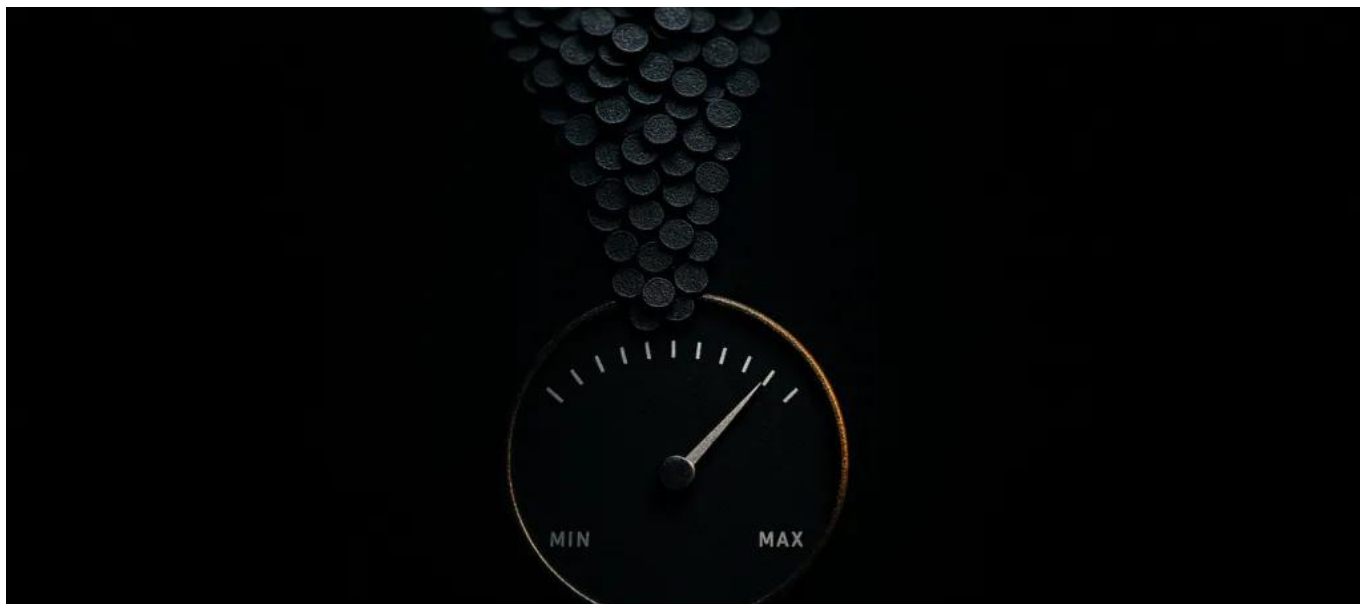




Microsoft Ships Decision-1: a 9B Model Built on Qwen3.5 That Costs
\$0.042/M In and Nothing Out



Microsoft Ships Decision-1: a 9B Model Built on Qwen3.5 That Costs \$0.042/M In and Nothing Out

Microsoft is charging \$0.042 per million input tokens and zero for output on its newest model. The base weights come from Alibaba's Qwen3.5-9B, and Microsoft did not release the result.

On October 9, 2026 Microsoft launched Microsoft-Decision-1 in Foundry public preview: a 9B model post-trained on Alibaba's open-weight Qwen3.5-9B that returns a structured choice plus a calibrated probability instead of prose. Input costs \$0.042 per million tokens, output is free, context tops out at 32,768 tokens, and the weights stay with Microsoft. If your agent stack runs routing, classification or LLM-as-judge on a frontier model, that line item is now mispriced.

The announcement went up on Microsoft's [Command Line site](#) on October 9, 2026, framed as a model for fast decision-making. Microsoft claims the highest accuracy across 36



Microsoft Ships Decision-1: a 9B Model Built on Qwen3.5 That Costs \$0.042/M In and Nothing Out

benchmarks covering nearly 150,000 questions, with the evaluation questions held blind from training. Reported P50 latency is roughly 35 times lower than GPT-6 Sol and 4.5 times faster than Quyet-1.0-Large. An internal Xbox Research test on more than 10,000 feedback items reported quality comparable to GPT-6 Sol at over 14 times the speed and 200 times lower cost.

The constraint is the interesting part. Decision-1 supports yes/no, multiple-choice, rating-scale and rubric-based evaluation. It does not do conversation, summarization or translation, and Microsoft says so explicitly in [the coverage of the launch](#). You hand it options, it hands back JSON with a selection and a probability per option. That is the whole product surface.

A model that only picks is worth more than its price tag suggests

Production agent stacks spend a surprising share of their token budget on decisions nobody reads. Which tool to call. Whether a support ticket is a refund request, whether the generated answer passes the rubric, whether to escalate. These calls go to a general-purpose model because that is what the SDK defaulted to, and each one burns a frontier price for a task whose output is three tokens wide.

The free output pricing is the structural tell. Microsoft can give output away because the output is a short JSON object with bounded cardinality. There is no generation loop to subsidise. That pricing shape only works for a model that physically cannot ramble, which is also why Microsoft had to strip the conversational capability rather than merely discourage it.

Calibrated probabilities matter more than the price, and most of the coverage has buried them. A frontier model asked "is this a refund request, yes or no?" gives you a token, and the logprob behind that token is a weak proxy for confidence because the model was never trained to make it mean anything. A model trained to emit a calibrated probability lets you set an actual threshold: route everything above 0.9 automatically, send 0.6 to 0.9 to a second pass, queue the rest for a human. That is a control surface you can tune with a dial instead of a prompt rewrite.



Is Decision-1 worth switching your routing layer to?

Conditionally yes for classification and judging. For anything with state, not yet. Three things limit it.

Context is 32,768 tokens, per [Foundry documentation](#). That is fine for a ticket, a diff, a single rubric evaluation. It is not fine for a judge that needs to read a long agent trajectory, and it rules out the “stuff the whole conversation in and ask if we should escalate” pattern that a lot of teams currently use.

The weights were not released. [Coverage of the launch](#) describes the service as API-only, so the cheapest call in your stack also becomes a hosted dependency with a Microsoft-shaped availability profile. At these prices that is probably acceptable, but it is a different risk category than a 9B model you could have run yourself. Microsoft also says training used public and synthetic data with no customer data.

The benchmark claim is Microsoft’s own. Thirty-six benchmarks, roughly 150,000 blind questions, highest accuracy: that is a strong claim made by the vendor about the vendor’s model, and nobody outside Microsoft has reproduced it as of the announcement. The Xbox Research number has the same shape. “Quality comparable to GPT-6 Sol” on 10,000 feedback items is a real-looking internal result rather than a published eval with a methodology section attached. I treat it as a reason to run your own test.

One more figure deserves scrutiny before anyone wires this into production. Secondary coverage cites 98.7% perturbation stability, meaning decisions held under small input changes. That figure comes from AI Weekly, not from Microsoft’s own announcement, and perturbation stability is exactly the property that decides whether you can trust an automated routing threshold. **Verify it on your own data before you wire it into anything irreversible.**

The Qwen3.5 base is the part nobody wants to discuss

Microsoft shipped a product built on an Alibaba open-weight model and said so in the Tech Community post. It means Microsoft’s answer to “we need a cheap, fast, narrow model” was to take the best available open base at this parameter count and post-train it



Microsoft Ships Decision-1: a 9B Model Built on Qwen3.5 That Costs \$0.042/M In and Nothing Out

hard for one behaviour, rather than training from scratch or distilling from GPT-6 Sol.

My take: that is the correct engineering call and it tells you where the value sits. At 9B parameters the base model is close to a commodity. What Microsoft sold is the post-training, the calibration and the eval suite, and none of those shipped as weights. For anyone evaluating build-versus-buy on a classification layer, the base is cheap and the calibration is the hard part.

WHERE I LAND

The headline comparison in most coverage is Decision-1 versus GPT-6 Sol, and I think that framing is wrong. The real competitor is the fine-tuned BERT-class classifier or small open model that mature teams already run for exactly these tasks, often at near-zero marginal cost on hardware they own. Against GPT-6 Sol, Decision-1 is 200 times cheaper. Against a DistilBERT you already deployed, it is more expensive, slower on the network hop, and adds a vendor dependency, but it arrives with no labelled data requirement and no training loop. That is the actual trade: you are buying out of a labelling project, not out of frontier pricing. Teams that never had the data to fine-tune anything are the ones who gain most here.

The cost story only holds if you measure the whole call

Free output tokens look decisive, and then cost accounting complicates it. The input side is where the spend lives: a rubric-based evaluation that ships a long rubric plus the candidate answer can run several thousand input tokens per call, so 200 times cheaper per unit does not automatically mean 200 times cheaper per decision unless you restructure your prompts for a narrower model.

There is also the caching question. If your current judge calls on a frontier model reuse a large static rubric prefix, you may already be paying a cached rate rather than list price, and I wrote about [how much that actually changes the arithmetic](#) for CTOs earlier this year. Compare Decision-1's \$0.042 against your effective cached rate, not against the published frontier list price, or you will overstate the win by a factor you cannot defend in a budget review.

Latency is the cleaner argument. A 35 times lower P50 against GPT-6 Sol changes what you can do synchronously. Decisions that currently run async because a slow judge call



would wreck the user-facing path can move inline. A price cut does not do that.

Instrument your label-shaped calls before you migrate anything

Count how many of your model calls return a label, a score or a routing decision rather than text a human reads. If that number is under ten percent of your spend, Decision-1 is a curiosity. If it is over forty percent, which I would expect for any heavily agentic stack, it is a line item worth a sprint.

Then take your hardest classification task, the one where the prompt has three paragraphs of edge-case instructions bolted on over six months, and run it against Decision-1 in Foundry preview with your own held-out set. Measure agreement with your existing model and, separately, agreement with your human labels where you have them. Those two numbers diverge more often than people expect, and the gap is where the switching decision actually gets made. The same discipline applies here as in any harness evaluation, which is why I keep arguing that [identical scaffolds produce wildly different benchmark results](#) depending on details nobody documents.

Keep a fallback path. Public preview means the API contract can move, and OpenRouter access was announced with some coverage describing it as initially coming soon, which suggests availability is still settling. Abstract the call behind your own interface so a rollback is a config change.

Every major lab ships a narrow decision model by mid-2027

I expect that within twelve months, because the economics Microsoft just demonstrated are not proprietary. Free output pricing on a bounded-cardinality task is a pricing structure anyone can copy, and the base models are open. The differentiation will land on calibration quality and on whoever publishes verifiable reliability metrics rather than vendor-run benchmark sweeps.

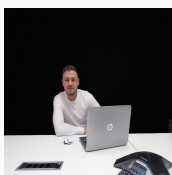
I also expect pressure on Microsoft to release weights or offer a dedicated deployment. A



Microsoft Ships Decision-1: a 9B Model Built on Qwen3.5 That Costs \$0.042/M In and Nothing Out

9B model that handles the highest-volume, lowest-value calls in a stack is precisely the thing enterprises want on their own silicon, and the Qwen3.5 base makes “we can’t open it” a harder position to hold than it would be for a from-scratch model.

The prediction I am least confident about: Satya Nadella [said Microsoft is testing the model for incident response, quality control and scientific discovery](#). Incident response and quality control fit a calibrated classifier cleanly. Scientific discovery does not, at least not in any form I can reconstruct from a 32,768-token context and a multiple-choice output format, and I would want to see what that workflow actually looks like before reading anything into it.



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 →