

Ten Marks, One Hand: The StarDevs Diagram Kit

The glyphs we draw our systems with, the rules for putting them together, and the files to draw your own.

Mykola Lobanov, Founder & Chief Product and Technology Officer
October 22, 2026 · 2151 words

Why one hand

Every StarDevs service has a mark, and each mark is a small drawing of what the service does. The gateway fans one call out to several models. The sandbox is a wall with one way out. The migration moves a payload and draws its way back beside it. Drawn together, the ten marks turned out to share a vocabulary: they are sentences in one language.

That language is useful beyond the marks. The diagrams in our proposals, on the service pages and in these articles describe the same systems the marks do: agents, gateways, boundaries, clusters and the way back. When every diagram invents its own pictures, a reader relearns the notation each time, and two drawings of one system look like two systems.

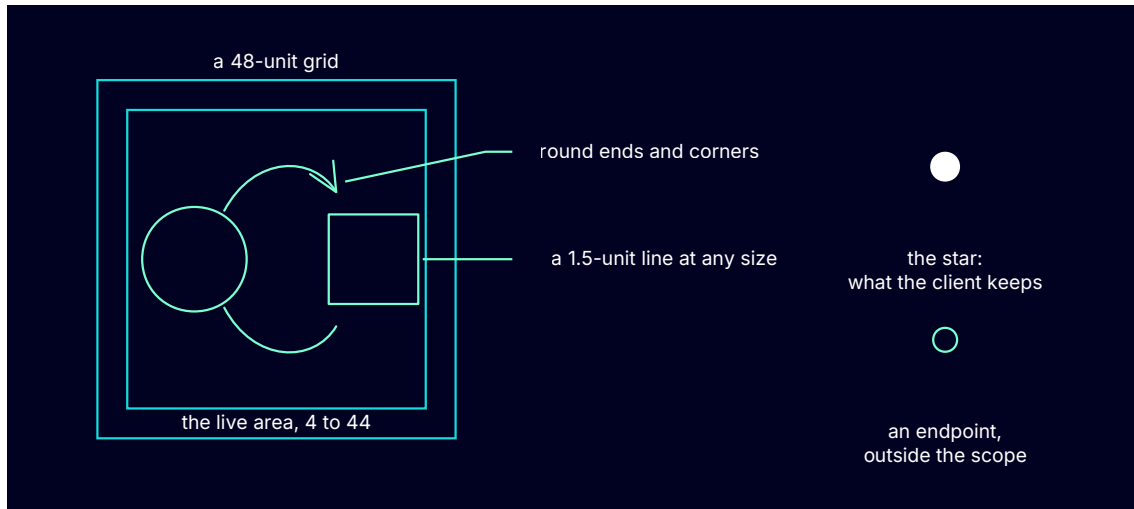
This kit is the vocabulary, written down: 54 glyphs, the construction they share, the rules for composing them and the files to use them. The two diagrams near the end are built from nothing else.

The construction

Every glyph is drawn on a 48-unit square and kept inside a live area from 4 to 44, so no two glyphs disagree about how much room they take.

- **Line only.** A glyph is one line, 1.5 units wide at whatever size it is placed. Scaled to 96 units, it has the same line as the diagram around it, not a line twice as heavy.
- **Round ends and corners,** the terminals of the StarDevs logo, whose rays meet at one white point.

- **One solid.** The only filled shape in the kit is the star, and a diagram has exactly one. It marks what the client keeps when the work is done.
- **An open dot is an endpoint:** something outside the drawing's scope, such as a model you call but do not run.



The agent glyph on its grid. Every glyph keeps to the live area, and its line stays 1.5 units at any size.

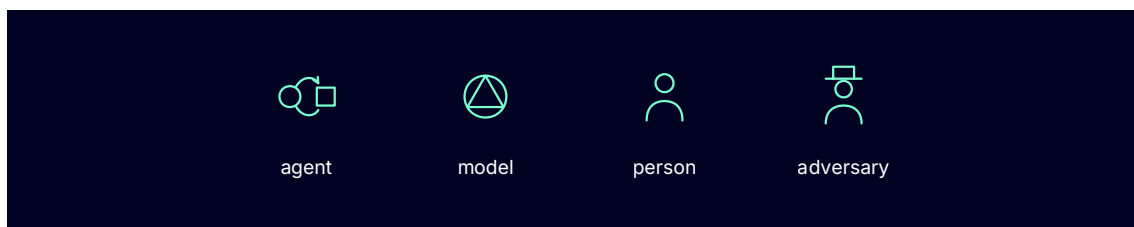
Colour follows the ground. On the dark grounds of this site the line is mint and the star white. On white paper the line is ink and the star indigo, because mint on white measures 1.2:1 and would not be seen. Each colour is a token in the site's design system, and a glyph never takes one from anywhere else.

The vocabulary

54 glyphs in six families. Things are placed as glyphs. Most flows are drawn as lines, and the flow glyphs are their legend keys.

Actors

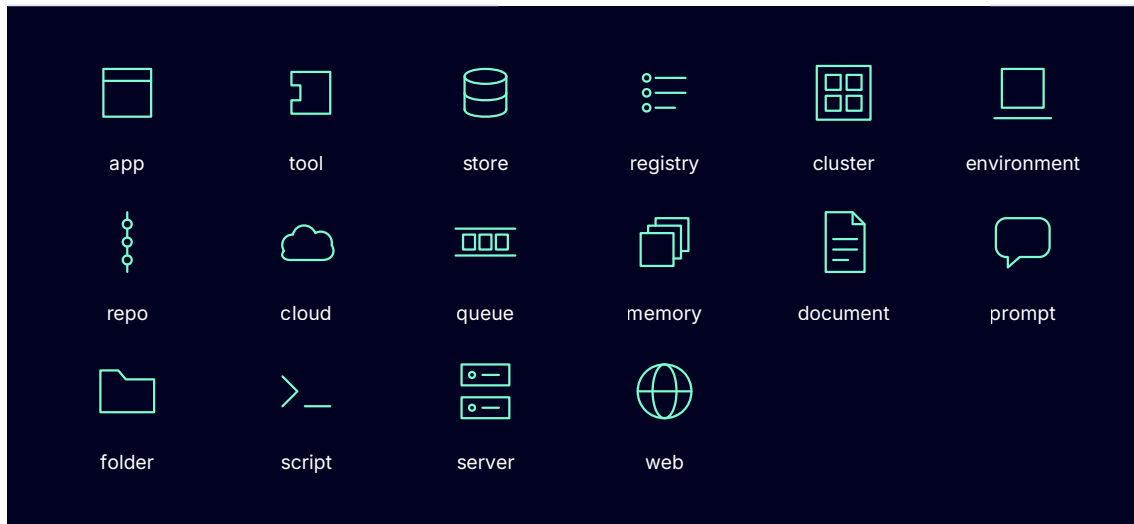
Who acts: an agent, the model it calls, the people around both, and whoever would turn the system against them.



The actors: agent, model, person, adversary.

Systems

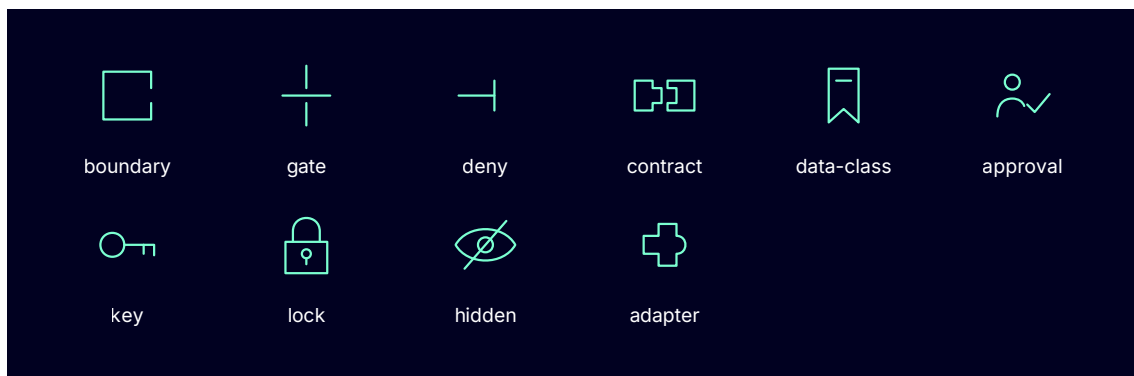
What is built and run: products, tools, stores, registers, clusters, the places they run, what an agent remembers and reads, what it is asked, the folders and scripts it is given, the servers that offer it tools, and the open web.



The systems: app, tool, store, registry, cluster, environment, repo, cloud, queue, memory, document, prompt, folder, script, server, web.

Boundaries

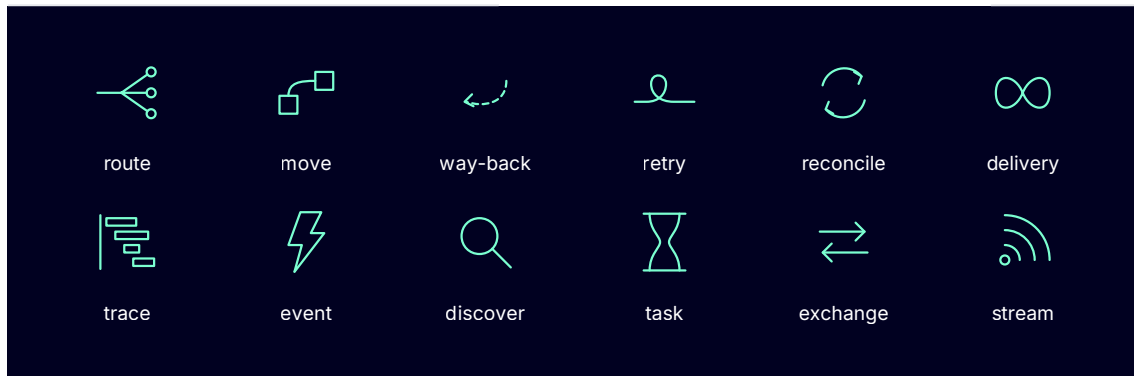
What is allowed: a wall with one way out, a policy at its gate, a call refused, a contract that fits one way, the class a piece of data carries, a person's approval, a credential and what it opens, what is withheld from view, and an adapter between two interfaces.



The boundaries: boundary, gate, deny, contract, data-class, approval, key, lock, hidden, adapter.

Flows

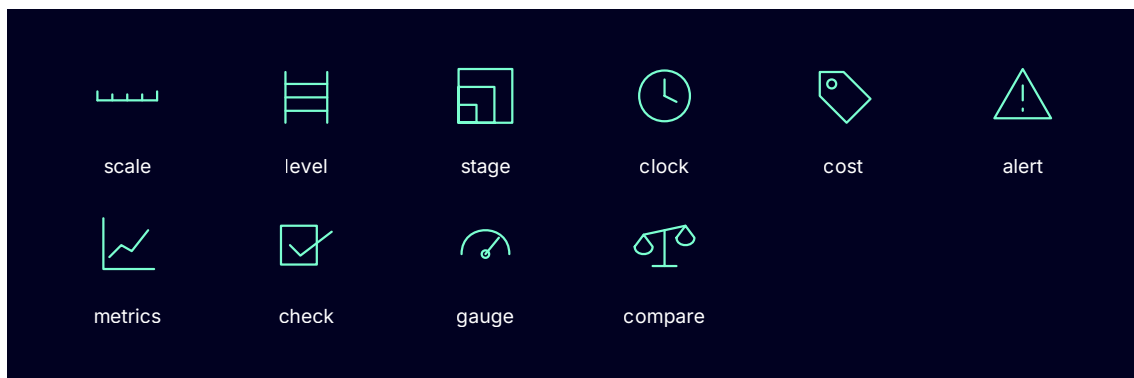
What moves. Most flows are drawn as lines; these are their legend keys, the parts the service marks are made of, and the shapes a protocol's messages take: an event, a discovery, long work polled, an exchange, a stream.



The flows: route, move, way-back, retry, reconcile, delivery, trace, event, discover, task, exchange, stream.

Measures

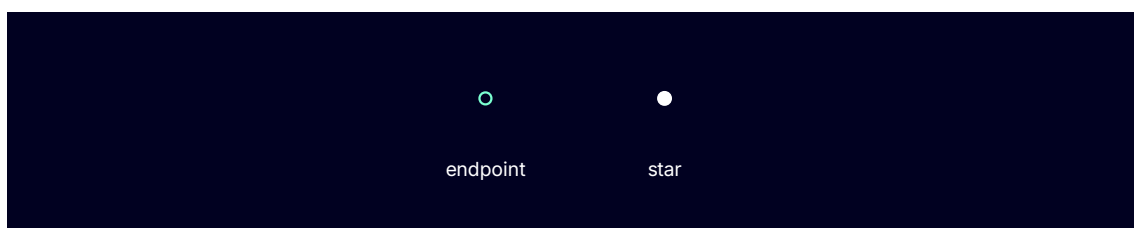
What is counted and when: a scale, levels, stages, time, cost, the metrics watched over time, the alert that wakes someone, a machine's check, a limit and how near it is, and a comparison with a baseline.



The measures: scale, level, stage, clock, cost, alert, metrics, check, gauge, compare.

Outcomes

Where a drawing ends: an endpoint outside its scope, and the one thing the client keeps.



The outcomes: endpoint, star.

Every glyph, with what it stands for and its two files:

Glyph	Use it for	Files
agent	An agent: a model calling tools in a loop	dark , light
model	A model, reached by name through a gateway	dark , light
person	A person: a user, an owner, whoever is on call	dark , light
adversary	An adversary: whoever would turn the system against its owner	dark , light
app	A product or service your users touch	dark , light
tool	A system an agent may call; the notch is its typed interface	dark , light
store	Data at rest: a database, a log, the evidence of a run	dark , light
registry	A register of models, workloads or owners	dark , light
cluster	A cluster and its nodes	dark , light
environ- ment	An environment, standing on the platform it runs on	dark , light
repo	A repository: the trunk and its commits	dark , light
cloud	A cloud provider, as a place things run	dark , light
queue	A queue or an event bus between systems	dark , light
memory	An agent's memory and context, carried from step to step	dark , light
document	A document: what retrieval reads, a policy, a runbook	dark , light
prompt	What is asked: a prompt, a template, a question put to the user	dark , light

Glyph	Use it for	Files
folder	A folder: a skill and its files, a workspace, the roots a server may touch	dark , light
script	A script the agent runs: only its output enters the context	dark , light
server	A server: what offers tools, resources and prompts to a client	dark , light
web	The open web: another origin, the public internet	dark , light
boundary	A wall with one way out: what an agent can touch	dark , light
gate	A policy, enforced where the boundary is crossed	dark , light
deny	A call stopped at the boundary	dark , light
contract	A typed contract: the tongue fits one notch	dark , light
data-class	A class of data: regulated, internal, public	dark , light
approval	A person's approval: the run waits for a yes	dark , light
key	A credential: a token, a secret, a client ID	dark , light
lock	Protected: passes only with a credential	dark , light
hidden	Withheld from view: redacted, masked, held in quarantine	dark , light
adapter	An adapter: one interface in, another out	dark , light
route	One call, routed to one of several	dark , light
move	A cutover: the same thing, moved	dark , light

Glyph	Use it for	Files
way-back	The way back: a rollback, a revert	dark , light
retry	A run that loops before it lands	dark , light
reconcile	A reconciler, keeping what runs equal to what is declared	dark , light
delivery	Continuous delivery: build and run as one loop	dark , light
trace	A trace: every step of a run, recorded	dark , light
event	An event: fired once, with no reply	dark , light
discover	Discovery: finding what is offered, a search, a retrieval	dark , light
task	Long work with a handle, polled until it is done	dark , light
exchange	An exchange: one thing given, another received, a token for a token	dark , light
stream	A stream the client opened, carrying what it subscribed to	dark , light
scale	A measure, read off a scale	dark , light
level	Maturity: levels to climb	dark , light
stage	One thing at growing scale	dark , light
clock	Time: a schedule, a rota, a deadline	dark , light
cost	A cost: a budget, a price per task	dark , light
alert	An alert: an incident, a page to whoever is on call	dark , light

Glyph	Use it for	Files
metrics	What is observed over time: a measure on its axes	dark , light
check	A check: an assertion or a test, passed by a machine	dark , light
gauge	A limit and how near it is: a budget, a context window, a rate	dark , light
compare	Measured against a baseline: with and without, old and new	dark , light
endpoint	An endpoint outside the drawing's scope	dark , light
star	What the client keeps: one per diagram	dark , light

Ten marks, written in the kit

Each service mark is a sentence in these words, which is how the kit was found. The marks stay drawn as they are, tuned as compositions; the glyphs are the atoms they are made of.

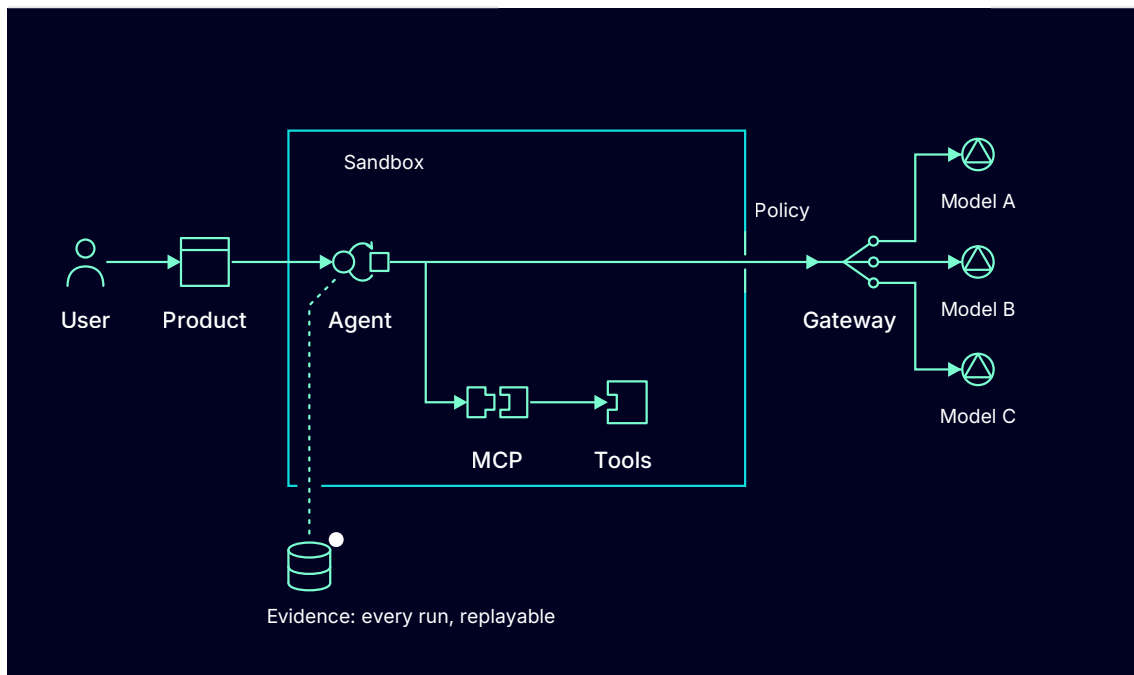
Service	Its mark, in the kit's words
LLM Gateway and Model Routing	route, with the star at its junction
AI Governance and Data Boundaries	a boundary crossed only at its gate, a call, an endpoint
Agent Sandboxing and Observability	boundary and trace
MCP and Tool Integration	contract
Agent Evaluation and Cost per Task	retry and scale
Agent Production Readiness	level
Cloud Operations	delivery
GitOps Operations	repo, and a branch merging back
Migrate to Cloud	move and way-back
Product and MVP Engineering	stage

Composing a diagram

1. **Place things, draw flows.** Actors, systems and boundaries are glyphs. A flow is a line: a solid line with a head is a call, a dashed line is the way back, a line of short dashes is a trace, and one line fanning to several is routing.
2. **One star, on the outcome.** Ask what the client keeps when the work is done, and put the star on that node's upper right, never over a caption.
3. **Containers are boxes.** A sandbox, a cloud or a network edge is a cyan outline, and a gate sits in the wall it controls.
4. **Read left to right; on a phone, top to bottom.** The glyphs that carry a direction, the gate, the contract, the adapter, the route and the exchange, turn a quarter with the flow. Nothing else rotates.
5. **Labels beside glyphs, never inside them,** at 15 units for names and 13 for notes. On a phone every label renders at 12 pixels or more.
6. **48 units for a thing, 36 for a thing repeated,** such as three models behind one gateway. A thing drawn as a box carries its glyph inside, 4 units from the left wall: 36 units, or 24 in a box under 40 units tall and throughout a phone drawing, with the label starting 6 units after it. The star goes one size up from the glyphs beside it, so it measures about 10 pixels on a phone and on a desktop alike.

Example: AI and Agent Engineering

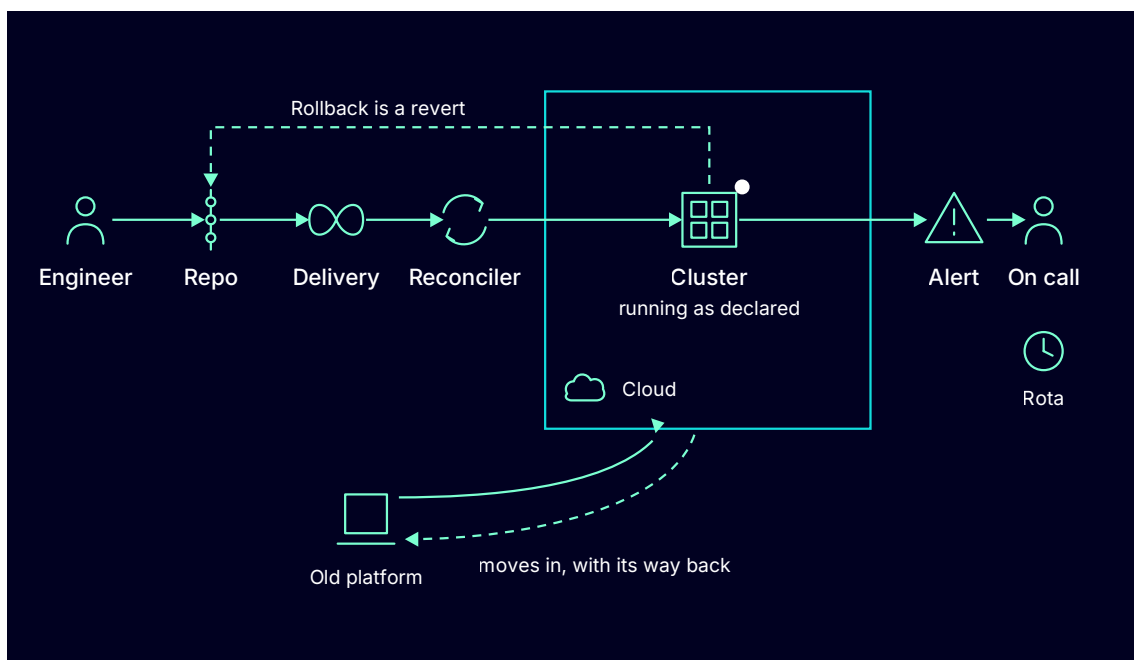
An agent in production, as our AI practice builds it: a sandbox that bounds what the agent can touch, tools reached through a typed contract, models reached only through the policy gate and the gateway behind it, and a trace of every run kept as evidence. The star is on the evidence, because that is what the client keeps: an agent that can show what it did.



Six services in one drawing: readiness, evaluation, governance, the gateway, MCP and the sandbox, all in the kit's words.

Example: Cloud-Native Engineering

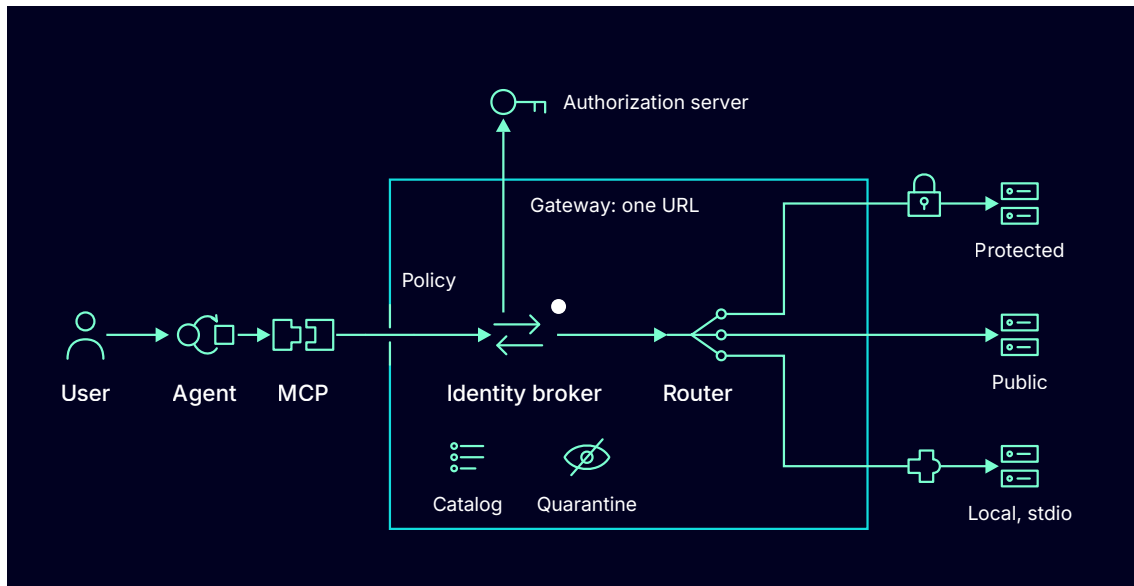
A change from merge to running, as our cloud practice runs it. Delivery builds and tests the change, a reconciler keeps the cluster equal to the repository, and the way back is a revert. Alerts reach whoever is on call, on a rota, and a workload moves in from the old platform with its own way back. The star is on the cluster: running as declared.



Cloud operations, GitOps and migration in one drawing. Both ways back are dashed, because nothing moves before there is one.

Example: an MCP gateway

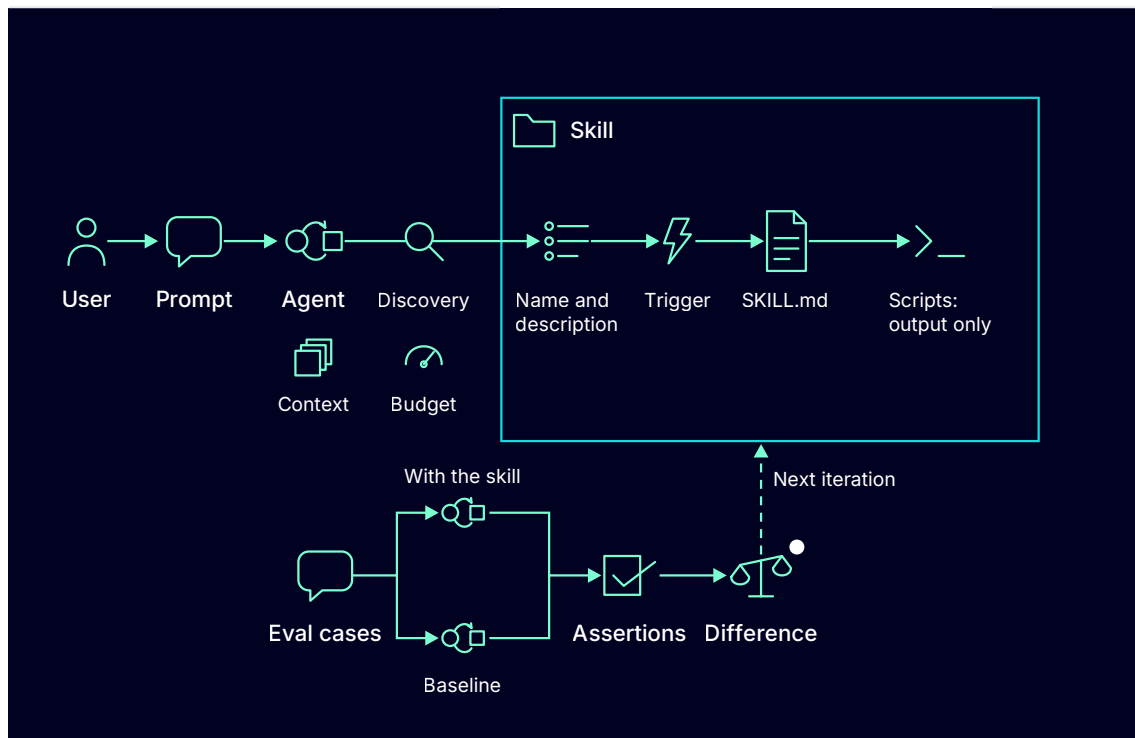
A gateway in front of MCP servers, as we build one for MCP integration: one URL for every server, a policy gate in its wall, an identity broker that trades the client's token for one meant for each server, a router, and a catalog with a quarantine for a server whose tools change under you. Upstream, one server is protected, one is public, and one is a local stdio server behind an adapter. The star is on the identity broker, because that is what the client keeps: no server ever sees a token that was not meant for it.



Six of the new glyphs in one drawing: a credential and the lock it opens, the exchange that trades one for another, what is withheld, a server, and an adapter in front of one.

Example: a skill, loaded and evaluated

An agent skill, as we write and test one: a folder the agent finds by its name and description, which are all of it the context holds until it is needed. A match triggers its instructions, and its scripts run with only their output entering the context, because the context has a budget. Then the same eval cases run with the skill and without it, assertions check each run, and the two are compared. The star is on the comparison, because that is what the client keeps: what the skill buys, measured, and the next iteration it starts.



Eight of the new glyphs: what is asked, a discovery, a folder, an event, a script, a limit, a machine's check, and a comparison with a baseline.

Using the kit

The kit ships as files, for the tools that draw:

- **The whole kit** (<https://stardevs.gg/downloads/stardevs-diagram-kit.zip>), zipped (63 KB): every file below and the licence.
- Every glyph as its own SVG, on dark and on light: the Files column of the table above.
- **An SVG sprite** (<https://stardevs.gg/downloads/stardevs-diagram-kit/stardevs-diagram-kit.svg>) for code (18 KB). Each glyph is a symbol named sd- and its name, as in `<use href="stardevs-diagram-kit.svg#sd-agent">`; its line takes the CSS color, and its star takes --kit-star.
- draw.io libraries, **dark** (<https://stardevs.gg/downloads/stardevs-diagram-kit/stardevs-diagram-kit-dark.xml>) and **light** (<https://stardevs.gg/downloads/stardevs-diagram-kit/stardevs-diagram-kit-light.xml>). In draw.io, choose File, Open Library, and the glyphs appear in the shape panel, ready to drag.

The glyphs are licensed **CC BY 4.0** (<https://creativecommons.org/licenses/by/4.0/>): use and adapt them for anything, commercial work included, with credit to the StarDevs diagram kit and a link to this page. The licence covers the glyphs only; the StarDevs name, the logo and the service marks are not part of it. The full terms are in **the licence file** (<https://stardevs.gg/downloads/stardevs-diagram-kit/LICENSE.txt>).

Four things the kit will not do: fill a glyph, place a second star, turn a glyph other than with the flow, or take a colour from outside the design system. Each of them makes a diagram say something it does not mean.

Published by StarDevs, Inc at <https://stardevs.gg/blog/diagram-kit/>. © 2026 StarDevs, Inc. This PDF is generated from the article at that address; the web version is the one that is kept up to date.