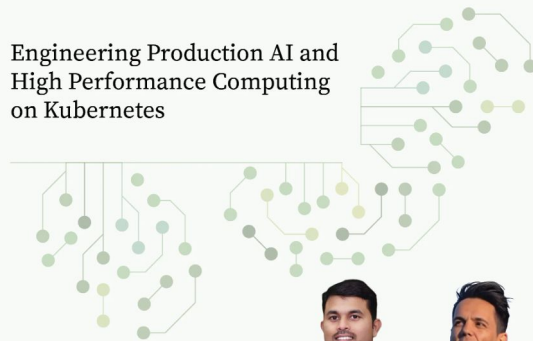
// publishing in Feb 2027 →

<packt>

FIRST EDITION

From Pods to *Petaflops*

Engineering Production AI and
High Performance Computing
on Kubernetes



Sanjeev Ganjihal | Mesut Oezdil