



Model concepts

Models CLI

See [/concepts/model-failover](#) for auth profile rotation, cooldowns, and how that interacts with fallbacks. Quick provider overview + examples: [/concepts/model-providers](#).

How model selection works

OpenClaw selects models in this order:

1. **Primary** model (`agents.defaults.model.primary` or `agents.defaults.model`).
2. **Fallbacks** in `agents.defaults.model.fallbacks` (in order).
3. **Provider auth failover** happens inside a provider before moving to the next model.

Related:

`agents.defaults.models` is the allowlist/catalog of models OpenClaw can use (plus aliases).

`agents.defaults.imageModel` is used **only when** the primary model can't accept images.

Per-agent defaults can override `agents.defaults.model` via `agents.list[].model` plus bindings (see [/concepts/multi-agent](#)).

Quick model picks (anecdotal)



GLM: a bit better for coding/tool calling.

MiniMax: better for writing and vibes.

>

Setup wizard (recommended)

If you don't want to hand-edit config, run the onboarding wizard:

```
openclaw onboard
```

It can set up model + auth for common providers, including **OpenAI Code (Codex) subscription** (OAuth) and **Anthropic** (API key recommended; `claude setup-token` also supported).

Config keys (overview)

```
agents.defaults.model.primary and agents.defaults.model.fallbacks  
agents.defaults.imageModel.primary and  
agents.defaults.imageModel.fallbacks  
agents.defaults.models (allowlist + aliases + provider params)  
models.providers (custom providers written into models.json )
```

Model refs are normalized to lowercase. Provider aliases like `z.ai/*` normalize to `zai/*`.

Provider configuration examples (including OpenCode Zen) live in

.

"Model is not allowed" (and why replies stop)

If `agents.defaults.models` is set, it becomes the **allowlist** for `/model` and for session overrides. When a user selects a model that isn't in that allowlist, OpenClaw returns:

 Model "provider/model" is not allowed. Use /model to list available models.

This happens **before** a normal reply is generated, so the message can feel like it “didn’t respond.” The fix is to either:

Add the model to `agents.defaults.models`, or

Clear the allowlist (remove `agents.defaults.models`), or

Pick a model from `/model list`.

Example allowlist config:

```
{  
  agent: {  
    model: { primary: "anthropic/clause-sonnet-4-5" },  
    models: {  
      "anthropic/clause-sonnet-4-5": { alias: "Sonnet" },  
      "anthropic/clause-opus-4-6": { alias: "Opus" },  
    },  
  },  
}
```

Switching models in chat (`/model`)

You can switch models for the current session without restarting:

```
/model  
/model list  
/model 3  
/model openai/gpt-5.2  
/model status
```

Notes:

 `/model` (and `/model list`) is a compact, numbered picker (model family + available providers).

`/model <#>` selects from that picker.
 >

`/model status` is the detailed view (auth candidates and, when configured, provider endpoint `baseUrl + api mode`).

Model refs are parsed by splitting on the `first /`. Use `provider/model` when typing `/model <ref>`.

If the model ID itself contains `/` (OpenRouter-style), you must include the provider prefix (example: `/model openrouter/moonshotai/kimi-k2`).

If you omit the provider, OpenClaw treats the input as an alias or a model for the `default provider` (only works when there is no `/` in the model ID).

Full command behavior/config: [Slash commands](#).

CLI commands

```
openclaw models list
openclaw models status
openclaw models set <provider/model>
openclaw models set-image <provider/model>

openclaw models aliases list
openclaw models aliases add <alias> <provider/model>
openclaw models aliases remove <alias>

openclaw models fallbacks list
openclaw models fallbacks add <provider/model>
openclaw models fallbacks remove <provider/model>
openclaw models fallbacks clear

openclaw models image-fallbacks list
openclaw models image-fallbacks add <provider/model>
openclaw models image-fallbacks remove <provider/model>
openclaw models image-fallbacks clear
```

`openclaw models` (no subcommand) is a shortcut for `models status`.

models list

Shows configured models by default. Useful flags:

- all : full catalog
- local : local providers only
- provider <name> : filter by provider
- plain : one model per line
- json : machine-readable output

models status

Shows the resolved primary model, fallbacks, image model, and an auth overview of configured providers. It also surfaces OAuth expiry status

for profiles found in the auth store (warns within 24h by default). --plain prints only the resolved primary model. OAuth status is always shown (and included in --json output). If a configured provider has no credentials, models status prints a **Missing auth** section. JSON includes auth.oauth (warn window + profiles) and auth.providers (effective auth per provider). Use --check for automation (exit 1 when missing/expired, 2 when expiring).

Preferred Anthropic auth is the Claude Code CLI setup-token (run anywhere; paste on the gateway host if needed):

```
claude setup-token  
openclaw models status
```

Scanning (OpenRouter free models)

openclaw models scan inspects OpenRouter's **free model catalog** and can optionally probe models for tool and image support.

Key flags:

```
--no-probe : skip live probes (metadata only)  
--min-params <b> : minimum parameter size (billions)  
--max-age-days <days> : skip older models  
--provider <name> : provider prefix filter  
--max-candidates <n> : fallback list size  
--set-default : set agents.defaults.model.primary to the first selection  
--set-image : set agents.defaults.imageModel.primary to the first image selection
```

Probing requires an OpenRouter API key (from auth profiles or OPENROUTER_API_KEY). Without a key, use --no-probe to list candidates

only.



Scan results are ranked by:

>

1. Image support
2. Tool latency
3. Context size
4. Parameter count

Input

```
OpenRouter /models list (filter :free )
```

Requires OpenRouter API key from auth profiles or OPENROUTER_API_KEY
(see [/environment](#))

Optional filters: --max-age-days , --min-params , --provider , --max-candidates

Probe controls: --timeout , --concurrency

When run in a TTY, you can select fallbacks interactively. In non-interactive mode, pass --yes to accept defaults.

Models registry (models.json)

Custom providers in models.providers are written into models.json under the agent directory (default ~/.openclaw/agents/<agentId>/models.json). This file is merged by default unless models.mode is set to replace .

[Model Provider Quickstart](#)

[Model Providers](#) >



Powered by **mintlify**

>
