From Open Access to Open Contribution: Democratising Academic Knowledge Production

A working proposal for Petrichor

Working draft — 15 November 2026

Abstract

Open access broadens the readership of research, but the opportunity to alter a research project remains a separate institutional question. Petrichor is a working proposal for an author-led environment in which a manuscript, its computational sources, references and editorial decisions form one versioned project. Contributors can propose substantive changes to the work and to its governing rules. A shared dashboard connects these proposals to executable analysis, generated outputs and recorded decisions. The initial arrangement combines a thirty-day contribution window with seven days for lead-author integration, while retaining a distinction between accepted contributions and agreed authorship. This paper describes the current implementation and identifies design questions for amendment. It reports no evaluation of whether the arrangement improves research quality, participation or fairness. Those outcomes require investigation rather than inference from software functionality.

1 The institutional problem

The Budapest Open Access Initiative established an influential account of removing barriers to reading and reusing scholarly literature (Budapest Open Access Initiative, 2002). Access to an existing argument does not by itself confer a recognised route into changing that argument. A reader may identify an omitted source, repair a script, challenge an interpretation or contribute a stronger method without becoming part of the original research team. The question addressed here is how such interventions could enter an accountable editorial process.

The UNESCO Recommendation on Open Science places scientific knowledge alongside infrastructures, engagement and dialogue within a broader account of openness (UNESCO, 2021). Petrichor treats participation in the production and revision of research as a design problem within that wider agenda. Its name does not designate an established scholarly institution, and its current implementation cannot demonstrate democratisation. Instead, this manuscript offers a concrete arrangement whose technical operations and allocation of decision rights can themselves be criticised and amended.

2 A shared, inspectable research project

A project dashboard brings manuscript source, computational files, bibliography, amendments, governance and settings into one working environment. LaTeX and Code & data views operate on the same project files. BibTeX entries can be inspected and citations inserted into the manuscript. Saving creates a source version rather than separating the written argument from the analysis that supports it.

Manuscript comments can reference a particular script and line, for example `% @code analysis.R:12`. Following that reference opens the matching code. Such a link assists inspection; it does not

prove that the cited calculation supports the surrounding claim. Contributors still need to evaluate assumptions, data construction and interpretation. The link can also become misleading when a script is reorganised, making maintenance of the references part of substantive revision.

R and Python run in disposable, fixed-package environments. The current implementation restricts network access and applies limits of thirty seconds and 512 MB per run. It records executed source, a source hash, the runtime image identifier, diagnostic output and generated files. These records describe the computational event within the supplied environment. They do not independently establish correctness, reproducibility on another system or the legitimacy of using the input data.

Generated figures and LaTeX tables can be linked to their producing script. A successful rerun refreshes the linked project files and queues manuscript compilation. The linkage is explicit and versioned, rather than a silent replacement of an unrelated upload. A failed script keeps the previous manuscript. Missing outputs and concurrent project changes prevent automatic replacement and require attention. Researchers must still check that captions, numerical statements and conclusions remain consistent with the new results.

3 Identity and the contribution window

People register personal accounts, while each project chooses its review visibility. Double-blind review hides author and reviewer bylines; single-blind review names authors while withholding reviewer bylines. Non-blind review makes identities visible to project participants, and open review makes named discussion and review publicly readable. Authentication and accountability are therefore distinct from the public presentation of the research team. Operators necessarily retain account and audit information; a hidden byline is not anonymity from the service itself.

A change in review mode must not retroactively expose material submitted privately. Contributions, reviews and messages retain the visibility policy under which they were created, and a later project policy cannot widen that permission. The interface identifies the current audience before a person posts. An author who also serves as an editor may still be inferable from their role; source text and self-citations can likewise reveal identity.

Nor can interface controls guarantee anonymity within scholarly material. A self-citation, dataset description, writing style, document metadata or comment may identify its author. Participants should inspect source files and the compiled PDF before opening a project or submitting an amendment. The same caution applies to reference libraries and generated outputs. Suppressing a byline does not remove these other signals.

The default contribution window lasts thirty days. A subsequent seven-day integration period allows the lead author to resolve submitted work, request permitted revisions and assemble a coherent manuscript. Dates are visible in the dashboard. New contributions close at the published deadline; integration work must fit within its own period or an approved extension. These durations are starting rules, not evidence-based estimates of how long every discipline requires. Their adequacy, including implications for caring responsibilities and uneven access to computing resources, is an appropriate subject for amendment.

4 Project discussion and specific review

Each project has a discussion space alongside its source and governance panels. Discussion can address research questions, coordination and unresolved design choices without being mistaken for a formal decision on a particular amendment. Amendment-specific comments remain attached to the proposed change and its revision history. Both spaces follow the project's visibility policy, while earlier private messages retain their original access restrictions.

5 Amendments, recognition and responsibility

An amendment consists of proposed source changes and an explanation of their intellectual purpose. The contribution editor supports edits to existing text files; arbitrary binary replacement through anonymous review is outside this contribution pathway. The lead can inspect a comparison, run proposed scripts, request revision, accept or reject, and record reasons. Acceptance creates another manuscript source version and queues compilation. Previous source snapshots and decisions remain available in the project history.

This procedure makes a contribution contestable and traceable, but does not make its evaluation impartial. A lead author can misunderstand a challenge or favour changes that preserve the original argument. A recorded justification permits scrutiny of that choice; it does not substitute for an independent decision maker. Contributors should therefore be able to question both the reasoning offered and the criteria under which it was made.

Acceptance and authorship are distinct decisions. A correction may warrant acknowledgement; a substantive contribution may lead to an invitation to authorship. Coauthors must accept responsibilities and approve the final manuscript and author order. CRediT offers a vocabulary for describing different kinds of contribution (National Information Standards Organization, 2022); it does not, by itself, determine who qualifies as an author. The platform records roles and eligibility reasons without claiming to implement a complete taxonomy-based assessment system.

This distinction also matters for computational assistance. COPE's position on AI tools preserves human responsibility for manuscript content and excludes tools from authorship (COPE Council, 2024). A successful run, an AI-assisted edit or an accepted suggestion cannot automatically allocate scholarly credit. Participants remain responsible for checking references, documenting assistance where appropriate and agreeing the contribution represented by a final byline.

6 Amending the rules of the current project

Governance is part of the contribution surface. Contributors can propose changes to the revision limit, authorship criteria, conflict rules and project deadlines. A proposal states its reason and the rule revision against which it was prepared. The lead records an approval or rejection. An accepted proposal applies to the current project, preserving the previous rules, the proposed replacement and the decision in the history.

Concurrent proposals cannot silently overwrite one another. A proposal based on an outdated rule revision must be reconsidered against the current arrangement. Changing a deadline requires a prospective date, and final author approvals are cleared when accepted governance changes alter the conditions of approval. These protections make rule changes visible and

procedurally constrained. They do not amount to community voting: democratic panels and distributed decision authority are questions for later design and evaluation.

Reports and appeals have a separate intake route. The current implementation does not provide an appointed independent institutional panel or automated adjudication. Independent adjudication requires an identified reviewer without a relevant conflict, an opportunity for affected parties to respond, a reasoned outcome and a further route of challenge. The integrity of that process depends on an identified, accountable reviewer.

7 What this proposal asks contributors to change

Transparency and openness policies require attention to incentives and research culture, not merely file availability (Nosek et al., 2015). Petrichor's central proposition is that an inspectable project and explicit amendment process may create a useful route from reading research to improving it. The software is a vehicle for examining that proposition, not a test of its effects.

Several questions should guide contributions to this manuscript. How can editorial authority remain accountable when the lead controls acceptance? Which contributions deserve authorship, acknowledgement or another durable form of recognition? Who carries the workload of verifying executable changes, especially when an inexpensive submission imposes substantial review costs? How should accessibility, language, disciplinary methods and low-resource participation shape the interface and timetable? What privacy and data-governance arrangements are required before sensitive material is hosted? Which decisions should remain with authors, and which require independent or collective oversight?

An evaluation would need evidence about participation, decision quality, review effort, contributor experiences and the distribution of recognition. It should examine exclusion and failure as well as successful integrations, with appropriate consent and research governance. None of those empirical outcomes is reported here.

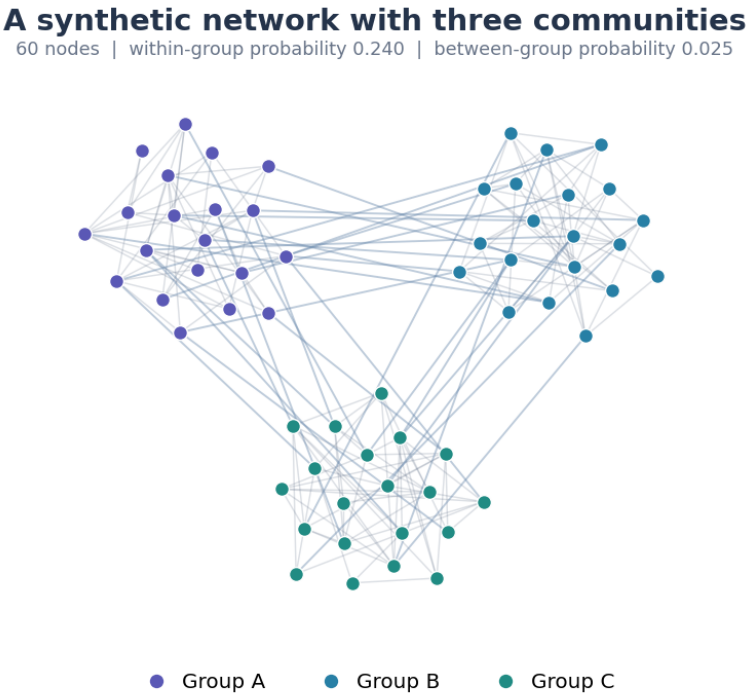
Finalisation requires the relevant authors' explicit approvals. Export bundles the compiled PDF and source files so that the project can proceed beyond the dashboard. Continued usefulness also depends on durable preservation, suitable reuse permissions and workable institutional stewardship. This paper remains a proposal for those arrangements: contributors are invited to amend its argument, its implementation assumptions and the rules by which it develops.

Synthetic network example

A stochastic block model generates 60 synthetic nodes in three preset groups; colour and position mark group membership. Within-group and between-group edge probabilities are set in the linked scripts. Python and R use the same specification and seed, but independent random generators produce different networks. These examples demonstrate reproducible outputs, not empirical evidence for the proposal.

Python implementation

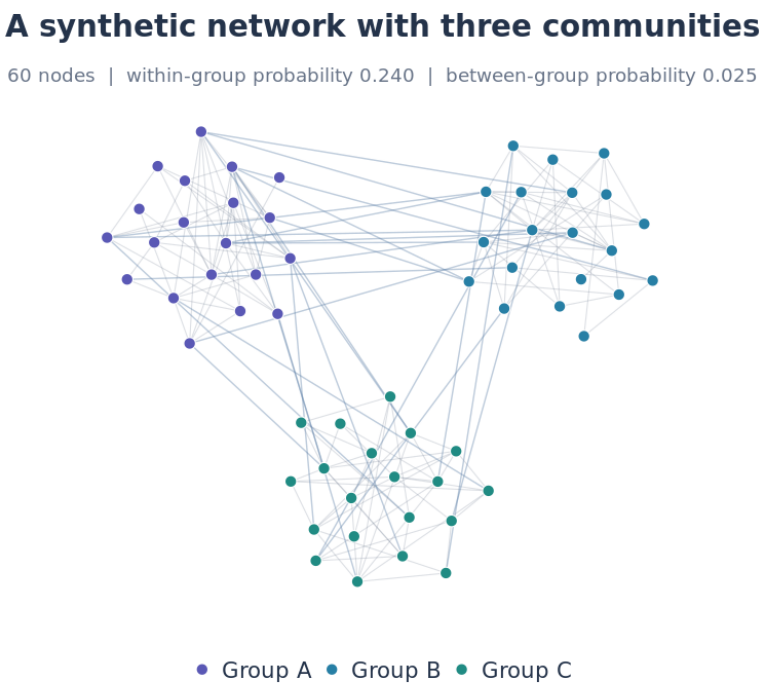
Within-group probability Between-group probability



Metric	Synthetic network
Nodes	60
Edges	167
Density	0.094
Mean degree	5.57
Cross-group edge share	0.192

R implementation

Within-group probability Between-group probability



Metric	Synthetic network
Nodes	60
Edges	160
Density	0.090
Mean degree	5.33
Cross-group edge share	0.169

Density is the share of possible undirected edges present. Mean degree is twice the edge count divided by the node count. Cross-group edge share is the proportion of realised edges joining different groups (defined here as zero if no edges are realised). Change either linked probability and rerun the corresponding script to refresh its figure, table and compiled manuscript.

References

Budapest Open Access Initiative. Budapest open access initiative, February 2002. URL https://www.budapestopenaccessinitiative.org/read/. Declaration released 14 February 2002. Accessed 2026-11-15.

COPE Council. Authorship and AI tools. Committee on Publication Ethics, 2024. URL https://doi.org/10.24318/cCVRZBms. COPE position statement. Accessed 2026-11-15.

National Information Standards Organization. ANSI/NISO Z39.104-2022: CRediT, contributor roles taxonomy, February 2022. URL https://www.niso.org/publications/z39104-2022-credit. Published 8 February 2022. Accessed 2026-11-15.

B. A. Nosek, G. Alter, G. C. Banks, et al. Promoting an open research culture. *Science*, 348(6242): 1422–1425, 2015. doi: 10.1126/science.aab2374. URL https://pubmed.ncbi.nlm.nih.gov/26113702/.

UNESCO. Recommendation on open science, November 2021. URL https://www.unesco.org/en/legal-affairs/recommendation-open-science. Adopted 23 November 2021 at the 41st General Conference. Accessed 2026-11-15.