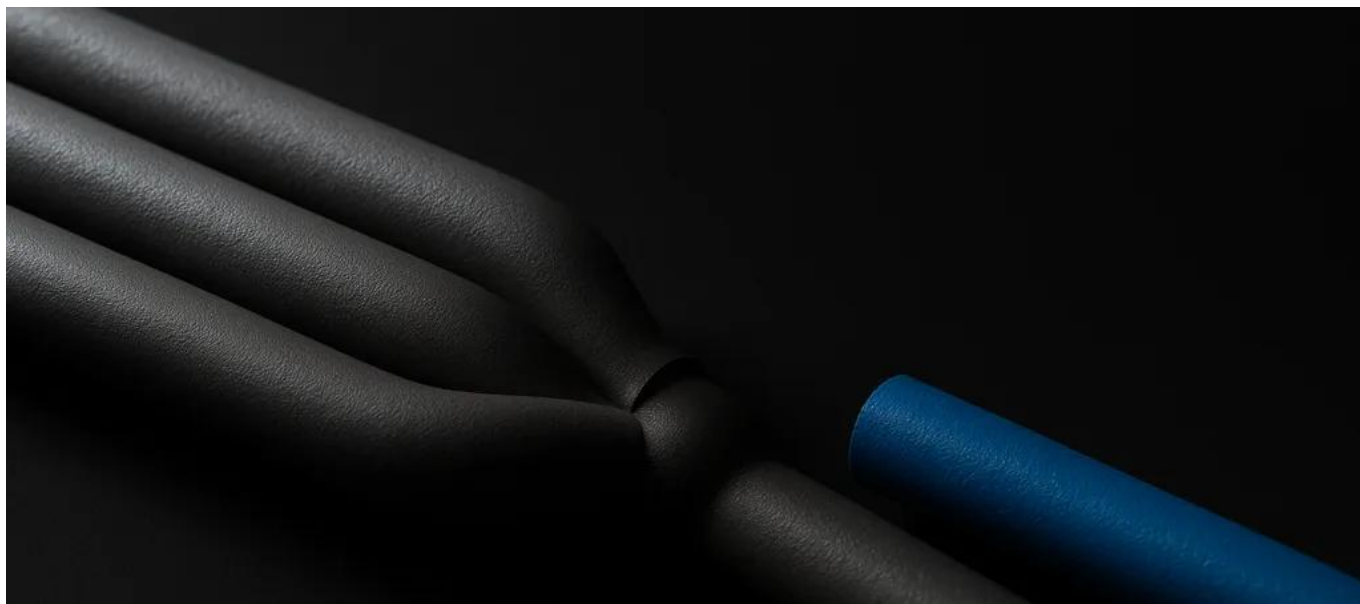




n8n vs Zapier vs Make vs OpenAI Agent Builder: The Honest Automation Stack Comparison for CTOs

OpenAI says Agent Builder and Evals “will no longer be available on the OpenAI platform” from November 30, 2026. The recommended path forward is the Agents SDK, which is code rather than a canvas.

If you picked Agent Builder as your orchestration layer, the visual layer is gone on November 30, 2026 (announced June 3, 2026), and the shortlist collapses back to n8n, Zapier and Make. For most teams that had a working Agent Builder flow, my pick is n8n: it bills per execution rather than per step, and it self-hosts. That second part decides the data question before you have to argue about it.



ChatKit survives the wind-down, the drag-and-drop canvas does not

The [OpenAI deprecations page](#) lists the Agent Builder shutdown as November 30, 2026, with the announcement dated June 3, 2026. The [AgentKit post](#) confirms that both Agent Builder and Evals stop being available on that date and points to the Agents SDK for workflows that continue as code.

Two details in the fine print matter more than the headline. First, the [Agent Builder guide](#) carries the shutdown banner and states that ChatKit remains available. ChatKit is not part of the wind-down, so if your product surface is a chat widget backed by ChatKit, that surface survives. What dies is the drag-and-drop orchestration behind it. Second, the phasing is uneven. Migration write-ups from [the September 2026 docs cycle](#) (via [developersdigest.tech](#) and [ecorpit.com](#)) record Evals going read-only on October 31, 2026 and removed November 30. Agent Builder itself has no documented read-only phase.

That asymmetry is the part I would plan around. A read-only phase means you can still open the thing and read your own configuration after the cutoff. Without one, the safe assumption is that on December 1 you cannot inspect what your flows used to do. Export the graph, the prompts, the tool definitions and the branch conditions to a repo now, while the UI still renders them. Cheap insurance against a documentation gap, and it costs you an afternoon of copy-paste rather than a reconstruction project.

One pricing note, because it shapes the comparison: OpenAI states that the AgentKit tools are “included with standard API model pricing,” with no platform surcharge. Agent Builder was never the expensive part of that stack. The tokens were. So the migration question is about replacing a UI, not escaping a bill.

Executions, tasks and operations are three different units of the same work

This is where vendor pricing tables mislead by construction. n8n bills per execution, Zapier per task, Make per operation. One multi-step workflow is one n8n execution, but many



n8n vs Zapier vs Make vs OpenAI Agent Builder: The Honest Automation Stack Comparison for CTOs

Zapier tasks and many Make operations. A five-step flow that fires 1,000 times a month costs you 1,000 n8n executions, roughly 5,000 Zapier tasks, and roughly 5,000 Make operations. The per-unit price is the smaller half of the story. The multiplier is the bigger half.

ATTRIBUTE	N8N	ZAPIER	MAKE	AGENT BUILDER
Billing unit	Per execution (whole workflow)	Per task (per step)	Per operation (per step)	No platform surcharge; standard API model pricing
Free tier	Self-host free under Sustainable Use License, internal business or non-commercial use	\$0 for 100 tasks/mo, two-step Zaps only	\$0 for 1,000 ops/mo	n/a
Entry paid tier	Starter €20/mo, 2,500 executions (annual)	Professional \$19.99/mo annual (\$29.99 monthly), 750 tasks	Core ~\$9/mo annual (\$10.59 monthly), 10,000 ops (~\$0.001/op)	n/a
Next tier up	Pro €50/mo, 10,000 executions	Team ~\$69/user/mo annual, from 2,000 tasks	Pro ~\$16/mo; Teams ~\$29/user/mo	n/a
Self-hosting	Yes; Business €667/mo self-hosted, 40,000 executions	No	No	No
Licence	Source-available (Sustainable Use License), not OSI open source; .ee . files under separate Enterprise License	Proprietary	Proprietary	Proprietary, shutting down Nov 30, 2026
Integration breadth	n/a	9,000+ apps, ~690 AI apps (July 2026)	n/a	n/a
Longevity signal	205.0k GitHub stars (Sept 17, 2026)	Standalone Agents folded back into Zaps by Sept 9, 2026	n/a	Removed Nov 30, 2026



A few honest gaps in that table. I do not have sourced integration counts for n8n or Make, so those cells read n/a rather than a number I half-remember. Zapier's 9,000+ figure comes from its own [pricing page and app directory](#), with the ~690 AI apps count logged in July 2026. Treat vendor-published catalogue sizes as a measure of surface area rather than depth: a directory entry tells you a connector exists, not that it covers the endpoint you need.

Make's Core rate works out to roughly \$0.001 per operation at 10,000 operations for about \$9 to \$10.59 a month. That is the cheapest unit price in the set by a wide margin, and it is exactly why the unit definition matters. Multiply by steps per run and the advantage narrows or disappears depending on how chatty your workflows are.

Zapier's own cliff is worth stating plainly: the free tier gives you 100 tasks a month and limits you to two-step Zaps, and the next step up is Professional at \$19.99/mo annual for 750 tasks. Two-step Zaps are not agent workflows. If you were using Agent Builder, you were doing something with branches and tools, and Zapier's free tier cannot hold it.

Which platform should replace Agent Builder before November 30

It depends on which part of Agent Builder you were actually using, and my read is that most teams were using two different things and calling them one.

If you were using it as a visual way to wire an LLM to a handful of tools, and the flows are small and stable, the Agents SDK is the direct answer, because it is what OpenAI recommends and because your model costs do not change. You trade a canvas for a repo. For an engineering team with CI already in place, that is no downgrade.

If you were using it as the glue between SaaS systems, where the LLM is one node among fifteen and the rest is Slack, Salesforce, Postgres and a webhook, then you want a general automation platform and the choice is n8n, Zapier or Make.

WHERE I LAND

My take: the billing unit should drive this, not the sticker price. High step counts per run



favour n8n's per-execution model. Low step counts with many distinct integrations favour Zapier's catalogue. Steady mid-volume work with simple flows favours Make's \$0.001 per operation. I would model your three highest-frequency workflows in all three units before reading another pricing page, because the ranking flips depending on shape.

The same n8n Business plan shows €667 in the EU and \$800 in the US

n8n's Business plan is listed at €667/mo in the EU view and \$800/mo in the US view, both captured in early September 2026, for the same 40,000-execution self-hosted plan. That is a regional display difference, not a feature difference. For a Swiss or EU buyer that is a small win, and for a US buyer it is a reason to check which storefront your procurement portal lands on. I would not build a strategy on it, but I would notice it during a renewal.

The licence is the part procurement will actually stop on. n8n's [Sustainable Use License](#) permits free self-hosting for internal business purposes or non-commercial use. Redistribution must be free and non-commercial, and files with .ee. in the name fall under a separate Enterprise License. n8n is source-available, not OSI open source, which matters for procurement and for anyone reselling automation as a service. If automation is part of what you sell to clients, read that licence with your lawyer before you read the pricing page. That is the most common way the "n8n is free" assumption breaks: free for your own operations, not free as a component of what you sell.

Self-hosting is also the whole data-residency answer. Zapier and Make are SaaS, so your payloads transit their infrastructure and your residency story is whatever their terms and regions give you. A self-hosted n8n keeps the orchestration inside your own boundary, so the only thing leaving is what you deliberately send to a model API. For anything touching Swiss or EU personal data under a strict internal policy, that difference is worth more than the price difference. It also shifts work onto you: you now run the thing, patch it, and own its uptime.



Zapier folding Agents back into Zaps is the more interesting signal

Zapier Agents reached general availability with Enterprise MCP support on [April 23, 2026](#). By September 9, 2026 Zapier had folded standalone Agents back into normal Zaps, with Enterprise trial migrations ending August 15, 2026. Two vendors, roughly five months apart, both retired a separate “agent” product surface in the same year.

My read: the standalone agent builder as a product category is not settling. OpenAI removed its canvas and pointed at an SDK, and Zapier absorbed its agent product into the workflow product it already had. The durable shape looks like a normal automation platform with model calls and tool calls as node types, sitting in place of a distinct agent studio. I would weight that heavily when choosing where to rebuild, because the cost of a second migration in 2027 is higher than any of the price gaps in the table above.

If you are wiring model context and tools into any of these, the auth posture of your MCP servers deserves a separate look; I went through that in detail in [MCP, explained](#). And if your bill is dominated by tokens rather than platform fees, which it will be once the orchestration layer is nearly free, [prompt caching economics](#) move more money than any of these tiers.

Export this week, count units next week

Export first, decide second. Agent Builder has no documented read-only phase, so treat the configuration as perishable and get it into version control this week. That step is independent of which platform you choose, and it is the one with a hard deadline.

Then count. Take your three highest-volume flows and write down steps per run and runs per month. Convert to executions, tasks and operations, and price each. If your flows average five or more steps, n8n’s per-execution billing usually wins on arithmetic alone, and the Pro tier at €50/mo for 10,000 executions covers a surprising amount of real production traffic. If your flows are two or three steps but touch a long tail of niche SaaS, Zapier’s 9,000+ catalogue is worth paying the per-task multiplier for, because building a connector yourself costs more than the tasks do.

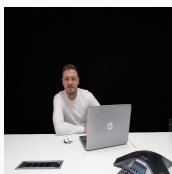


n8n vs Zapier vs Make vs OpenAI Agent Builder: The Honest Automation Stack Comparison for CTOs

One catch on n8n's free tier, since it trips people up: the Sustainable Use License allows self-hosting free for internal business purposes or non-commercial use, and requires any redistribution to be free and non-commercial. If automation is part of what you sell, **that free tier is not your tier**, and files named with `.ee.` sit under a separate Enterprise License.

Make is the one I find hardest to place honestly. Its unit price is the lowest in the set at roughly \$0.001 per operation, its free tier at 1,000 operations a month is ten times Zapier's, and its Teams tier at ~\$29/user/mo undercuts Zapier's ~\$69/user/mo. Those are real advantages. What I do not have from the research is an integration count or a self-hosting option, and no self-hosting means no residency answer beyond the vendor's own. So I would shortlist Make for cost-sensitive, non-sensitive workloads and drop it from anything where the data boundary is the requirement.

For the team migrating off Agent Builder with a mixed workload, personal data in the payloads, and more than a couple of steps per run, **n8n self-hosted** is the choice I would defend in a review: per-execution billing, a real residency story, and 205.0k GitHub stars on September 17, 2026 (203.5k on September 6) as a rough proxy for the project not disappearing the way the thing you are migrating off just did. The licence reading and the ops burden are the price of that, and both are payable.



WRITTEN BY

Artur Markus

Software engineer in Basel building AI infrastructure and agentic systems, with a background in pharmaceutical automation, GxP compliance, and validation. I help teams turn AI from a chatbot into a reliable operational layer.

[Book a meeting →](#)