

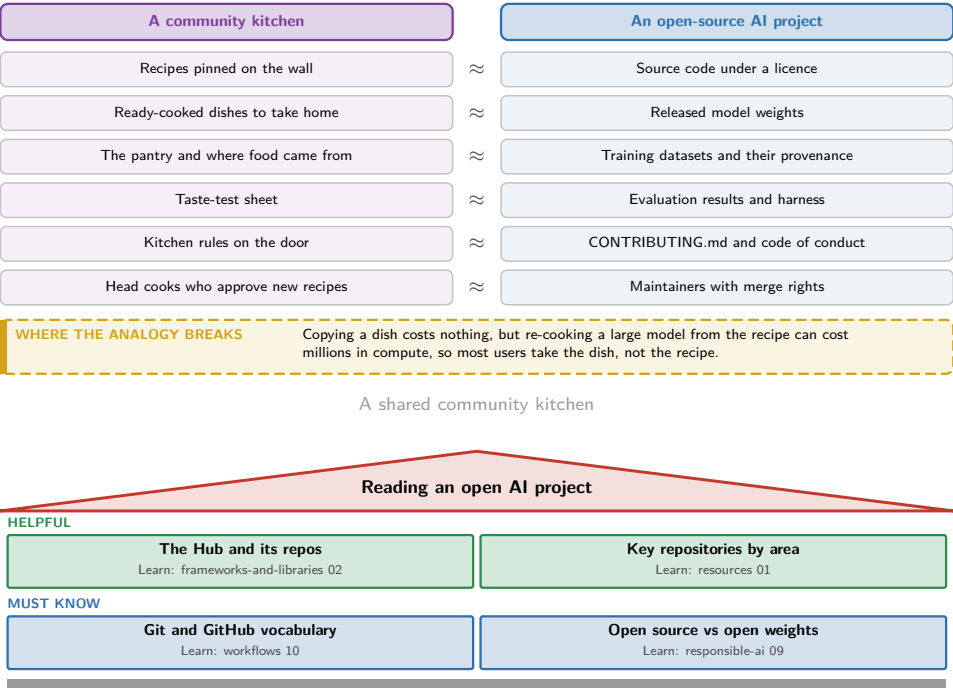


How Open-Source AI Projects Work

Open source in AI 01 | Kinds of open AI artefacts, governance models (BDFL, company-led, foundation, community), roles and how people gain them, repository anatomy, decision channels (issues, RFCs, meetings), funding and health signals; as of 2026

[#OpenSource](#) [#Governance](#) [#Maintainers](#) [#Foundations](#)

What is open in open-source AI



Prerequisites for this sheet

Artefact	Example (as of 2026)	Where it lives	What is special
Libraries, frame-works	PyTorch, transformers, vLLM, llama.cpp	GitHub	fast release cadence; GPU-specific CI is costly
Model weights	OLMo, Qwen, gpt-oss, Llama	Hugging Face Hub	files of GB to TB; licence may restrict use; cannot be reviewed line by line
Datasets	FineWeb, The Stack, Common Crawl	Hub, cloud buckets	rights of third parties inside the data; personal data
Evaluations, bench-marks	lm-evaluation-harness, Inspect	GitHub	results depend on prompt format and version; contamination

Specifications, for-mats	ONNX, safetensors, GGUF, MCP	GitHub, foundations	neutral governance matters more than code
Applications, agents	Open WebUI, coding agents	GitHub	fast-moving; security of tool use

Six kinds of open AI artefact and what makes each different from ordinary open source

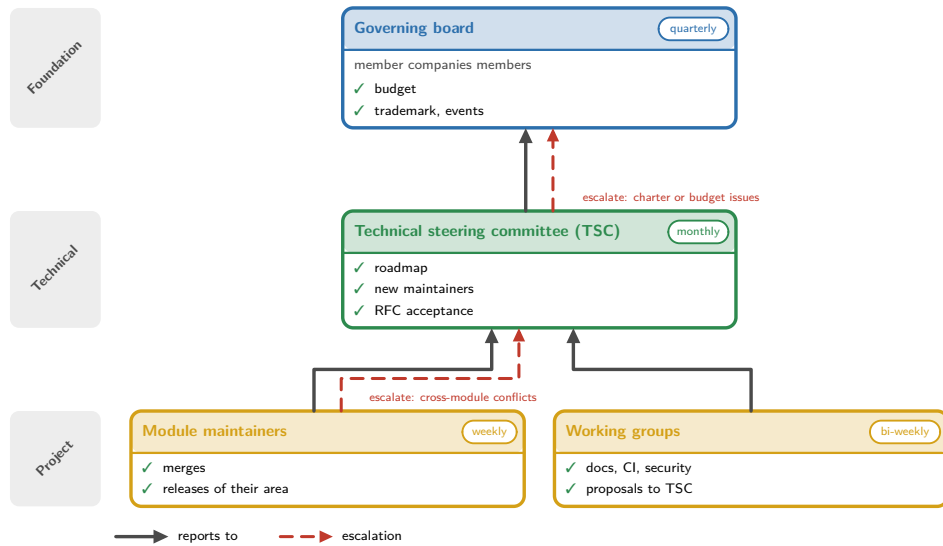
Recall in one line

Openness is a gradient from API-only to fully open (weights, code, data information); only the far end meets the OSI Open Source AI Definition 1.0 (taught in **responsible-ai 09**).

Governance: who decides

	BDFL / founder	Company-led	Recommended Foundation	Community council
Final say	one person	the company	TSC under a neutral charter	elected council or committee
How you gain rights	founder's trust	hired or invited	documented merit path	votes of core devs
Typical AI example	early llama.cpp, many solo repos	transformers (Hugging Face)	PyTorch, vLLM (PyTorch Foundation)	scikit-learn, Python
Main risk	bus factor 1	priorities follow the business	slower, process-heavy	consensus can stall

Four governance models seen in AI projects (many projects move from left to right as they grow)



A typical foundation-hosted AI project (simplified; each charter differs)

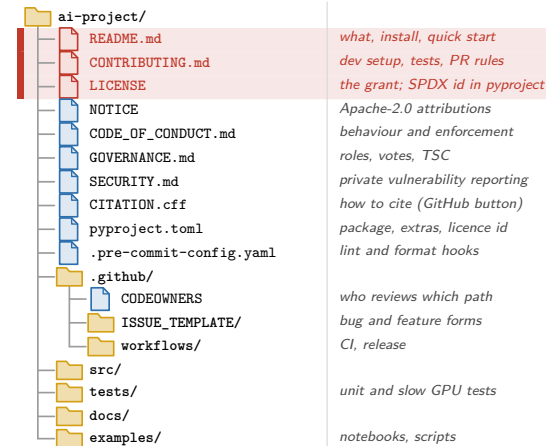
Role	Can do	How you get there
User	report bugs, ask questions, star	use the software
Contributor	open PRs, comment on issues and RFCs	first merged PR
Triager	label, close duplicates, reproduce bugs	invited after helpful issue work; GitHub 'Triage' permission
Reviewer / code owner	approve PRs in an area (CODEOWNERS)	sustained good PRs and reviews in that area
Maintainer / core dev	merge, release, vote	nomination and vote or lazy consensus of existing maintainers
Steering / TSC member	decide direction, resolve disputes	election or appointment under GOVERNANCE.md

Roles and how people move up the ladder

- Issue thread:** bug triage, small features; a maintainer label (accepted, wontfix) closes the question.
- Pull request review:** approval by code owners plus green CI; most projects require one or two approvals.
- Discussion or forum** (GitHub Discussions, Discourse, Discord): usage questions and early ideas, not decisions.
- RFC:** a design document in a dedicated repository or folder (e.g. `pytorch/rfcs`), commented for weeks, accepted by the TSC or maintainers.
- Community or TSC meeting:** recorded minutes; votes on maintainers, deprecations and roadmaps.

Where decisions are made, from small to large

Anatomy of an AI repository



Files to read before contributing (highlighted = read first)

pytorch / **pytorch**

Python (BSD-3-Clause)

Tensor library; PyTorch Foundation (Linux Foundation) since 2022

```
pip install torch
```

github.com/pytorch/pytorch

huggingface / **transformers**

Python (Apache-2.0)

Model definitions and pipelines; company-led

```
pip install transformers
```

github.com/huggingface/transformers

vllm-project / **vllm**

Python (Apache-2.0)

LLM serving engine; hosted by the PyTorch Foundation since 2025

```
pip install vllm
```

github.com/vllm-project/vllm

ggml-org / **llama.cpp**

C++ (MIT)

Local LLM inference, GGUF format; founder-led

```
brew install llama.cpp
```

github.com/ggml-org/llama.cpp

scikit-learn / **scikit-learn**

Python (BSD-3-Clause)

Classical ML; community governance, NumFOCUS fiscal sponsor

```
pip install scikit-learn
```

github.com/scikit-learn/scikit-learn

onnx / **onnx**

Python (Apache-2.0)

Open model format; LF AI and Data project

```
pip install onnx
```

github.com/onnx/onnx

Six AI projects with different governance (licences and owners as of 2026)

Money, health and trust		
Model	How it works	Watch for
Employed maintainers	companies pay staff to work upstream	roadmap tilts to the employer
Open core / hosted service	free core, paid cloud or enterprise features	features moved behind the payroll
Open weights as distribution	lab releases weights to build an ecosystem around its models	licence terms can change between versions
Foundation membership	member companies pay fees for neutral home and events	pay-to-play perception
Donations and sponsors	GitHub Sponsors, Open Collective (.github/FUNDING.yml)	rarely enough for a full salary
Public grants	e.g. Sovereign Tech Agency (Germany) for open infrastructure	time-limited

How open AI projects are funded

Health check before you depend on (or invest time in) a project

- ✓ baseline / mandatory ☐ recommended – verify in your context
- ✓ Licence file present and OSI-approved (or terms you accept)
- ✓ Release in the last few months; changelog maintained
- ✓ More than one active maintainer (bus factor above 1)
- ✓ SECURITY.md or private vulnerability reporting enabled
- ☐ **SHOULD** Issues get a first response within days, not months
- ☐ **SHOULD** Governance written down (GOVERNANCE.md, charter)
- ☐ **COULD** Contributors from more than one organisation CHAOSS 'elephant factor'
- ☐ **COULD** OpenSSF Scorecard run on the repository

★ Myths and recap

MYTH

✗ Open source projects are run by volunteers in their spare time

✗ Moving to a foundation changes the licence

✗ Stars measure project health

✗ Anyone who contributes gets a vote

FACT

✓ Large AI projects are mostly built by paid engineers
Volunteers remain vital for docs, triage and long-tail fixes.

✓ It changes ownership of trademark and governance, not the licence
PyTorch kept its BSD-style licence when it joined the Linux Foundation.

✓ Stars measure attention
Look at release cadence, response time, maintainer count, open security issues.

✓ Votes belong to the roles GOVERNANCE.md defines
Usually maintainers or core developers.

Remember

- 01

Artefacts differ
code, weights, data, evals, specs each bring their own rules
- 02

Read governance before code
who decides, how rights are earned
- 03

Decisions climb a ladder
issue → PR → RFC → TSC
- 04

Follow the money
employer, open core, foundation, grants shape the roadmap

- Q1

Name the four governance models of open AI projects.
A BDFL or founder-led, company-led, foundation-hosted (TSC), community council; projects often move toward foundation or council as they grow.
- Q2

What does a fiscal sponsor do?
A Holds and spends donations on behalf of a project (e.g. NumFOCUS for scikit-learn) without owning its technical governance.
- Q3

Which file says who must review changes to a path?
A .github/CODEOWNERS: GitHub requests reviews from the listed owners automatically.
- Q4

Where do large design changes get discussed?
A In an RFC: a public design document (often a dedicated repo such as pytorch/rfcs), accepted by maintainers or the TSC.
- Q5

What is the bus factor?
A How many people must leave before the project stalls; 1 means everything depends on one person.
- Q6

What is lazy consensus?
A A proposal passes if nobody objects before a stated deadline; common for routine decisions and new committers.
- Q7

Why are model weights harder to review than code?
A They are billions of numbers, not readable logic; trust comes from the card, evaluations, provenance and signatures.
- Q8

Which project hosts vLLM and DeepSpeed as of 2026?
A The PyTorch Foundation, part of the Linux Foundation (both joined in 2025).

Q9 What does CHAOSS provide?

A Definitions of community-health metrics (time to first response, contributor diversity, bus factor) for open-source projects.