

```
> axis-cli --version 1.0.0
```

Full control of Axis IP cameras from your terminal.

VAPIX, CamOverlay, CamStreamer, CamSwitcher, and fleet health.

Scriptable automation replaces isolated point-and-click workflows.

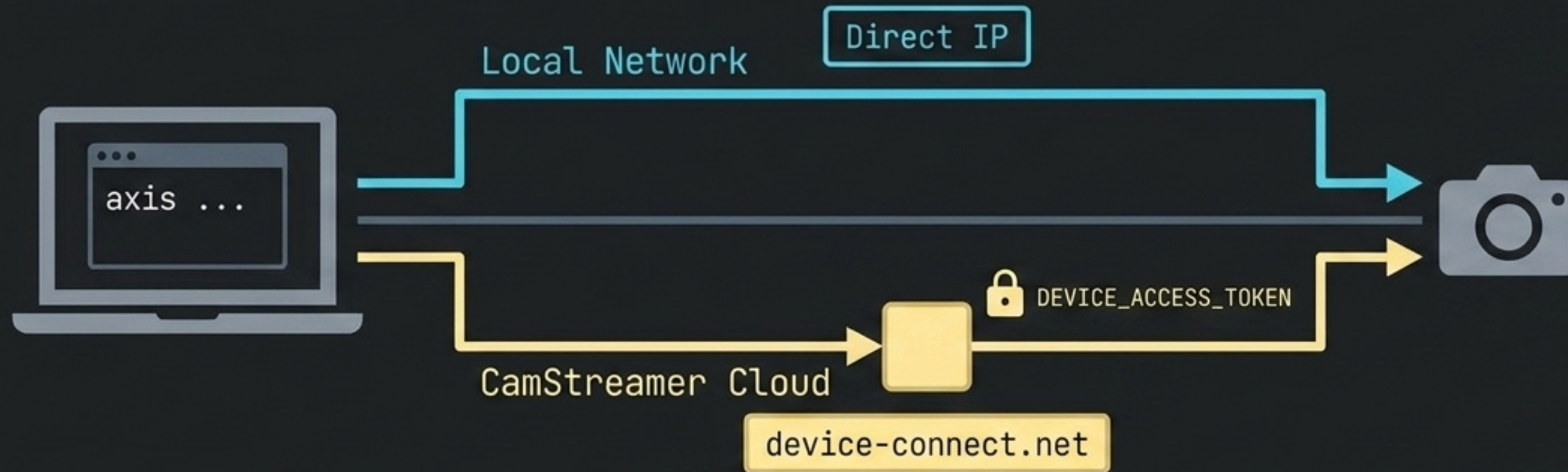
Dimension	Web UI / Manual Curl	axis-cli
Workflow Paradigm	⚠ Point-and-click, isolated, manual	✅ Scriptable, aliased, cron-ready
Scale of Operation	One camera at a time	Entire fleets via batch commands and health checks
Authentication	Re-entered per session or hardcoded	Abstracted into secure named profiles
Ecosystem Reach	Fragmented interfaces	Unified stack: VAPIX, PTZ, CamStreamer
Extensibility	Closed system	Open architecture

Manage profiles, not IP addresses.



Save credentials exactly once per camera.
From then on, execute all commands against a semantic name,
eliminating hardcoded IPs from your automation scripts.

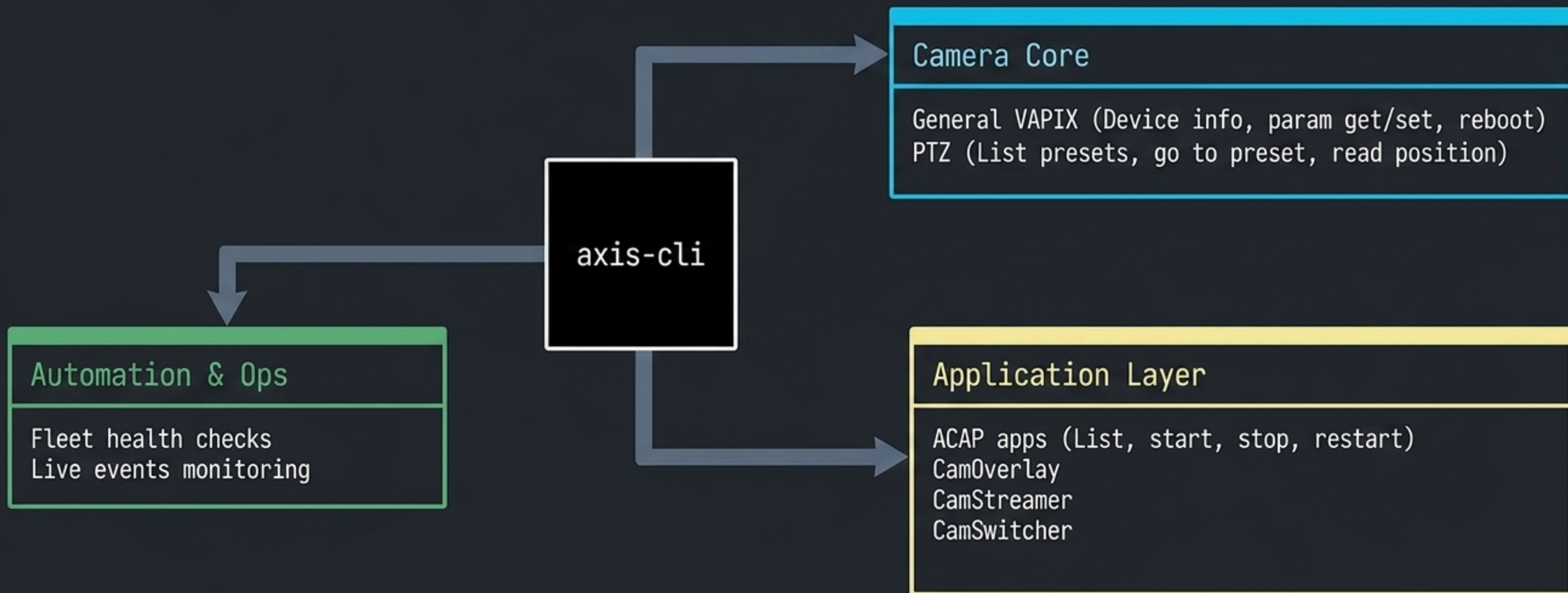
Unified syntax across local networks and cloud infrastructure.



No context-switching required. Whether querying a camera on a local switch or bouncing through a cloud proxy, the CLI abstracts the connection layer entirely.

The universal adapter for the Axis ecosystem.

One tool commands the entire stack. Bypass disparate UIs and custom scripts with a unified command-line interface spanning hardware control, application layers, and fleet operations.



Direct manipulation of camera hardware and VAPIX parameters.

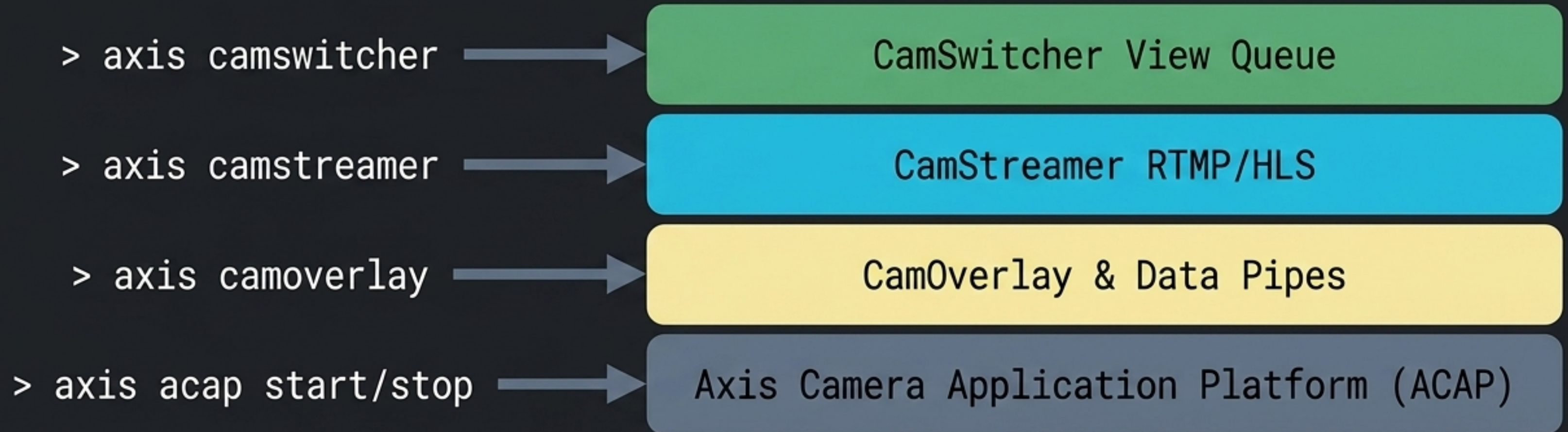
- Execute general VAPIX tasks instantly.
- Query device info, manipulate configuration parameters, or trigger reboots.
- Drive PTZ mechanics by listing saved presets, driving to coordinates, or polling current positions for automation triggers.

```
> axis ptz list Warehouse_PTZ
```

Name	Position
Door	1
LoadingDock	2

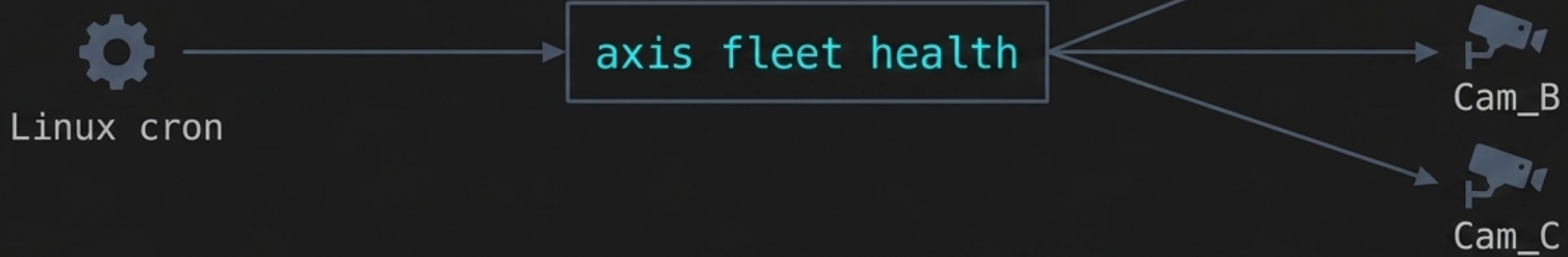
```
> axis ptz goto Warehouse_PTZ --preset  
LoadingDock  
[ OK ] Camera successfully moved to  
preset: LoadingDock
```

Programmatic control over ACAPs and stream management.



Built for developers extending camera functionality.
Wire your backend logic directly to CamOverlay text fields
or trigger CamSwitcher views via terminal commands.

Execute fleet-wide health checks with a single cron job.



Profile	Reachability	Firmware	SD Card	ACAP Count
Cam_A	UP (Green)	10.12.1	Mounted	3/3 Running
Cam_B	UP (Green)	10.12.1	Mounted	2/2 Running
Cam_C	DOWN (Red)	Unknown	Unknown	Unknown

Designed for scheduled monitoring. Evaluate the operational status of dozens of cameras simultaneously without clicking through a single web interface.

Monitor VAPIX events in real time.

```
> axis events watch Warehouse_PTZ
```

```
[14:02:01] [MOTION] Region 1: Active
[14:02:05] [MOTION] Region 1: Inactive
[14:02:01] [MOTION] Region 1: Inactive
[14:15:22] [I/O] Pin 2 state changed to HIGH
[14:15:22] [I/O] Pin 2 state changed to HIGH
[14:15:25] [ACAP] CamOverlay triggered: Custom
Text Push
[14:15:25]
```



Motion

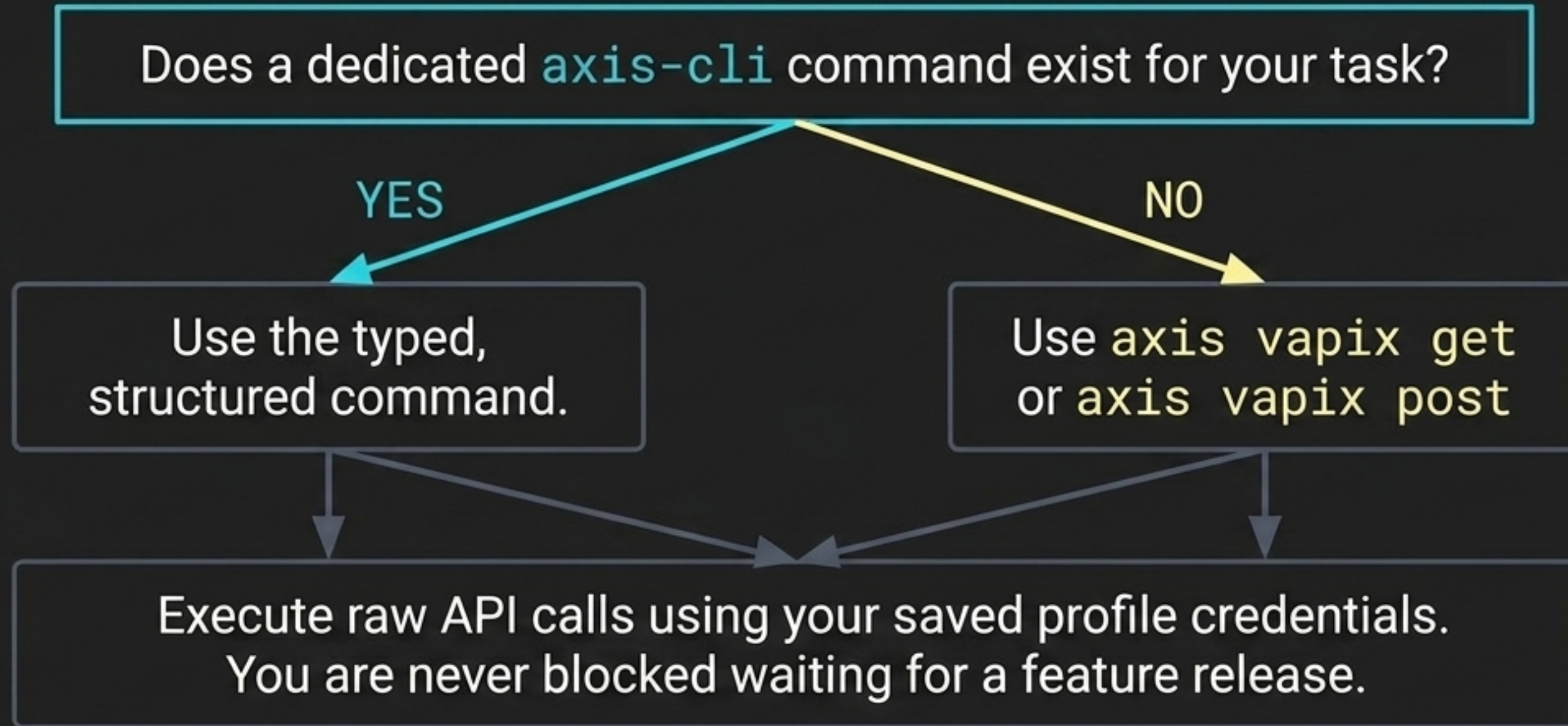


I/O states

Instantly log Motion detection, I/O pin states, and custom ACAP events directly to stdout.

Network Requirement: Live event watching mandates a direct local profile connection (`--ip`). Direct WebSocket streaming is not supported over cloud proxy routing.

The raw VAPIX escape hatch.

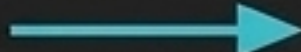
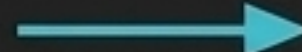


The CLI provides structured wrappers for common workflows, but exposes the raw VAPIX interface for edge cases and custom scripting against any Axis endpoint.

Built on a non-reinvented, extensible foundation.

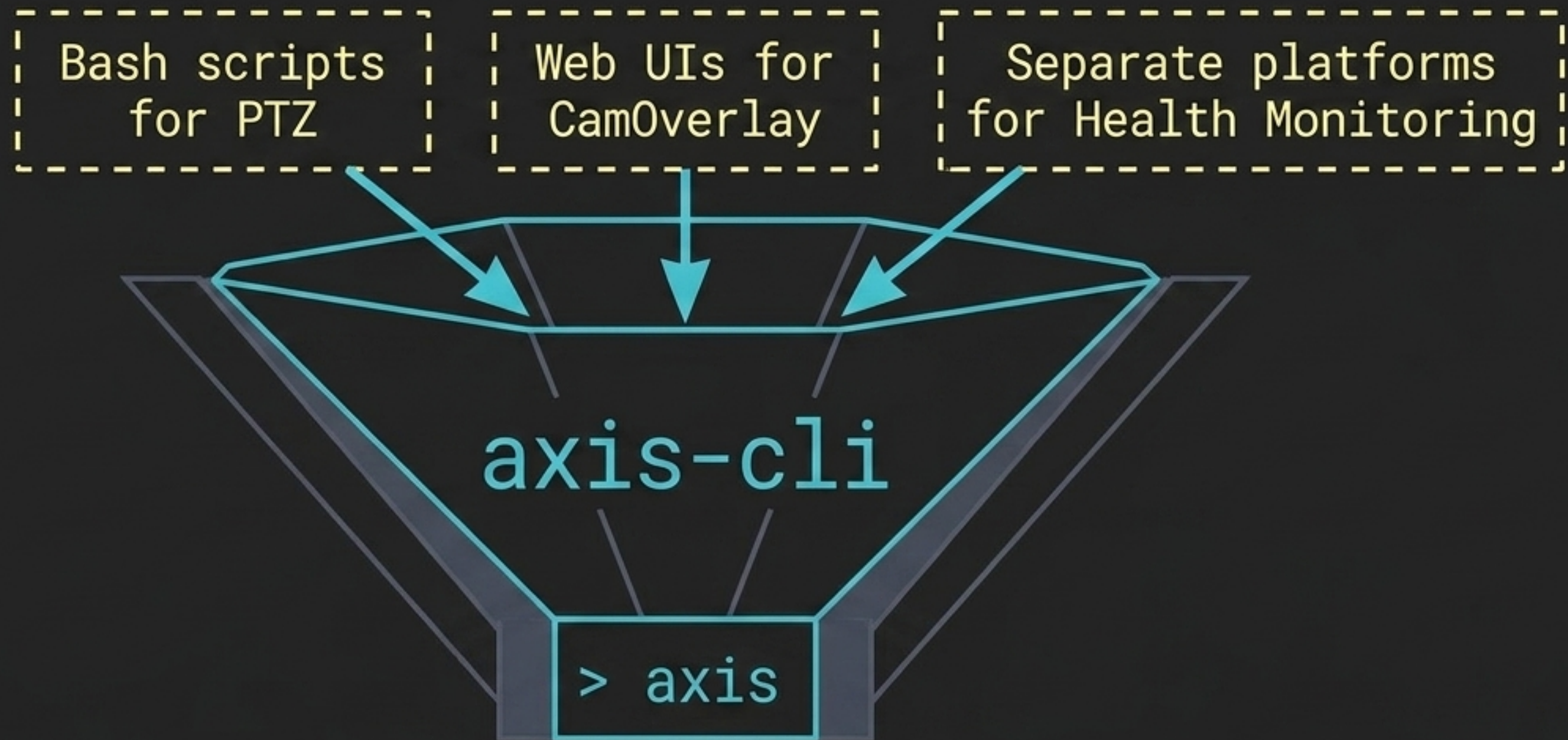
	Environment & Dependencies Matrix
Core Engine	Node >=18
Domain Libraries	camstreamerlib ^4.0.27
CLI Framework	commander ^14.0.0, cli-table3 ^0.6.5, chalk ^4.1.2
License	ISC Open Source

Extensibility Blueprint

Create file in `src/commands/`  Export `registerCommands(program)`  Wire into `src/index.ts`

Leveraging deep domain expertise in Axis integrations. Adding new functionality is frictionless.

One command. The entire stack.



axis-cli acts as the universal adapter. It transforms fragmented camera management into a unified, version-controllable, and automatable engineering workflow.

Initialize your fleet.

```
# Install globally via npm
```

```
npm install -g axis-cli
```

```
# Add your first camera profile
```

```
axis profile add --name FrontGate --ip 192.168.1.50 --user root
```

```
# Verify device status
```

```
axis vapix info FrontGate
```

Available now via npm. Documentation and source on GitHub.