Fabric — model routing

local ↔ cloud · a capability goes in, the best ready model comes out

Caller — engine · Weave · Studio · executors

`router.run(capability, min_cap=…) — asks for a lane, never a model`

① fabric_router — capability facade & routing decision

ranks registry cards by the lane's PRIMARY axis · applies the min_cap floor · empty set → `pool_exhausted`

7 lanes: `fast` · `reason` · `code` · `review` · `vision` · `plan` · `embed` (each = a capability, not a model)

Policy split — private/low-latency lanes stay local; only reason & code may burst to cloud

reads

Model registry — data, not code

`fabric_models.json + .state overlay`

cards scored on 10 capability axes

`models_for_capability(lane, min_cap)`

Local lanes — on-device always

`fast · vision · embed` latency + privacy

Cloud-eligible lanes

`reason · code` if provider ready & cap declared

② fabric_providers — local backend pool

`select_backend(model)`

vLLM node(s) · GPU | Ollama node(s) · Mac

order: healthy → resident/warm → backpressure → weighted least-in-flight · hybrid never hides Ollama

② fabric_ai — cloud specialist adapter

`_chat_with_failover() · Anthropic Claude`

reason · code · vision | monthly USD budget

explicit + grounded context leaves the machine · budget-exhausted or unhealthy → degrade to local

Model plugins — open-weight local (lazy load + evict to fit RAM) · frontier cloud (policy-gated)

Fast	Reason	Coder	Vision	Embed 🔒	Claude — reason · code · vision
gemma3:12b	qwen2.5:72b	qwen3-coder-next	qwen2.5vl:32b	nomic-embed	remote · not downloaded

Benchmark routing — fabric_bench

measured latency + capability per model seeds cold routing; no learned signal yet → benchmark decides the lane pick

Learned routing — fabric_learn

validated + human-approved outcomes bias future picks; cloud successes become local routing teacher signals

learned bias feeds back into the router

Principle: the capability contract is the seam — swap any model and nothing above the router changes

Forge → Gateway → lane contract → provider → typed result · routing receipt → Event Spine

caller & learning | routing & lanes | local infra & models | cloud specialist | — dashed = feedback / failover