Building a Distributed AI Swarm for Under $1,100

February 12, 2026 · by Mark Bartlett

Download this guide as PDF

> **Quick Answer:** A three-node distributed AI swarm costs $1,026: an RTX 3090 from r/hardwareswap ($850), a Lenovo ThinkCentre M710Q from eBay ($85), a Raspberry Pi 5 2GB ($65), a TP-Link gigabit switch ($16), and cables ($10). The 3090 handles 32B+ models and heavy inference. The M710Q runs embeddings, small model queries, and API routing on 35 watts. The Pi coordinates the swarm, monitors health, and routes tasks. Tailscale connects everything for free. This is a real build that runs on my desk right now.

📚 **More on this topic:** Rescued Hardware, Rescued Bees · From 178 Seconds to 19 · Best Used GPUs for Local AI · Budget Local AI PC · Planning Tool

I spent $1,026 on three machines, a switch, and some cables. Together they form a distributed AI cluster that routes queries to the right hardware automatically, runs 32B models on the heavy node, handles embeddings and light tasks on a machine that draws less power than a lightbulb, and coordinates the whole thing from a $65 single-board computer.

This is the parts list and the reasoning behind each choice. If you want the philosophy, read Rescued Hardware, Rescued Bees. This article is the spreadsheet.

The Bill of Materials

Component	Source	Price	Role
RTX 3090 (used, in workstation)	r/hardwareswap	$850	Heavy inference: 32B+ models, code review, document analysis
Lenovo ThinkCentre M710Q Tiny	eBay (fleet liquidation)	$85	Light inference: embeddings, 7B models, API routing
Raspberry Pi 5 (2GB)	PiShop US	$65	Coordinator: health monitoring, capability mapping, task routing
TP-Link TL-SG105 (5-port gigabit switch)	Amazon	$16	Wired networking between all nodes

Component	Source	Price	Role
Cat6 Ethernet cables (3ft, 3-pack)	Amazon	$10	One per node
Total		**$1,026**	

$74 left in the budget if something goes wrong. In practice, nothing has gone wrong with this build in the weeks it's been running. The most expensive debugging I've done was figuring out that the M710Q needed its BIOS updated to boot Ubuntu cleanly, which cost me an hour and a USB stick.

Node 1: The RTX 3090 Workstation ($850)

The RTX 3090 is still the value king for local AI in 2026. Twenty-four gigabytes of VRAM for $850 used. The only current-gen card with 24GB under $1,000 is… also the 3090, because NVIDIA decided 16GB was enough for the RTX 5070 Ti and priced the 4090 at $1,400+ used.

The memory shortage that's been driving up RAM and SSD prices has hit the GPU market too. A year ago you could find 3090s for $600-700. Today $800-900 is the range on eBay, with r/hardwareswap and Facebook Marketplace occasionally dipping lower if you're patient and local.

What the 3090 does in the swarm:

- Runs Qwen 2.5 32B at Q4_K_M with room to spare
- Handles document analysis, code review, and complex reasoning queries
- Generates 45+ tokens per second on 7B models, 18-20 tok/s on 32B
- Processes about 20% of total queries but all the hard ones

The workstation itself is a mid-tower I already owned. Any PC that can physically fit a 3090 (it's a three-slot card, 313mm long) and has a 750W+ power supply works. If you're buying the workstation too, budget another $200-400 for a used Dell Precision or HP Z-series from eBay. That pushes you over $1,100 but you get a complete system.

I'm assuming you already have a desktop that can take the card. If not, our $500 budget build guide covers a full system.

Power draw

The 3090 pulls 350 watts under full inference load. The whole system draws around 400-450W. At $0.18/kWh (the US average as of early 2026), that's about 8 cents per hour of active

inference. In practice, this machine is idle most of the time. The swarm only routes heavy queries here. Idle draw is around 60W for the full system.

Node 2: The ThinkCentre M710Q ($85)

This is the node that surprises people.

The Lenovo ThinkCentre M710Q Tiny is a one-liter PC that companies buy in bulk, use for three years, and dump on eBay by the hundreds when the lease expires. An i5-7500T with 8GB RAM and a 256GB SSD runs $75-100 depending on the day. I paid $85 for mine.

It sits behind my monitor. I forget it's there until I check the swarm dashboard and see it handling 60-70% of all queries.

What the M710Q does in the swarm:

- Runs nomic-embed-text for all embedding tasks in my RAG pipeline
- Handles 7B model inference on CPU at about 8 tokens per second
- Is the API endpoint that clients connect to
- Routes queries it can't handle to the 3090

Eight tokens per second sounds slow. It is slow compared to GPU inference. It is fast enough for "what's the capital of France" and "summarize this paragraph" and "classify this email as spam or not." The majority of daily queries don't need a 32B model. They need a 7B model and a correct answer. The M710Q delivers that on 35 watts.

Why the M710Q and not something newer

The M710Q hits a specific price-to-usefulness ratio that newer machines don't match for this role.

Mini PC	CPU	RAM	Price	Power Draw	Notes
M710Q (i5-7500T)	4C/4T, 2.7GHz	8GB	$85	35W load	Sweet spot for price
M910Q (i5-7500T)	4C/4T, 2.7GHz	8GB	$110	35W load	Same CPU, $25 more for vPro
M720Q (i5-8400T)	6C/6T, 1.7GHz	8GB	$140+	40W load	8th gen, two extra cores
Intel NUC 11	i5-1135G7	16GB	$200+	28W TDP	Faster but 2x the cost

The M910Q is the same machine with a “business” label and a $25 markup. Skip it. The M720Q gets you 8th gen Intel with two extra cores, which matters if you’re doing heavier CPU inference, but costs $55-65 more. For embeddings and light 7B inference, the M710Q’s quad-core i5 is enough.

If you buy one of these and it comes with 4GB RAM, add a second 8GB SO-DIMM stick for $12-15 on Amazon. The M710Q has two SO-DIMM slots and supports up to 32GB.

Power draw

11-13 watts idle. 30-45 watts under CPU load. The 65W external brick is included with every eBay unit I’ve seen. Annual electricity cost running 24/7: about $20. That’s less than one month of a ChatGPT subscription.

Node 3: The Raspberry Pi 5 Coordinator ($65)

The Pi doesn’t do inference. It watches the other nodes and directs traffic.

I’m using a Pi 5 2GB for this because the coordinator role needs almost no memory. It runs a lightweight daemon that:

- Monitors which nodes are online via heartbeat checks
- Maintains a capability map (which node has which GPU, how much VRAM, what models are loaded)
- Exposes the swarm API for health checks and status queries
- Routes incoming tasks to the appropriate node based on the capability map

The Pi 5 2GB costs $65. The 1GB model would work fine for this role at $45 and would bring the total build to $1,006, but the 1GB was out of stock when I ordered.

Why prices went up

If you’re wondering why a Raspberry Pi costs $65 when they used to cost $35, blame the memory shortage. LPDDR4 prices have spiked because AI infrastructure is consuming memory fab capacity worldwide. The Raspberry Pi Foundation has raised prices twice since late 2025:

Pi 5 Model	Original Price	Current Price (Feb 2026)
1GB	$45 (new)	$45

Pi 5 Model	Original Price	Current Price (Feb 2026)
2GB	$50	$65
4GB	$60	$75
8GB	$80	$110
16GB	$120	$205

The Foundation says these increases are temporary and will reverse when memory prices normalize. For a coordinator node, the 2GB model is more than enough. Don't buy the 8GB or 16GB for this role. That's $45-140 of RAM you'll never touch.

Alternatives to the Pi

If you already own any single-board computer or old laptop, use that instead. The coordinator role needs a machine that stays on, has a network connection, and can run a Python daemon. A ten-year-old laptop with a broken screen works. A first-gen Raspberry Pi 4 works. An Orange Pi 5 works. Whatever you have in a drawer.

The Network: Wired Ethernet + Tailscale ($26)

Local wiring

A TP-Link TL-SG105 five-port gigabit switch costs $16. Three Cat6 cables cost $10. Plug everything in. That's the local network.

Do not use WiFi for the cluster nodes. The latency difference matters:

Connection	Latency	Throughput
Wired Gigabit	~0.5ms	1 Gbps (stable)
WiFi 5GHz	~4.6ms	300-866 Mbps (variable)
WiFi 2.4GHz	~12.9ms	50-150 Mbps (variable)

For single queries, a few milliseconds of extra latency is invisible. For distributed inference where the coordinator is checking node health, routing tasks, and shuttling API responses between machines, those milliseconds compound. Wired also eliminates the WiFi dropout that kills a query mid-stream and makes you re-run it.

The $26 for a switch and cables is the best-spent money in the entire build.

Tailscale for remote access

Tailscale connects your swarm to any device anywhere. Install it on all three nodes and your laptop. Now your laptop can reach the swarm from the couch, the coffee shop, or across town.

Tailscale's free plan includes:

- 100 devices (you'll use 4)
- 3 users
- Unlimited subnet routers
- No bandwidth limits on peer-to-peer traffic
- MagicDNS (auto hostnames like `thinkcentre.tail12345.ts.net`)
- End-to-end WireGuard encryption

We covered the full Tailscale + Ollama setup in From 178 Seconds to 19. The short version: install Tailscale on each node, set `OLLAMA_HOST=0.0.0.0` on the GPU machine, and your laptop becomes a thin client for the 3090's inference power. No port forwarding, no firewall rules, no dynamic DNS.

Software Stack

Every machine runs Linux. The M710Q runs Ubuntu Server 22.04. The 3090 workstation runs Ubuntu 24.04 with NVIDIA drivers. The Pi runs Raspberry Pi OS Lite. All three run Ollama.

Node	OS	Software	Models Loaded
3090 workstation	Ubuntu 24.04	Ollama, Tailscale	Qwen 2.5 32B (Q4_K_M), Llama 3.2 7B
M710Q	Ubuntu Server 22.04	Ollama, Tailscale	Llama 3.2 7B (CPU), nomic-embed-text
Pi 5	Raspberry Pi OS Lite	mycoSwarm daemon, Tailscale	None (coordinator only)

The mycoSwarm daemon on the Pi handles discovery and routing. It uses mDNS to find nodes on the local network, queries each node's Ollama API for loaded models and hardware capabilities, and maintains the capability map. The code is on GitHub and it's about 2,000 lines of Python.

You don't need mycoSwarm to use this hardware. You can manually point your laptop at whichever Ollama instance you want by setting the `OLLAMA_HOST` environment variable. mycoSwarm automates the routing so you don't have to think about it.

What This Build Actually Does

Here's a typical day for the swarm:

I open Open WebUI on my laptop. The interface connects to the M710Q's Ollama instance. I ask a simple question. The M710Q answers it from its 7B model in a few seconds. Cost: a fraction of a penny in electricity.

I paste in a technical document and ask for a structured analysis. The M710Q recognizes this is too complex for a 7B model and routes the query to the 3090. The 3090 loads Qwen 2.5 32B (already warm in VRAM) and produces a detailed breakdown. Twenty seconds. The response streams back through the M710Q to my browser.

I search my personal knowledge base using RAG. The M710Q generates the embedding with nomic-embed-text, retrieves relevant chunks from the vector store, and either answers from its 7B model or escalates to the 3090 depending on the complexity.

The Pi sits there watching. Its dashboard shows me which nodes are up, what models are loaded, how much VRAM is free on the 3090, and how many queries each node has handled today. If the M710Q goes offline (hasn't happened yet, but it will eventually), the Pi updates the capability map and routes everything to the 3090 until the M710Q comes back.

All of this happens on my local network. No API keys. No per-token billing. No rate limits. No terms of service. No outages because OpenAI had a bad Tuesday.

What I'd Change

If I were buying everything today instead of two months ago, I'd make two changes.

I'd get a Pi 5 1GB ($45) instead of the 2GB ($65). The coordinator role doesn't need 2GB. When I built this, the 1GB wasn't available yet (it launched in December 2025). That saves $20.

I'd also look harder for the RTX 3090 on Facebook Marketplace. eBay and r/hardwareswap charge $800-900 consistently. Facebook Marketplace has more variance. I've seen 3090s listed

for $700-750 in the Bay Area by people who just want the card gone. You have to be quick and willing to drive, but the savings are real.

With those two changes, the build drops to roughly $930-980. Under a thousand dollars for a distributed AI lab.

Scaling Up

This build is three nodes. But nothing about the architecture limits it to three.

Have an old gaming PC with a GTX 1070? Plug it in. It can handle 7B inference and take load off the M710Q. Found another M710Q on eBay for $85? Add it. More light-inference capacity, more redundancy. A friend has a Mac Mini M2 with 16GB unified memory? Add it to the Tailscale network and the swarm gains a node that can run 14B models at decent speed.

Each new node announces its capabilities via mDNS, the coordinator updates the map, and the routing layer starts using it. No configuration files to edit. No IP addresses to hardcode.

If you add…	Cost	What it brings
Second M710Q	$85	Redundancy, more embedding capacity
GTX 1070 desktop	~$100-150 (card)	8GB VRAM, handles 7B GPU inference
RTX 3060 12GB desktop	~$300 (card)	12GB VRAM, runs 13B models
Mac Mini M2 16GB	~$450 used	16GB unified, runs 14B models well
Second RTX 3090	$850	Double the heavy inference capacity

The total cost of the base build plus any one of those expansions stays under $1,500. Two of them and you’re still under $2,000, with a cluster that handles more concurrent users, more model variety, and more resilience than any single machine at the same price.

The Real Cost Comparison

The question people ask: “Why not just pay for a cloud API?”

Here’s the math. The 3090 generates roughly 160,000-400,000 tokens per hour on a 7B model, depending on quantization. Electricity cost: about 8 cents per hour.

	Cloud (GPT-4o output)	Cloud (Claude Sonnet)	Local (electricity)
1M tokens	$10.00	$15.00	~$0.02
10M tokens	$100.00	$150.00	~$0.20
100M tokens	$1,000.00	$1,500.00	~$2.00

The hardware costs $1,026 upfront. If you were spending $100/month on API calls, the hardware pays for itself in about 11 months including electricity. After that, every query is functionally free.

If you're spending $50/month, break-even takes longer. If you're spending $500/month, the hardware pays for itself before the second bill arrives.

The comparison isn't perfectly fair. GPT-4o and Claude Sonnet are better than a local 7B model at complex reasoning. But for the 70-80% of queries that don't need frontier-level intelligence (embeddings, simple Q&A, classification, drafts, RAG retrieval), a local 7B or 32B model does the job. Use cloud for the 20-30% of tasks that genuinely need it, and the savings on everything else still add up fast.

For a deeper cost analysis, see our full cost breakdown.

Parts List (Copy-Paste Shopping)

RTX 3090 ($800-900):

- Search eBay for RTX 3090 24GB sorted by price + shipping
- Check r/hardwareswap daily (set alerts for "3090")
- Facebook Marketplace for local pickup deals
- Avoid mining cards with damaged fans (they're fine for inference, but fan replacement is $20-40)
- See our full buying guide

ThinkCentre M710Q ($75-100):

- Search eBay for "ThinkCentre M710Q i5" sorted by price + shipping
- Look for i5-7500T / 8GB / 256GB SSD configurations
- Bulk liquidation sellers often have the best prices
- Skip i3 models (slower on CPU inference)
- Skip i7 models (costs $50+ more, minimal benefit for this role)

Raspberry Pi 5 ($45-65):

- Buy from official resellers (PiShop US, CanaKit, Adafruit)
- The 1GB ($45) or 2GB ($65) is enough for coordination
- Don't buy from Amazon scalpers charging $80+ for a $65 board
- You need a USB-C power supply (the official 27W one is $12) and a microSD card ($8-10)

TP-Link TL-SG105 ($16):

- Amazon, Best Buy, or any electronics store
- Any 5-port unmanaged gigabit switch works. This one is fanless and has a metal case

Cat6 Ethernet cables ($8-12 for a 3-pack):

- Amazon basics, Monoprice, or whatever's cheap
- 3-foot cables if everything's on one shelf, 10-foot if spread across a desk

---

📚 **Related:** Rescued Hardware, Rescued Bees · From 178 Seconds to 19 · mycoSwarm vs Exo vs Petals vs Nanobot · Best Used GPUs for Local AI

mycoSwarm: GitHub
