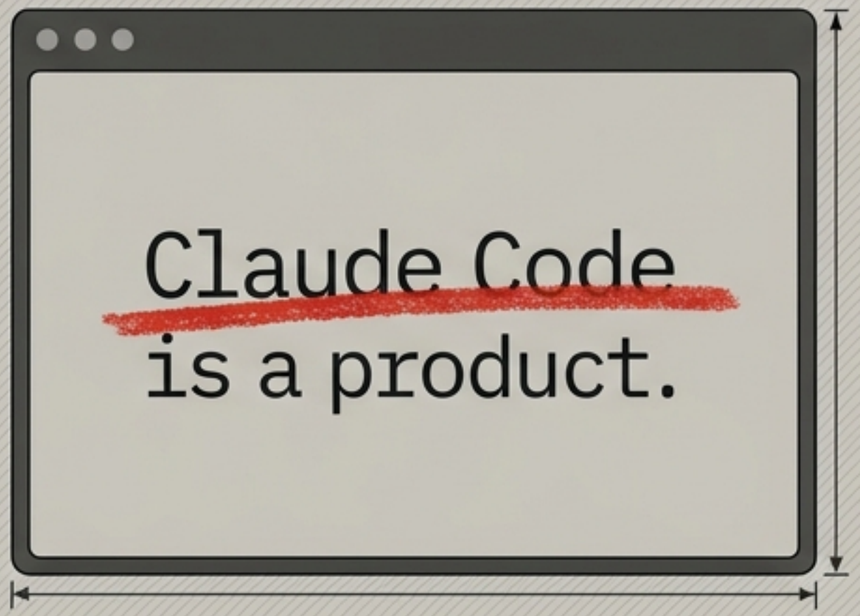
The Architecture of Control

Migrating to Claude Agent SDK and Deploying Safe MCP Creative Workflows

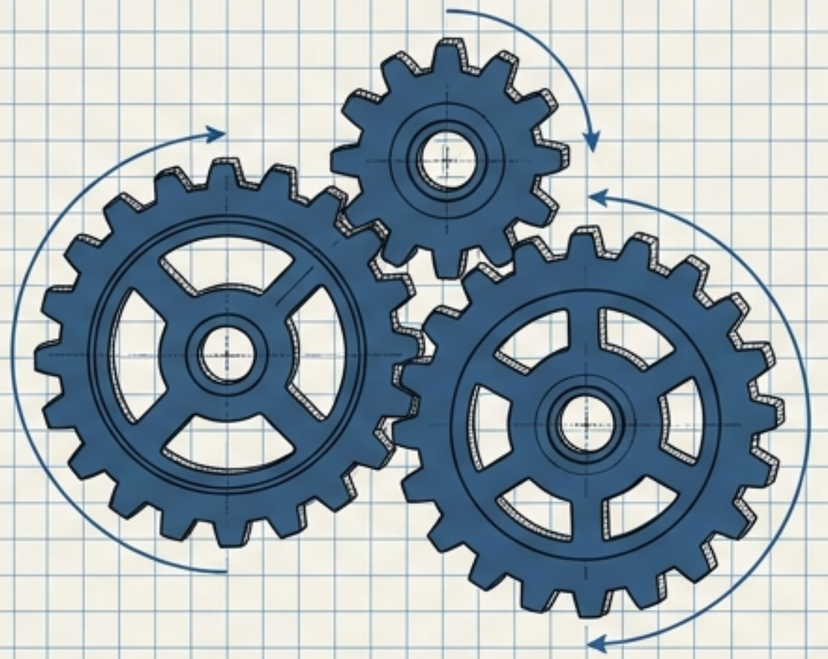
Target Audience: Technical Builders, Creative Technologists, and Software Architects
Framework Version: Updated for mid-2026 infrastructure

Consumer App



- Chat-based terminal interface
- Prohibited partner branding
- No programmatic orchestration

Platform Infrastructure



The Agent SDK is the platform.

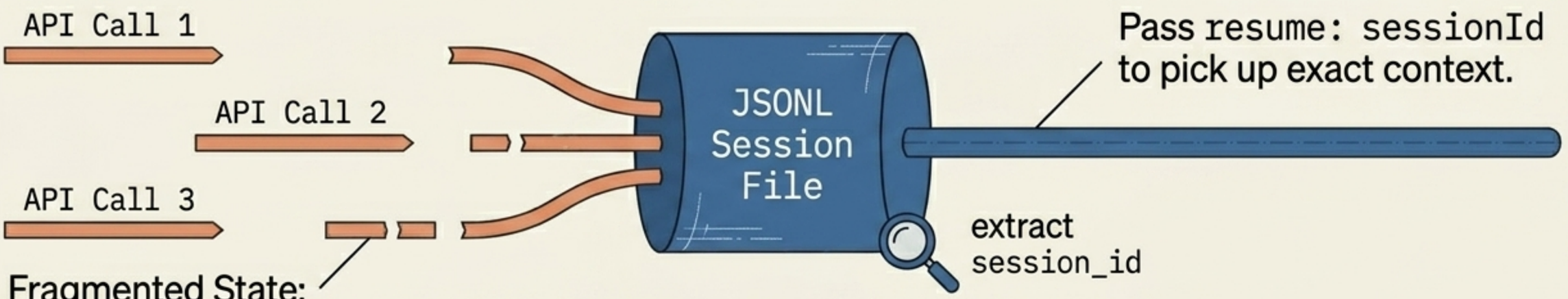
- Headless, programmatic integration
- Custom Python/TypeScript apps
- Autonomous file, shell, and subagent loops

The SDK Migration Matrix

Dimension	Before (April 2026)	After
TypeScript Package	`@anthropic-ai/claude-code`	`@anthropic-ai/claude-agent-sdk`
Python Package	`claude-code-sdk`	`claude-agent-sdk`
Options Class	`ClaudeCodeOptions`	`ClaudeAgentOptions`
Binary Footprint	Separate installation required	Native binary bundled as optional TS dependency
API Continuity	`query()` async generator	Unchanged. Generator API remains strictly identical.

Session Continuity: Stop Re-reading Files

Continuity Thread



1. Listen

Wait for SystemMessage (subtype `init`) before work begins.

2. Extract

Grab the `session_id` (e.g., `sess_01XxXxxXx...`).

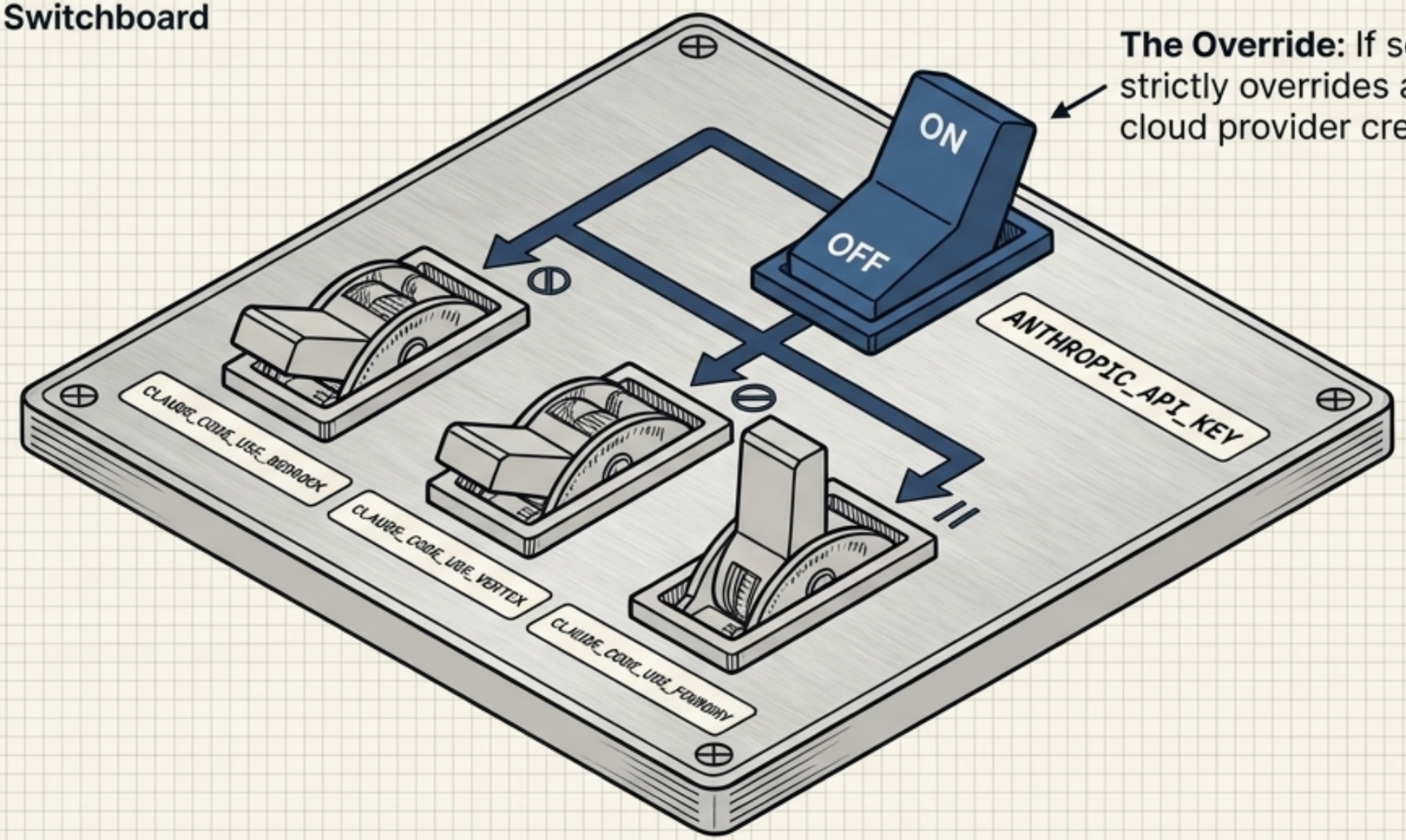
3. Resume

Inject `resume: sessionId` in options for zero redundant tool calls.

Multi-Cloud Auth via Environment

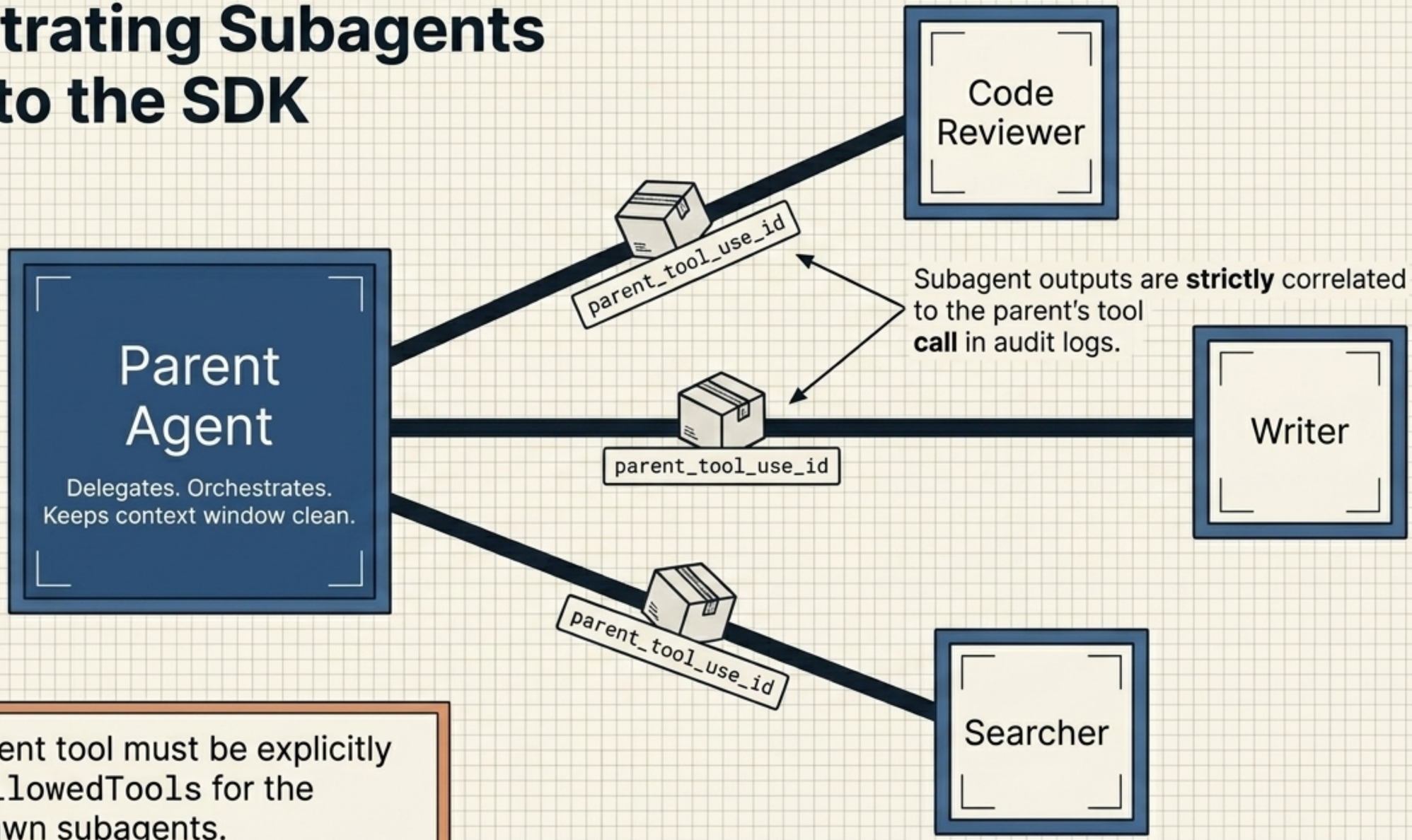
Zero constructor arguments. Configured entirely via the environment.

Toggle Switchboard



The Override: If set, this strictly overrides all cloud provider credentials.

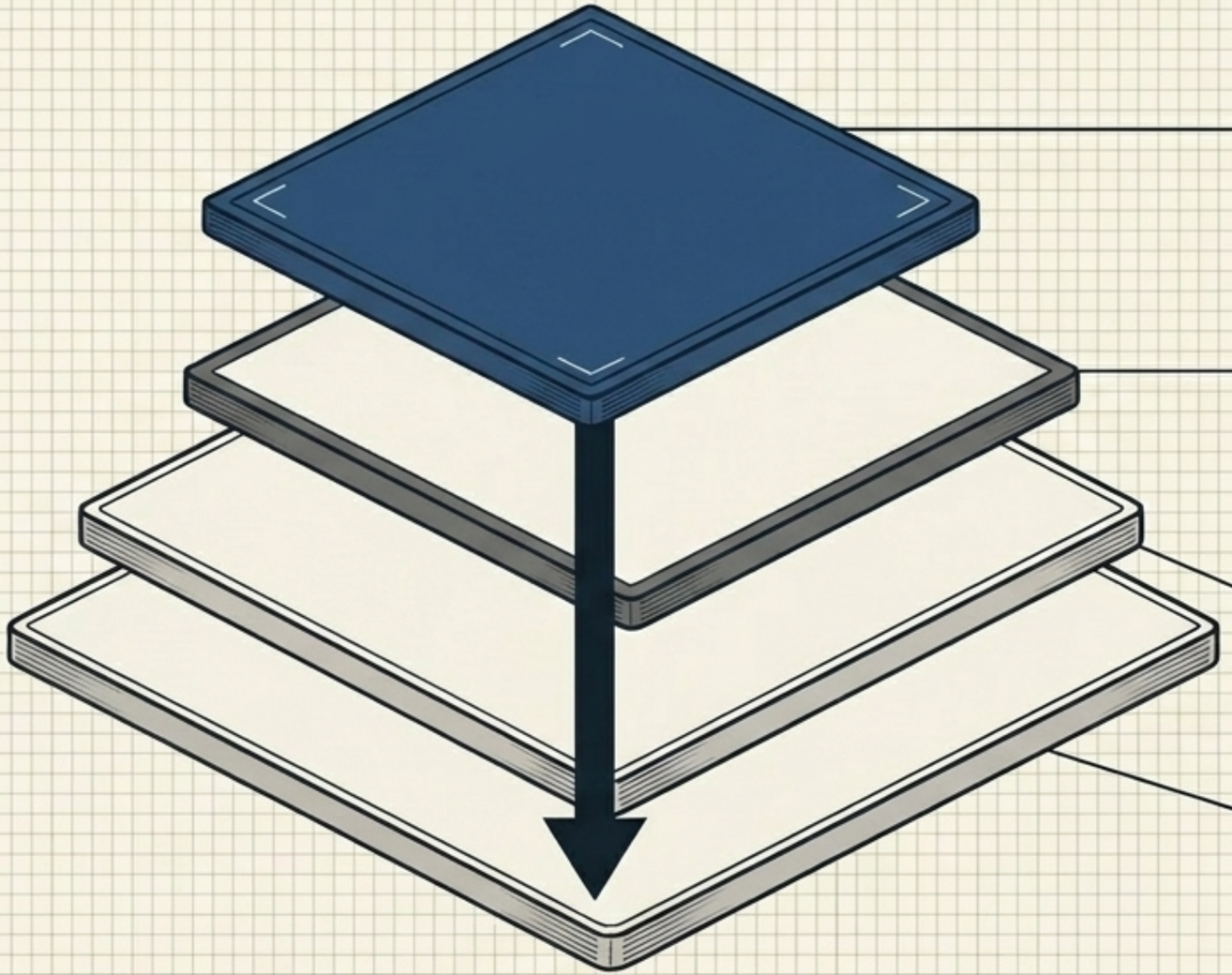
Orchestrating Subagents Native to the SDK



Rule: The Agent tool must be explicitly enabled in `allowedTools` for the parent to spawn subagents.

Configuration Loader Hierarchy

Debug flow: Top layers strictly override bottom layers.



System Prompt
The ultimate override for the current session.

Run Explicit (settingSources)
Used in CI to block global user settings from leaking into automated runs.

Project Scaffolding (.claude/)
Overrides global. Houses hidden primitives like Skills (SKILL.md) and Slash commands.

Global Setup (~/.claude)
Baseline developer settings.

The Agent Tool Risk Matrix

There is no 'allow all' shortcut. Every tool must be explicitly declared.

Safe to Allow Broadly (Read-Only & Search)

- `Read`
- `Glob`
- `Grep`
- `WebSearch`
- `WebFetch`
- `Monitor`
- `AskUserQuestion`

Caution: Version Control Required (Destructive Potential)

- `Write`
- `Edit`

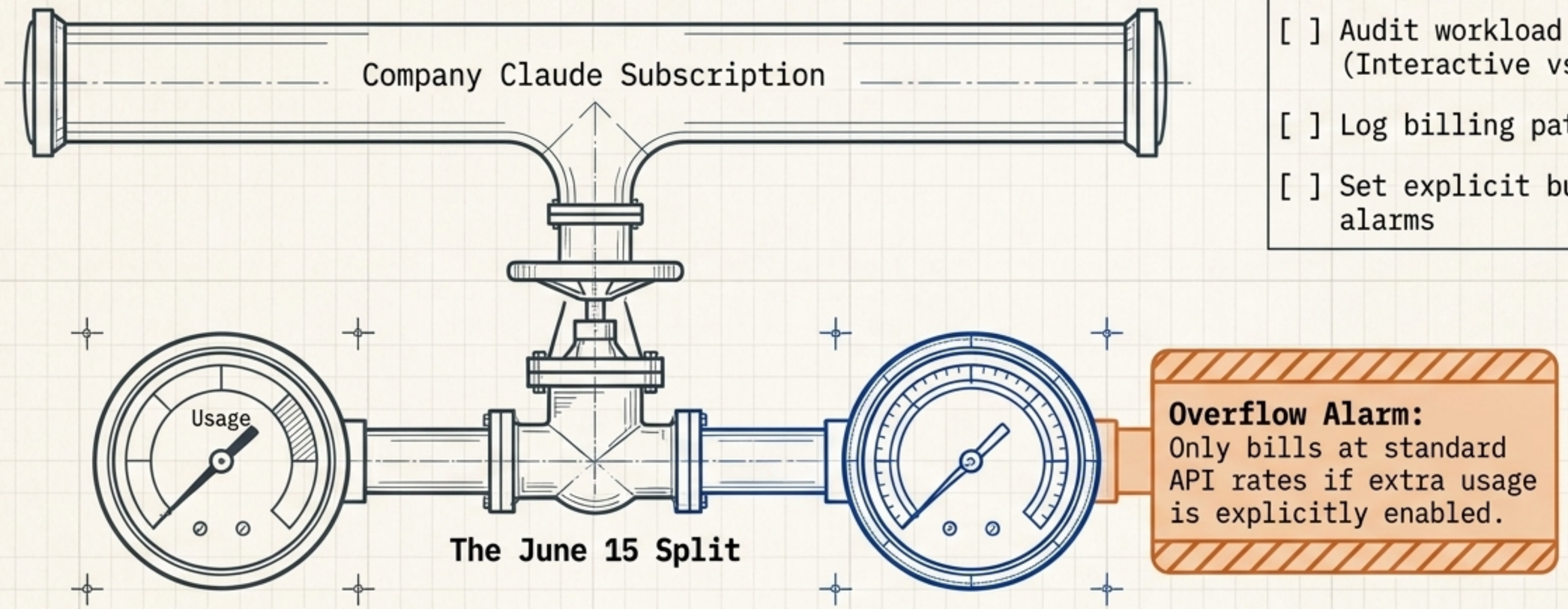
Strict Sandbox Required (Arbitrary Code Execution)

- `Bash`

Strictly reserved for fully sandboxed containers. Never deploy to open developer workstations.

Navigating the June 15 Billing Split

Data Routing Funnel



Interactive Plan

Used by conversational chat and claude -p

Agent SDK Credit

Used by Agent SDK and custom apps.

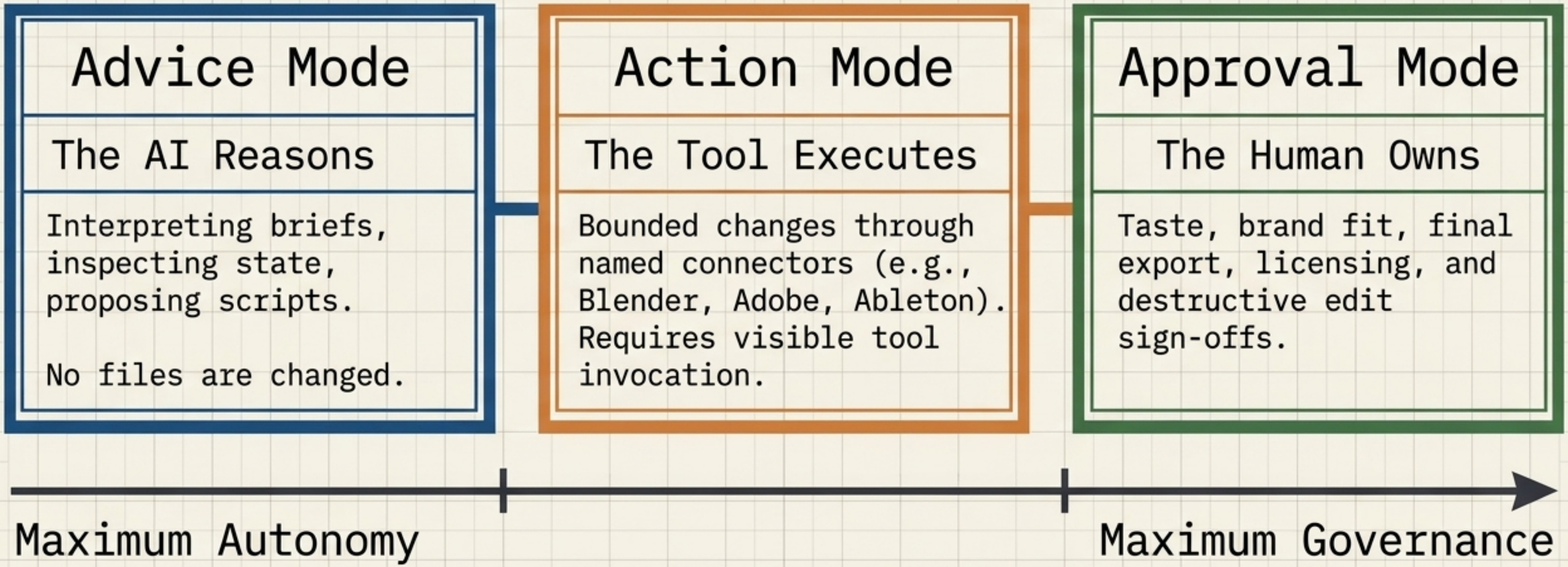
Overflow Alarm:
Only bills at standard API rates if extra usage is explicitly enabled.

Checklist:

- [ ] Audit workload paths (Interactive vs Managed)
- [ ] Log billing paths in CI
- [ ] Set explicit budget alarms

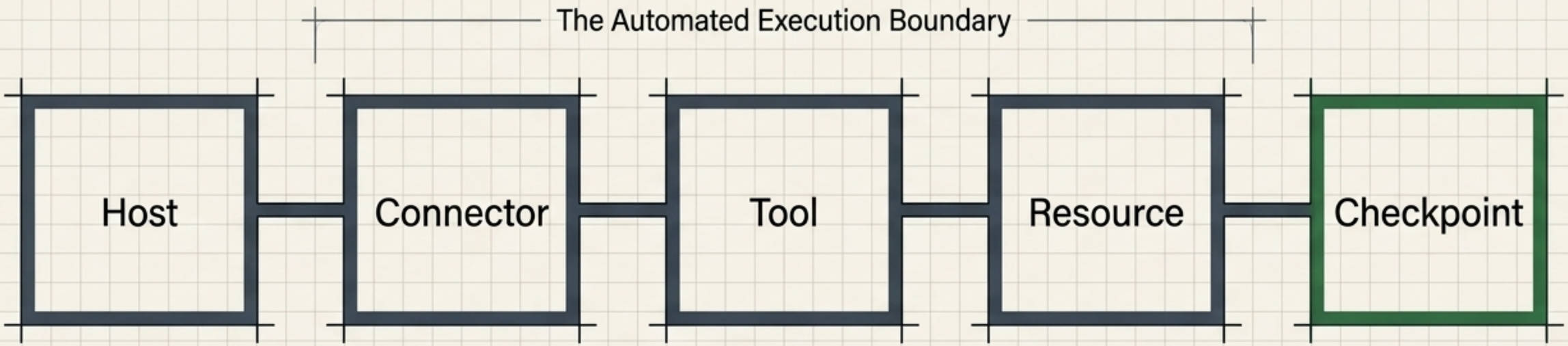
Deploying MCP: Three Operational States

Moving from AI autonomy to absolute human governance.



The Anatomy of an MCP Map

A concrete, 5-node pipeline for designing creative workflows.



The AI interface where the conversation lives.

The bridge to external software (e.g., Affinity, Splice, Resolume).

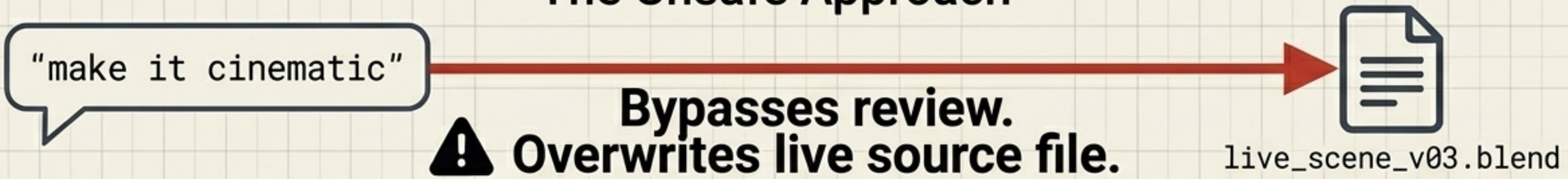
The callable operation (e.g., run a script, query a sample library).

The live file state (.blend scene, audio project, layer stack).

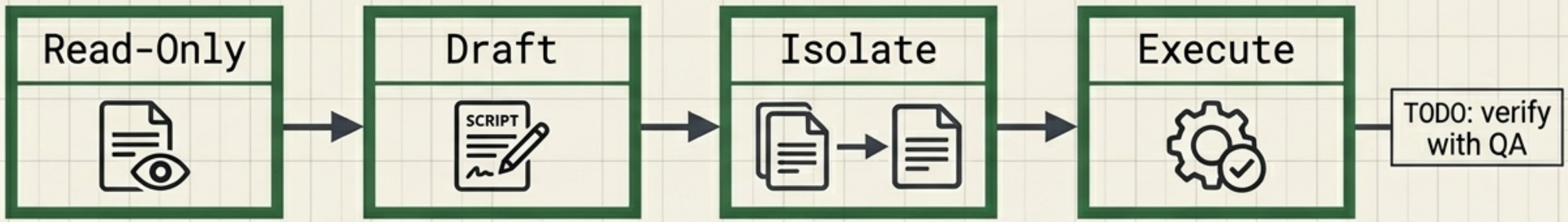
The structured human review step.

Safe Execution: The Duplicate & Verify Approach

The Unsafe Approach



The **Safe** Approach



Claude inspects active scene.

Generates a commented Python script.

System deliberately duplicates “lamp_loop_v03.blend”.

Script applied strictly to the duplicate.

The Creative Control Matrix

Exact operational boundaries between AI drafting and Studio ownership.

Claude Autonomy Zone	Human Ownership Zone
- Generative drafting and ideation - System inspection (summarizing lighting/material state) - Repetitive production task execution - Generating structural checklists from documentation	- Final asset brand fit and art direction - Commercial licensing clearance (e.g., Splice sample rights) - Destructive file overwrites and state commits - Ethical/legal safety and final release delivery

Never surrender creative direction to the model.
Use MCP to expose actions, not to delegate taste.

The 6-Step Production Control Loop

Standard operating procedure for every creative connector action.

1. Brief

Define the precise outcome.

2. Scope

Restrict allowed files and tool actions.

3. Preview

Claude proposes the intended change.

4. Execution

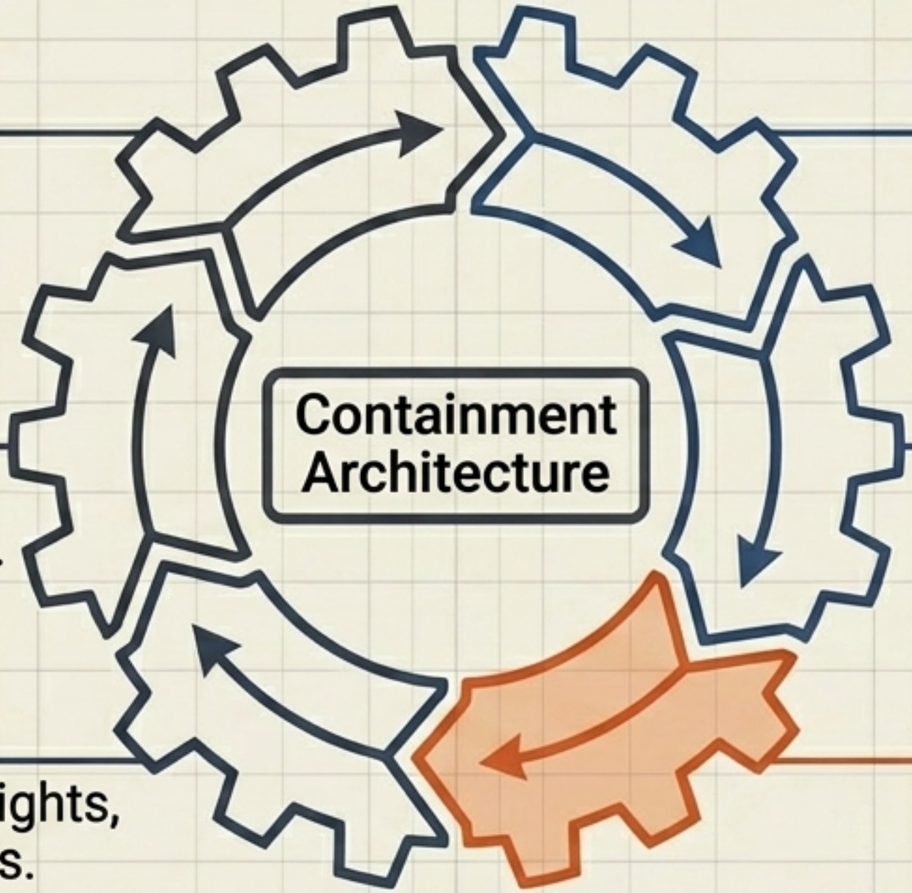
The MCP Connector fires the bounded action.

5. Review

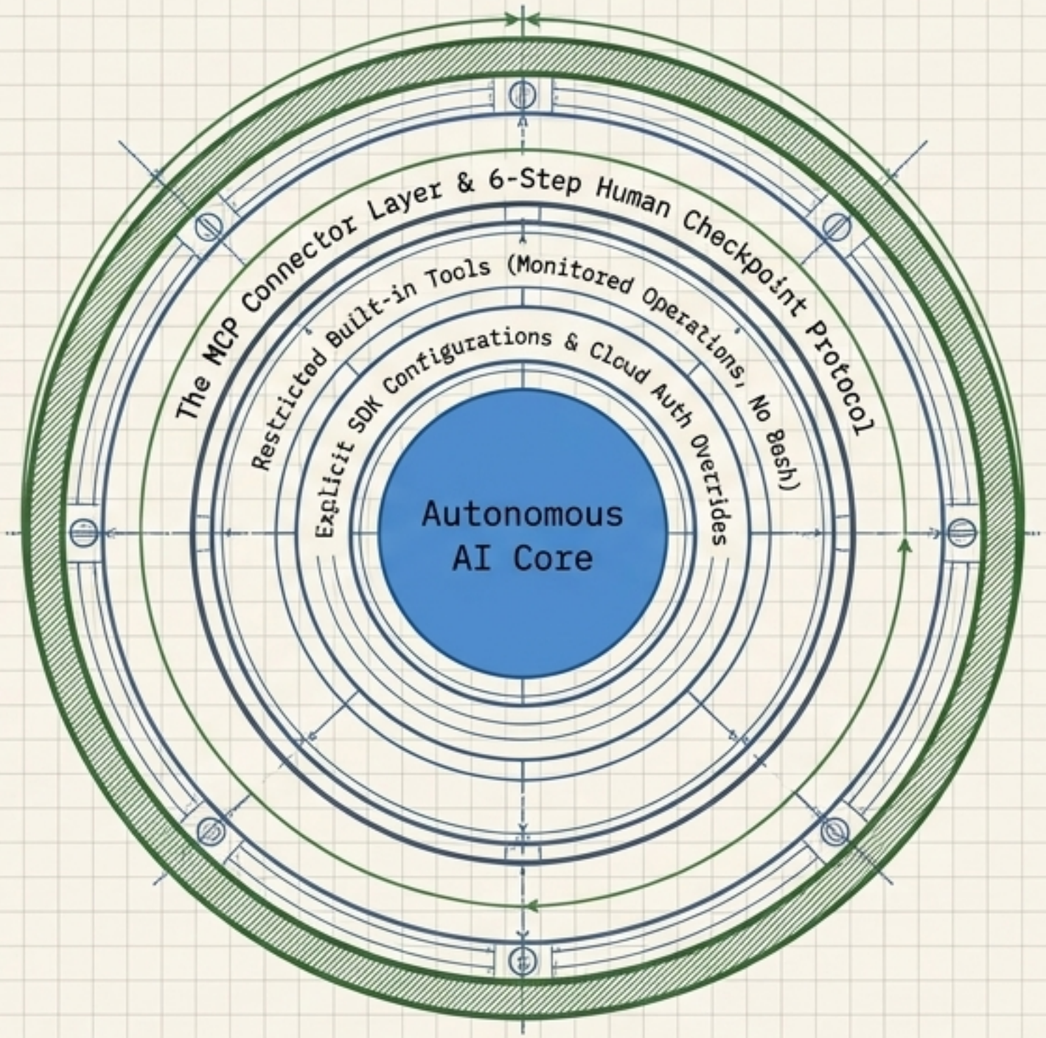
Assess against taste, rights, and delivery constraints.

6. Rollback

Verify the ability to revert to the prior state.



The Bounded Agent Architecture



True automation in production environments isn't about giving the AI more freedom. It is about designing perfect, visible boundaries so the AI can operate at maximum velocity within a secure sandbox.