Metric: AI Consent

Question

Has the community established explicit consent for AI use — including which AI models are approved — and how effectively is that consent enforced in practice?

NOTE: the output quality/value would be a separate metric.

Overview

AI is changing how we build software and how we collaborate, and it comes with as-yet-unknown impacts on communities where AI is being used to communicate and build. At the center of disputes around AI use in communities are two things: trust and consent.

Consent. People hold diverse values about AI, including concerns about environmental impact, bias in training data, how their contributions may be used, and obligations by maintainers to review work partially or wholly generated by AI. Disclosing AI use allows others to make informed choices about their participation.

This metric measures three linked things:

1. Whether a community has explicitly decided and documented its consent for AI use across the tools, platforms, and community spaces it depends on or owns.
2. If AI is permitted, which specific models or model classes the community has approved.
3. How effectively (1) and (2) are enforced in practice.

Taxonomy

Consent Metric - Established consent (location/binary)

- As a contributor, I want to know whether the community has established rules around AI and where I can find them
- As a contributor, I want to be able to easily find and navigate the applicable AI policies that may exist at multiple levels (i.e. project policy, community policy/CHAOSS, governing body policy/LF, state/national policy, etc)
- As a community architect, I want to know how many of the communities I oversee have implemented policies around AI

Consent Metric - Supervision level/ keypolicy features

- As a contributor, I want to know where a community's policies stand/what key features it has at a high level (i.e. no AI allowed, "human takes responsibility/is accountable", disclosure required, limited unsupervised use allowed, fully unsupervised use allowed, etc)
- As a user of open source, I want to be able to evaluate the potential risk of choosing one project as a dependency over another based on how well their policy position surrounding the unsupervised use of AI aligns with my own

Consent Metric - Actions/tool types allowed

- As a contributor, I want to know which tools/models im allowed to use in/introduce to a community (like a notetaker bot, etc)
- As a community architect/company who uses lots of open source, I'd like to be able to quickly determine which of the communities I depend on would allow a specific tool I want to use
- *User story*

Consent Metric - AI use allowed

- as a contributor I want to be able to quickly learn what ways a particular community allows AI use so I can know whether my contributions are a good fit for the community (vs a self-maintained fork)

User story “as a maintainer I worry about”

Data Collection Strategies

What we would look for to create baselines and track progress.

Documentation side:

- Repository scan for AI policy files (AI-DECLARATION.md or similar emerging standard); dedicated sections in CONTRIBUTING.md, GOVERNANCE.md, CODE_OF_CONDUCT.md
- Describes enforcement accountable role and mechanisms
- Inventory of detection mechanisms in place (bot allow-lists, AI-disclosure labels on PRs, commit signing requirements, periodic audits)
- Enforcement role (owner)

Covers what types of usage are allowed:

- Pull Requests (code, documentation, for issue-type such as security, bugs, etc)
- Issues (opening, commenting, closing)
- Content (documentation, blogs, social media)
- Moderation (flagging, tagging, hiding, deleting)
- Vulnerability scanning (opening CVEs)
- Use of AI in community communication channels (irc, forums, mailing lists, Discord, Slack)

Approved models documented:

- Whether an approved-model list exists (allow-list, deny-list, or criteria-based)
- Approval criteria documented (e.g., open weights, training data disclosure, open data, license compatibility, environmental disclosure, attribution behavior)
- Named models or named providers approved/disallowed
- Review cadence for re-evaluating the list

Enforcement side:

Evidence of enforcement:

- Blocked PRs / deleted / hidden comments per documentation standard

- Log of enforcement events: warnings issued, contributions reverted or rejected for policy violation, policy clarifications triggered by incidents
- Community member experience reporting/flagging violations
- Time-to-resolution for reported AI policy violations, comparable to incident response metrics

Filters

- Project size (number of active contributors)
- Project age
- Technology Ecosystem (Python, Rust etc)
- Usage
- Community size
- Governance accountable (single maintainer, foundation, company etc)
- Funding status
- Whether an approved-model list exists
- Approved-model openness (open weights / closed / mixed / unspecified)
- Platforms for engagement (i.e., where consent can be received, given, and acknowledged)
    - Examples: Forums, chat rooms, mailing lists, etc.

Tooling & Resources

Available tools / signals

- Repository scan for AI policy files (AI-DECLARATION.md or similar emerging standard); dedicated sections in CONTRIBUTING.md, GOVERNANCE.md
- GraphQL (comment metadata like hidden comments) — for enforcement evidence such as hidden/deleted comments

Visualizations

- Two-axis matrix: documentation completeness × enforcement maturity, plotting communities into quadrants (e.g., "documented but unenforced," "instrumented and enforced")
- Timeline of enforcement events against policy adoption date

- Approved-model coverage matrix (community × approval criterion)

(Cite tool/website and date created/retrieved. Citation should be below the image as Figure X: Title (Source, Year).)

References

Framing

- Irwin, E. (2026). After the Pull Request: Four Proposed Areas of Work for Open Community Representation. https://sunnydeveloper.com/
- Irwin, E. AI Consent for Open Communities. https://sunnydeveloper.com/ai-consent-for-open-communities/
- Creative Commons (2026). Update on CC Signals: What Changed and Why. https://creativecommons.org/2026/04/23/update-on-cc-signals-what-changed-and-why/
- CHAOSS AI Alignment Working Group, Issue #61. https://github.com/chaoss/wg-ai-alignment/issues/61

Related efforts in adjacent spaces (mapped in WG Issue #61)

Existing efforts that touch the same governance categories this metric covers. The cluster below is selected for relevance to Approval / Type specifically.

On which specific models are acceptable (Approval / Type):

- Model Openness Framework — supply-side standard for what makes a model open across four classes. https://lfaidata.foundation/blog/2024/04/17/introducing-the-model-openness-framework-promoting-completeness-and-openness-for-reproducibility-transparency-and-usability-in-ai/
- OSI's Open Source AI Definition — supply-side standard for what qualifies as "open source AI". https://opensource.org/ai
- MLCommons — formal benchmarks and ISO standards work. https://mlcommons.org/2025/11/iso-aus/

Cross-cutting (touches Use, Approval, Actions, and Improvements):

- Metagov's Collective Governance for AI — framework mapping where community governance could sit across the AI stack. https://metagov.org/cg-ai/

Gap this metric addresses: existing approval-side efforts define what openness looks like from the lab side. None define what a specific community requires from a model to work with it in good faith. This metric is the demand-side counterpart — it captures whether a community has articulated, documented, and enforces its own answer.

Examples of documented project AI use policies (the "consent artifact" being measured)

- Python (Devguide). https://devguide.python.org/getting-started/generative-ai/
- OpenSSF. https://github.com/ossf/tac/pull/605
- Zulip CONTRIBUTING. https://github.com/zulip/zulip/blob/main/CONTRIBUTING.md#ai-use-policy-and-guidelines
- Ladybird Browser. https://github.com/LadybirdBrowser/ladybird/blob/master/CONTRIBUTING.md#on-usage-of-ai-and-llms
- Servo. https://book.servo.org/contributing.html#ai-contributions
- Gentoo Council AI policy. https://wiki.gentoo.org/wiki/Project:Council/AI_policy
- FastAPI. https://fastapi.tiangolo.com/contributing/#automated-code-and-ai
- Ghostty AI_POLICY.md. https://github.com/ghostty-org/ghostty/blob/main/AI_POLICY.md
- Linux Kernel discussion (kernel summit thread). https://lore.kernel.org/ksummit/20251114183528.1239900-1-dave.hansen@linux.intel.com/
- Castle Game Engine: Rules for contributing and using AI. https://castle-engine.io/ai
- Contributor Covenant, Notes for Translators. https://github.com/EthicalSource/contributor_covenant/blob/release/CONTRIBUTING.md#translators-and-native-speakers

Foundation- and organization-level policies (consent and model-approval above the project layer)

- Linux Foundation generative AI policy. https://www.linuxfoundation.org/legal/generative-ai
- OpenInfra Foundation AI policy. https://openinfra.org/legal/ai-policy

- Oxide RFD 0576. https://rfd.shared.oxide.computer/rfd/0576

Communities deliberating on consent (the process of establishing it)

- Fedora AI policy WIP discussion. https://discussion.fedoraproject.org/t/ai-policy-in-fedora-wip/144297
- SciPy: A policy on generative-AI-assisted contributions. https://discuss.scientific-python.org/t/a-policy-on-generative-ai-assisted-contributions/1702
- Ansible AI policy forum thread. https://forum.ansible.com/t/ansible-ai-policy/42642
- CHAOSS Augur AI policy issue. https://github.com/chaoss/augur/issues/3371
- pip-tools discussion. https://github.com/jazzband/pip-tools/discussions/2278
- Servo policy discussion. https://github.com/servo/servo/discussions/36379

Contributors

- Emma Irwin
- Adrian Edwards
- Coraline Ada Ehmke
- Justin Wheeler