Hollowing the Collegial: Six Displacements of Collegial Attention in Software Engineering Work

Varad Vishwarupe[1,2,3,*], **Alan F. Blackwell**[4], **Nigel Shadbolt**[1,2], **Marina Jirotka**[1,3], **Ivan Flechais**[1]

[1]**Department of Computer Science, University of Oxford**
[2]**Institute for Ethics in AI, University of Oxford**
[3]**Responsible Technology Institute, University of Oxford**
[4]**Department of Computer Science and Technology, University of Cambridge**

[*]Corresponding author: varad.vishwarupe@cs.ox.ac.uk

Software teams coordinate through mundane but consequential exchanges: a question asked across a desk, a design rationale worked through aloud, or a judgement made visible in review. Interviews with 33 UK software professionals suggest that LLM use can move some of these exchanges into private worker–model conversations. We examined passages describing changes in consultation, witnessing, explanation, contribution and team learning using qualitative content analysis informed by constant comparison. The analysis yielded six related displacements: witness withdrawal, consultation rerouting, articulation atrophy, reciprocity collapse, attribution obscuration and collective deskilling. Participant accounts show how these shifts appear in practice and connect them to CSCW work on common ground, situated action, boundary objects, practical wisdom and attention. We offer the taxonomy as a sensitising lens rather than a measurement instrument, and close with design directions for keeping model-assisted reasoning inspectable and available to colleagues.

Large language models; LLMs; human–AI interaction; software engineering; CSCW; team work; qualitative content analysis; grounded theory; digital bridges

1. INTRODUCTION

Software teams do not coordinate through artefacts alone. Coordination also happens in the small exchanges around them: a developer asks why a design should change, a researcher tries out a claim before presenting it, two engineers debug aloud, or a reviewer questions the judgement behind a patch. CSCW has long treated these interactions as part of the work itself because they establish shared understanding, expose local reasoning and make decisions open to challenge (Suchman 1987; Clark and Brennan 1991; Star and Griesemer 1989). With the arrival of large language models (LLMs) in software workflows, these interactions have begun to shift.

LLMs increasingly take on roles that previously brought colleagues into the process. A worker may now test a draft, ask a technical question or rehearse a rationale with a model before anyone else sees it. The immediate benefits are easy to describe: faster iteration, fewer interruptions and readily available suggestions (Sarkar and Drosos 2025). The team-level consequences are less visible. When the interaction moves to a model, colleagues lose not only the interruption but also an opportunity to notice uncertainty, correct a misunderstanding, explain local practice or learn how a decision was reached. Figure 1 contrasts a team-visible collegial episode with a model-rerouted one, showing how the output can remain shared even when the reasoning that formed it becomes private.

This short paper names that loss as *the hollowing of the collegial*. It asks:

RQ1. What acts of collegial attention are rerouted from colleagues to LLMs in day-to-day software engineering work?

RQ2. What analytic vocabulary can name these acts in a form that dialogues with established CSCW frames?

© Vishwarupe et al. Published by BCS Learning and Development Ltd.

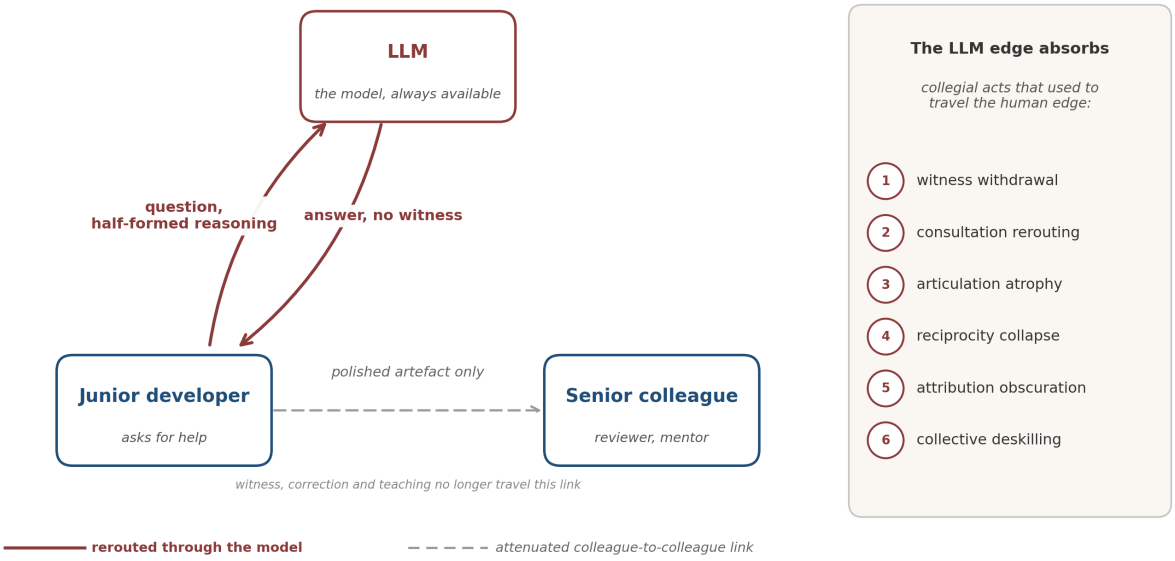

Figure 1: *The rerouting of a collegial episode. When an LLM sits between two colleagues, the junior's question, half-formed reasoning and the model's answer travel the LLM edge (solid, dark). The colleague-to-colleague link (dashed, grey) is thinned and now carries only the polished artefact. The right panel names the six collegial acts the LLM edge absorbs, elaborated in Section 4.*

RQ3. What design orientations follow for LLM-supported software tools that seek to preserve rather than absorb team-scale reasoning?

We develop a compact taxonomy of six displacements: witness withdrawal, consultation rerouting, articulation atrophy, reciprocity collapse, attribution obscuration and collective deskilling. Each is grounded in a participant quotation and interpreted through a well-established CSCW frame. As a prequel paper in the sense described by the BritCHI Short Papers track, its aim is not to close the empirical question but to open it in a form that later work can measure. The BritCHI theme of Digital Bridges is at stake: if workplace LLMs are to be digital bridges, they must bridge people back to one another rather than replace the bridges that already exist.

2. SITUATING THE ARGUMENT IN CSCW

CSCW has spent four decades showing that collaborative work is held together by shared understanding, situated practice and the artefacts through which teams coordinate. Suchman's account of situated action established that plans and formal instructions cannot capture the improvisational character of practical work; competent action is enacted through moment-to-moment sensitivity to the local situation (Suchman 1987, 2007). Clark and Brennan showed that conversation is a joint activity in which participants continually establish common ground (Clark and Brennan 1991). Star and Griesemer's boundary objects framework demonstrated how shared artefacts allow diverse actors to work together without full agreement, provided that the history of their use remains inspectable (Star and Griesemer 1989). These frames remain as central to software engineering teams now as they were to the settings in which they were first developed.

Recent HCI work on trust in automation and human–AI teaming has extended these traditions to intelligent systems (Lee and See 2004; Hoff and Bashir 2015; Amershi et al. 2019; Shneiderman 2020; Bansal et al. 2021). Empirical work on LLMs in software development has begun to describe conversational programming and the accompanying shifts in how developers author, inspect and debug code (Sarkar and Drosos 2025). Cross-cultural HCI has warned against reading narrow samples as universal (Henrich et al. 2010; Cuadra et al. 2024; Ashktorab et al. 2019). Existing work explains how people coordinate and how developers work with LLMs, but says less about the relation between these two processes. Our concern is not simply whether an individual can complete a task with a model. It is whether moving part of the interaction changes what colleagues can observe, teach, question and later reconstruct.

Adjacent conceptual work provides useful vocabulary. Blackwell treats attention as a moral resource of the social domain, at risk of extraction by conversational systems (Blackwell 2024). Frey and Weiss argue that humanlike dialogue agents produce a distinctive class of one-person dialogue that differs from both

human–human conversation and earlier human–machine interaction (Frey and Weiss 2025). Vallor argues that AI delegation can erode practical wisdom by removing the occasions on which judgement is exercised (Vallor 2016, 2024). The present paper takes these individual-scale accounts and asks a team-scale question, using the six-pattern taxonomy in the next section as an analytic lens. The present paper builds directly on our earlier study of how designers and developers negotiate the tool-versus-teammate framing of LLMs (Vishwarupe et al. 2026a), and complements adjacent work on the human–AI collaboration gap (Vishwarupe et al. 2026b).

3. METHOD

3.1. Corpus and setting

We draw on 33 semi-structured interviews with UK-based software professionals, a subset of a larger corpus of 64 conducted between late 2023 and early 2025. Participants included software developers, applied AI engineers, UX researchers, product practitioners and technical leads working on or around LLM-enabled software systems in three large multinational technology firms (hereafter MNCs) and three UK-based startups. Interviews were 45 to 90 minutes and covered recent LLM use, adoption decisions, oversight practices, accountability concerns and observed changes in team workflow. Data were collected under institutional ethics approval (reference on file with the corresponding author), transcribed, anonymised and stored under the approved data-management plan. Participants are referred to by anonymised identifiers (P01 to P33) with role indicated where relevant. The UK-only focus in this paper permits domain-consistent reporting of software engineering in a single national setting; the two-country comparison in the parent corpus is treated at length elsewhere.

The instrument used a common opening (“Walk me through a recent piece of work in which you used an LLM”) and probes on who else would have been involved (e.g., “Who else would have seen this if you had not asked the model?”), on review and oversight, on accountability when model-supported work went wrong, and on what juniors on the team no longer saw happening around them. Probes followed responses rather than a fixed script; the full guide is available on request.

3.2. Analytic procedure

We used qualitative content analysis with a constant-comparison sensibility (Elo and Kyngäs 2008; Charmaz 2014). We began by identifying passages describing a change in who was consulted, who witnessed reasoning, who owned the resulting judgement, or how team learning occurred around LLM use. These passages were coded line by line, using open codes such as *senior no longer asked*, *review becomes private*, *draft accepted before explanation*, *hard to credit contribution*, and *confidence without shared witness*. Focused coding grouped these into six recurring displacements. We then tested each candidate pattern against negative cases in which LLM use preserved rather than displaced collegial attention, refining pattern boundaries until further coding produced no new distinctions.

Two authors coded independently and reconciled through discussion. Inter-coder discussion focused on boundary cases where a single passage instantiated more than one displacement. The taxonomy is analytic rather than mutually exclusive: a single episode of LLM use will typically instantiate several patterns at once.

4. SIX DISPLACEMENTS

Figure 2 names the six displacements. Table 1 summarises the collegial act each rerouted, the workflow trace through which it becomes observable and the CSCW frame it dialogues with. This section illustrates each displacement with a representative participant quotation.

4.1. Witness withdrawal

Several participants used LLMs as first-pass reviewers, draft editors or rubber ducks. This let them refine work before involving anyone else. It also meant that colleagues encountered the output after uncertainty, hesitation and revision had been filtered out.

> “Once it feels like it is collaborating with me, it becomes unclear who is responsible for the output. My reviewer sees a clean version. They do not see the three times I asked the model to reconsider.” (P14, senior engineer)

Review becomes inspection of a product rather than participation in the formation of judgement. In Clark and Brennan’s terms, common ground is no longer established; it is presumed (Clark and Brennan 1991). This also connects to well-documented failure modes in the mechanisms that produce consensus-seeking systems (Kuklani et al. 2025).

4.2. Consultation rerouting

Participants often described the shift as convenience: opening a tab was quicker than finding a colleague. Yet the exchange that disappeared had done more than deliver an answer. Asking a senior placed the worker in a relation of apprenticeship, and gave the senior a chance to make local practice explicit.

> “I used to walk over. Now I open a tab. The answer is usually fine, but nobody knows I was stuck, so nobody knows what to teach next week.” (P22, applied AI engineer)

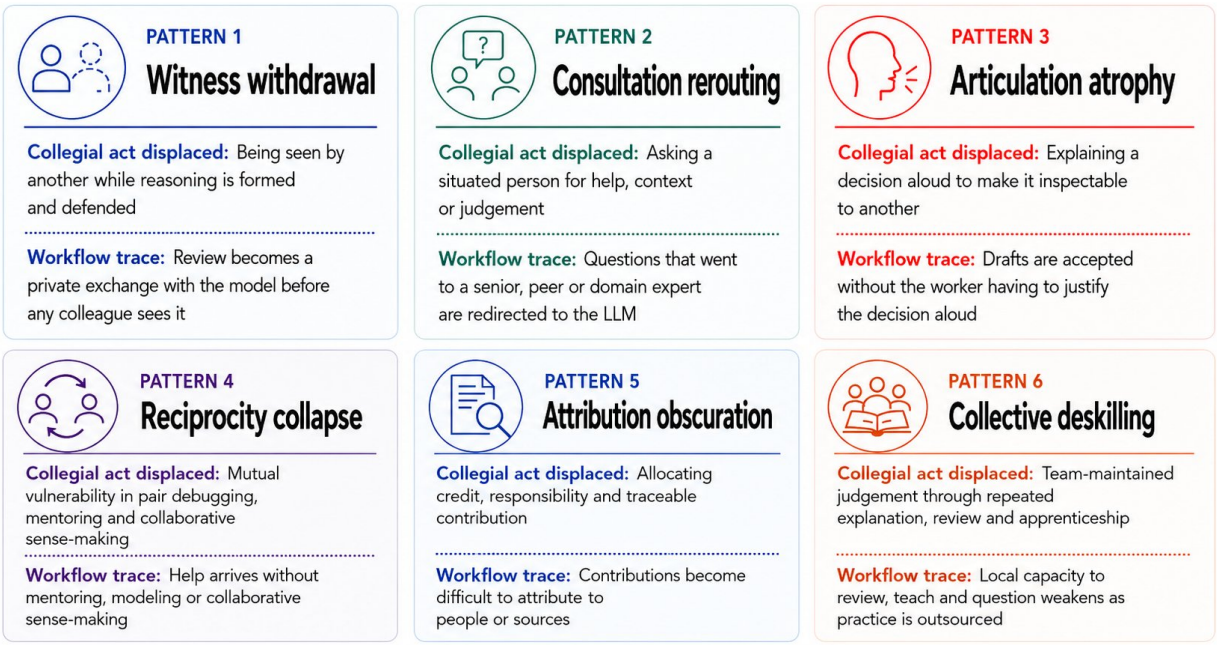

Figure 2: *The six-pattern taxonomy. Each panel names a collegial act that is displaced when an LLM enters a software workflow, paired with the workflow trace through which it becomes observable in interview data. The patterns are analytic distinctions rather than mutually exclusive classes; participants frequently described several in the same passage.*

Displacement	Collegial act displaced	Workflow trace	CSCW frame in dialogue
Witness withdrawal	Being seen by another while judgement is formed and defended	Review becomes a private exchange with the model before any colleague sees it	Common ground (Clark and Brennan 1991)
Consultation rerouting	Asking a situated person for help, context or judgement	Questions that went to a senior, peer or domain expert are redirected to the LLM	Situated action (Suchman 1987, 2007)
Articulation atrophy	Explaining a decision aloud to make it inspectable to another	Drafts accepted without the worker having to justify the decision aloud	Practical wisdom (Vallor 2016, 2024)
Reciprocity collapse	Mutual exposure in pair debugging, mentoring and collaborative sense-making	Help arrives without mentoring, modelling or collaborative sense-making	One-person dialogue (Frey and Weiss 2025)
Attribution obscuration	Allocating credit, responsibility and traceable contribution	Contributions become difficult to attribute to people or sources	Boundary objects (Star and Griesemer 1989)
Collective deskilling	Team-maintained judgement through repeated explanation, review and apprenticeship	Local capacity to review, teach and question weakens as practice is outsourced	Attention as social resource (Blackwell 2024)

Table 1: *Six displacements of collegial attention in LLM-mediated software engineering work. The rightmost column links each displacement to a CSCW frame the paper places it in dialogue with. Patterns are analytic distinctions, not mutually exclusive classes; a single episode typically instantiates several.*

The lost occasion is not the answer but the teaching moment that a similar consultation used to carry. This is a thinning of the informal knowledge exchange that keeps local knowledge in circulation.

4.3. Articulation atrophy

Explaining a choice to another person forces a worker to make the reasoning explicit. Some model-assisted workflows removed that requirement: the decision entered a pull request before anyone had heard its rationale.

> "I know why I chose it. I mean, I think I do. But I did not have to say it out loud to anyone before it went into the PR." (P07, senior developer)

The need to justify the decision to another person is no longer required at any point before the decision becomes load-bearing. In Vallor's vocabulary, this is delegation-induced atrophy of practical judgement, visible within a code review (Vallor 2016, 2024).

4.4. Reciprocity collapse

Human help is relationally costly in ways model help is not. Asking exposes uncertainty; answering exposes the helper's judgement. Those risks can be awkward, but they also allow trust and knowledge to develop on both sides. A humanlike dialogue agent is responsive, available and unembarrassable (Frey and Weiss 2025): it supports exploration without social risk, and removes the relational work through which trust is made.

> "It is useful when it feels like brainstorming with another engineer. But you still need to be clear that the final reasoning is yours. And the model does not remember whether I was right last week." (P28, applied AI engineer)

The useful simulation gradually replaces the reciprocal practice. The loss is bidirectional: the asker is no longer shaped by a colleague's correction, and the colleague no longer learns what the asker found difficult. This bidirectional loss is a workplace instance of the collaboration gap in human–AI work (Vishwarupe et al. 2026b).

4.5. Attribution obscuration

Participants sometimes retained formal responsibility for an artefact without being able to reconstruct how a model-shaped decision had emerged. The problem was therefore not only credit, but the team's ability to trace and defend the grounds of the decision. When an idea passes through a prompt, a model completion, a human edit and a later review, the team may retain formal ownership while losing the social clarity of contribution.

> "It was hard to justify why the model did that. I mean, the answer works. But if you ask me to defend it in the design meeting, I have to reconstruct a story I did not actually go through." (P17, ML engineer)

The difficulty is not that the answer is wrong. It is that the grounds are no longer available to the team that must defend it. Boundary objects worked because the history of their use was inspectable (Star and Griesemer 1989), and the Cognitive Dimensions of Notations framework has long argued that the visibility and juxtaposability of an artefact's history are properties that determine how well a team can reason with it (Green and Blackwell 1998). Model-shaped artefacts are boundary objects whose history is compressed into a conversational trace that only one person saw.

4.6. Collective deskilling

Taken together, these shifts may reduce a team's opportunities to practise reasoning together. The concern is not that every individual necessarily loses skill, but that fewer episodes remain in which judgement is demonstrated, questioned and passed on.

> "Our juniors get answers faster now. But they get fewer chances to watch how the team reasons through a hard case. I do not know yet what that will cost." (P11, engineering manager)

The deskilling is collective because it is located not in one worker's competence but in the relational practice that previously kept judgement and know-how in circulation. LLM use does not inevitably deskill teams. Several participants described constrained, reviewable and accountable configurations. One tech lead reported that they had "deliberately limited what the model could do so the review stayed human" (P05). The design question is whether such constraints are oriented toward preserving collegial attention or only toward assigning liability after the fact.

5. DISCUSSION

The six displacements identify different points at which a model-mediated workflow can remove an interaction from team view. They overlap in practice. A question redirected to an LLM may also eliminate an occasion for explanation, hide uncertainty from a reviewer and make later attribution harder. The taxonomy therefore extends familiar CSCW concerns rather than replacing them: it asks what becomes of grounding, situated learning and inspectable artefacts when part of the work is completed in a private dialogue with a model.

The taxonomy also sits alongside recent HCI work on human–AI interaction, but shifts its focus. Amershi et al. (2019) give designers eighteen guidelines for calibrating an individual user's interaction with an AI system; witness withdrawal and articulation atrophy suggest a nineteenth concern, namely whether a well-calibrated interaction still leaves a colleague with something to review. Bansal et al. (2021) report that AI explanations can improve complementary team performance in dyadic tasks; the second-order question is whether such explanations are ever heard by a third party. Shneiderman (2020) frames human-centred AI around reliable, safe and trustworthy interaction; consultation rerouting and reciprocity collapse press further on who remains in the interaction once the model is trustworthy enough. Ashktorab et al. (2019) study repair after chatbot breakdown with one user; attribution obscuration points to a companion problem of repair among colleagues after a model-shaped decision has entered a shared artefact.

Three design orientations follow for LLM-supported software tools.

Evaluate the workflow, not only the interaction. A tool can be usable, trusted and accountable in the human–model interaction while still weakening mentoring, review and shared judgement. Evaluation protocols

should ask who is no longer consulted, which drafts are no longer witnessed, and where repair now happens.

Design for the return leg. Systems that treat the human–model interaction as complete when the answer is generated inherit the displacements above. Systems that route reasoning back to the team — by prompting for a colleague reviewer, preserving alternatives, or routing uncertainty into human discussion before closure — treat collegial attention as a design property. Candidates include code assistants that generate review questions rather than finished patches.

Document the transition while it is still visible. Many participants had worked before widespread LLM adoption. Future cohorts will not remember the pre-LLM consultation pattern; what is now experienced as displacement may later appear as ordinary process. Recording the baseline now gives later studies a point of comparison they will otherwise lack.

6. LIMITATIONS AND FUTURE WORK

This paper reports the UK subset of a larger corpus; the India comparison is treated elsewhere. The corpus is drawn from three large MNCs and three UK-based startups; patterns may present differently in public-sector settings, small firms outside technology markets, open-source communities, or user populations with distinct interaction profiles such as older adults (Maheshwari et al. 2026). Because our evidence comes from retrospective interviews, we identify reported mechanisms rather than longitudinal effects and cannot demonstrate their trajectory or test a repair. Future work should examine whether rationale logs, colleague-review prompts, attribution trails and uncertainty-routing workflows measurably preserve witness, articulation, attribution and repair in situated software teams.

7. CONCLUSION

LLMs do not merely add another tool to software work; they can alter who participates in the formation of judgement. Across these interviews, questions, explanations, corrections and learning moments sometimes moved from colleague-to-colleague exchanges into private conversations with a model. The six displacements provide a vocabulary for examining that shift. The next step is to test which workflow designs keep model-assisted reasoning visible, contestable and available for collective learning.

8. GENERATIVE AI USE DISCLOSURE

Generative AI tools assisted with LaTeX formatting checks, transcription of interview audio, a line-editing pass at initial drafting, and drafting assistance on the interview-instrument description and the paragraph engaging Amershi et al. (2019); Bansal et al. (2021); Shneiderman (2020); Ashktorab et al. (2019) at camera-ready revision. All AI-generated transcripts were checked twice by the authors against the recordings; every quotation was verified against the source audio. The argument, taxonomy, evidence selection and analytical claims are the authors' own; no AI system was used for coding, interpretation, generation of claims or fabrication of examples, quotations or references.

REFERENCES

Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P. N., Inkpen, K., Teevan, J., Kikin-Gil, R., and Horvitz, E. (2019) Guidelines for human–AI interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, 1–13.

Ashktorab, Z., Jain, M., Liao, Q. V., and Weisz, J. D. (2019) Resilient chatbots: Repair strategy preferences for conversational breakdowns. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, 1–12.

Bansal, G., Wu, T., Zhou, J., Fok, R., Nushi, B., Kamar, E., Ribeiro, M. T., and Weld, D. S. (2021) Does the whole exceed its parts? The effect of AI explanations on complementary team performance. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. ACM, 1–16.

Blackwell, A. F. (2024) *Moral Codes: Designing Alternatives to AI*. MIT Press.

Charmaz, K. (2014) *Constructing Grounded Theory*. 2nd edition. SAGE Publications.

Clark, H. H. and Brennan, S. E. (1991) Grounding in communication. In Resnick, L. B., Levine, J. M., and Teasley, S. D. (eds), *Perspectives on Socially Shared Cognition*. American Psychological Association, 127–149.

Cuadra, A., Wang, M., Stein, L. A., Jung, M. F., Dell, N., Estrin, D., and Landay, J. A. (2024) The illusion of empathy? Notes on displays of emotion in human–computer interaction. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems (CHI '24)*. ACM, 1–18.

Elo, S. and Kyngäs, H. (2008) The qualitative content analysis process. *Journal of Advanced Nursing*, 62(1): 107–115.

Frey, D. and Weiss, D. H. (2025) One person dialogues: Concerns about AI–human interactions. *Harvard Data Science Review*, 7(2). DOI: 10.1162/99608f92.01674a29.

Green, T. R. G. and Blackwell, A. F. (1998) Cognitive Dimensions of information artefacts: a tutorial. Version 1.2, October 1998.

Henrich, J., Heine, S. J., and Norenzayan, A. (2010) The weirdest people in the world? *Behavioral and Brain Sciences*, 33(2–3): 61–83.

Hoff, K. A. and Bashir, M. (2015) Trust in automation: Integrating empirical evidence on factors that influence trust. *Human Factors*, 57(3): 407–434.

Kuklani, R., Shinde, G. and Vishwarupe, V. (2025) Semantic jailbreaks and RLHF limitations in LLMs: A taxonomy, failure trace, and mitigation strategy. *International Journal of Computer Applications*, 187(27): 38–43.

Lee, J. D. and See, K. A. (2004) Trust in automation: Designing for appropriate reliance. *Human Factors*, 46(1): 50–80.

Maheshwari, S., Vishwarupe, V. and Shejul, S. (2026) FlickAI: Intent-based interfaces for enabling older adults to navigate everyday online tasks. In *Proceedings of the International Conference on Human–Computer Interaction (HCII 2026)*, 155–172.

Sarkar, A. and Drosos, I. (2025) Vibe coding: programming through conversation with artificial intelligence. In *Proceedings of the 36th Annual Conference of the Psychology of Programming Interest Group (PPIG 2025)*. arXiv:2506.23253.

Shneiderman, B. (2020) Human-centered artificial intelligence: Reliable, safe and trustworthy. *International Journal of Human–Computer Interaction*, 36(6): 495–504.

Star, S. L. and Griesemer, J. R. (1989) Institutional ecology, "translations" and boundary objects: Amateurs and professionals in Berkeley's Museum of Vertebrate Zoology, 1907–1939. *Social Studies of Science*, 19(3): 387–420.

Suchman, L. A. (1987) *Plans and Situated Actions: The Problem of Human–Machine Communication*. Cambridge University Press.

Suchman, L. A. (2007) *Human–Machine Reconfigurations: Plans and Situated Actions*. 2nd edition. Cambridge University Press.

Vallor, S. (2016) *Technology and the Virtues: A Philosophical Guide to a Future Worth Wanting*. Oxford University Press.

Vallor, S. (2024) *The AI Mirror: How to Reclaim Our Humanity in an Age of Machine Thinking*. Oxford University Press.

Vishwarupe, V., Flechais, I., Shadbolt, N. and Jirotka, M. (2026a) "To LLM, or not to LLM": How designers and developers navigate LLMs as tools or teammates. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems (CHI '26)*, Barcelona, Spain. ACM.

Vishwarupe, V., Jirotka, M., Shadbolt, N. and Flechais, I. (2026b) The collaboration gap in human–AI work. arXiv:2604.18096.