BAR Brazilian Administration Review
ANPAD
Editorial

Open Science in Three Acts: Foundations, Practice, and Implementation — Third Act

Pablo Rogers[1], Ricardo Limongi[2]

[1] Universidade Federal de Uberlândia, Faculdade de Gestão e Negócios, Uberlândia, MG, Brazil
[2] Universidade Federal de Goiás, Faculdade de Contabilidade Economia e Administração, Goiânia, GO, Brazil

How to cite: Rogers, P., & Limongi, R. (2025). Open science in three acts: Foundations, practice, and implementation – Third act. BAR-Brazilian Administration Review, *22*(3), e250162.
DOI: https://doi.org/10.1590/1807-7692bar2025250162

Publication date: September 03, 2025.

Corresponding author: Pablo Rogers, Universidade Federal de Uberlândia, Faculdade de Gestão e Negócios, Av. João Naves de Ávila, 2121, Santa Mônica, CEP 38400-902, Uberlândia, MG, Brazil.

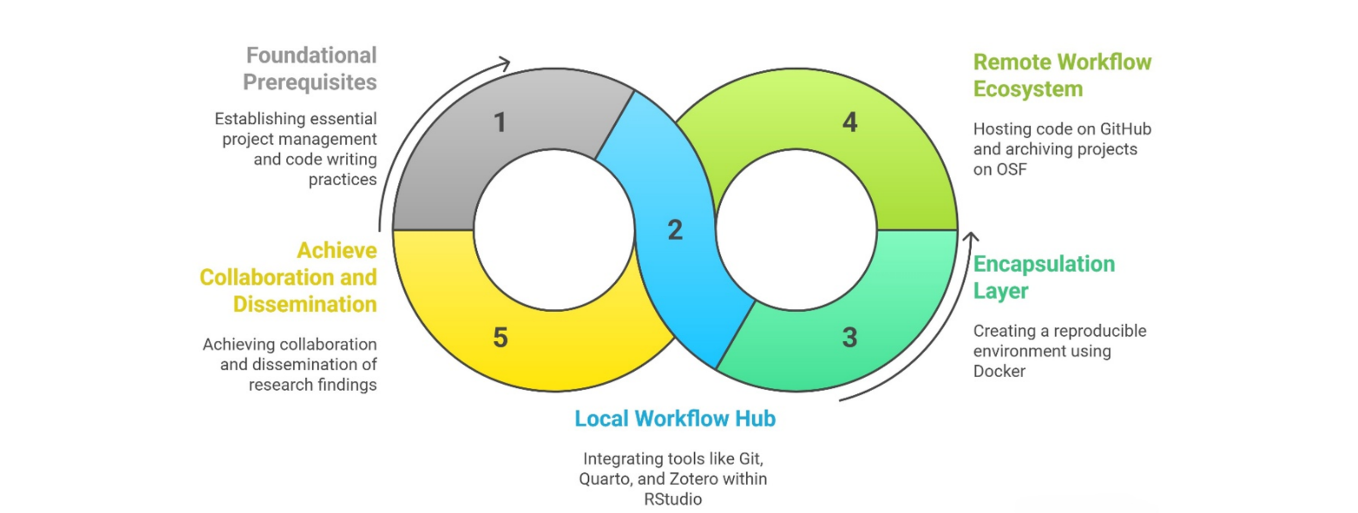
Editor-in-Chief: Ricardo Limongi (Universidade Federal de Goiás, Faculdade de Contabilidade Economia e Administração, Brazil).

Editorial assistants: Eduarda Anastacio and Simone Rafael (ANPAD, Brazil).

INTRODUCTION

This editorial concludes our three-act series dedicated to the implementation of open science (OS) in the field of research in management and applied social sciences. This text serves as the logical conclusion of the journey, offering a methodological proposal for the challenges and principles discussed previously (Limongi & Rogers, 2025a, 2025b). Our goal is to bridge the gap between theoretical knowledge and practical application, empowering researchers to adopt a more transparent, rigorous, and reproducible workflow.

In the first act of this series (Limongi & Rogers, 2025a), the fundamental problem that plagues contemporary science was diagnosed: the reproducibility crisis. It was argued that OS, with its emphasis on transparency, emerges as one of the answers to this challenge. The second act (Limongi & Rogers, 2025b) advanced from philosophy to practice, presenting an 'arsenal' of tools and workflows. The discussion has evolved from fundamental prerequisites, such as project management, to a local workflow hub, where tools such as Git (https://git-scm.com/), Quarto (https://quarto.org/), and Zotero (https://www.zotero.org/) can be integrated into RStudio (https://posit.co/), for example. Next, the encapsulation layer was explored, which creates a reproducible research environment using technologies such as Docker (https://www.docker.com/). Finally, we connected this on-premises environment to a remote workflow ecosystem, hosting code on GitHub (https://github.com/) and archiving projects in the open science framework (OSF) (https://osf.io/), to improve collaboration and dissemination of research findings. Figure 1 summarizes the discussion of the second editorial (Limongi & Rogers, 2025b).



Source: Ilustration available at: https://doi.org/10.5281/zenodo.16882841.

Figure 1. Integrated workflow for open and reproducible research.

This third and final act serves as the bridge between the knowledge of these tools and their effective application. It is recognized that the possession of an arsenal is not enough; a structure that integrates it in a cohesive way is needed. Thus, the ARTE workflow (*Article Reproducibility Template & Environment*) is implemented in the repository: https://github.com/phdpablo/article-template, a reproducible article template that operationalizes the principles of Act 1 (Limongi & Rogers, 2025a) using Act 2 technologies (Limongi & Rogers, 2025b). The purpose of this last act is therefore to present this template, demonstrate its value as a pragmatic solution, and provide a clear roadmap for its implementation, addressing different levels of rigor, from minimum to full reproducibility.

The transition to OS practices, while desirable, presents a major obstacle: the implementation gap. The second editorial (Limongi & Rogers, 2025b) of this acts presented a comprehensive toolbox, but the very wealth of options can generate a new challenge for the researcher: cognitive overload and analysis paralysis. The social scientist, whose expertise lies in their field of study and not necessarily in software engineering, is left with the complex task of selecting, configuring, and integrating a diverse set of technologies.

To appreciate the value of an integrated solution, it is critical to analyze the specific friction points in the traditional workflow:

- Initial setup complexity: Starting a new research project in a reproducible way requires a series of manual steps that represent barriers to entry, such as creating a logical folder structure, initializing a Git repository, and setting up a dependency management system.
- Inconsistency between projects: In the absence of a pattern, each new project can end up with a different structure, hindering collaboration and the researcher's own ability to revisit their past work — the so-called 'future self' problem (Rodrigues, 2023).
- The disconnect between analysis and writing: The traditional workflow, with the manual copying of results from statistical software to a word processor, is one of the most critical sources of errors and non-reproducibility.
- The learning curve: For many social scientists, tools like Git and Docker represent a fundamentally new paradigm of work, with terminologies and concepts that can be intimidating.

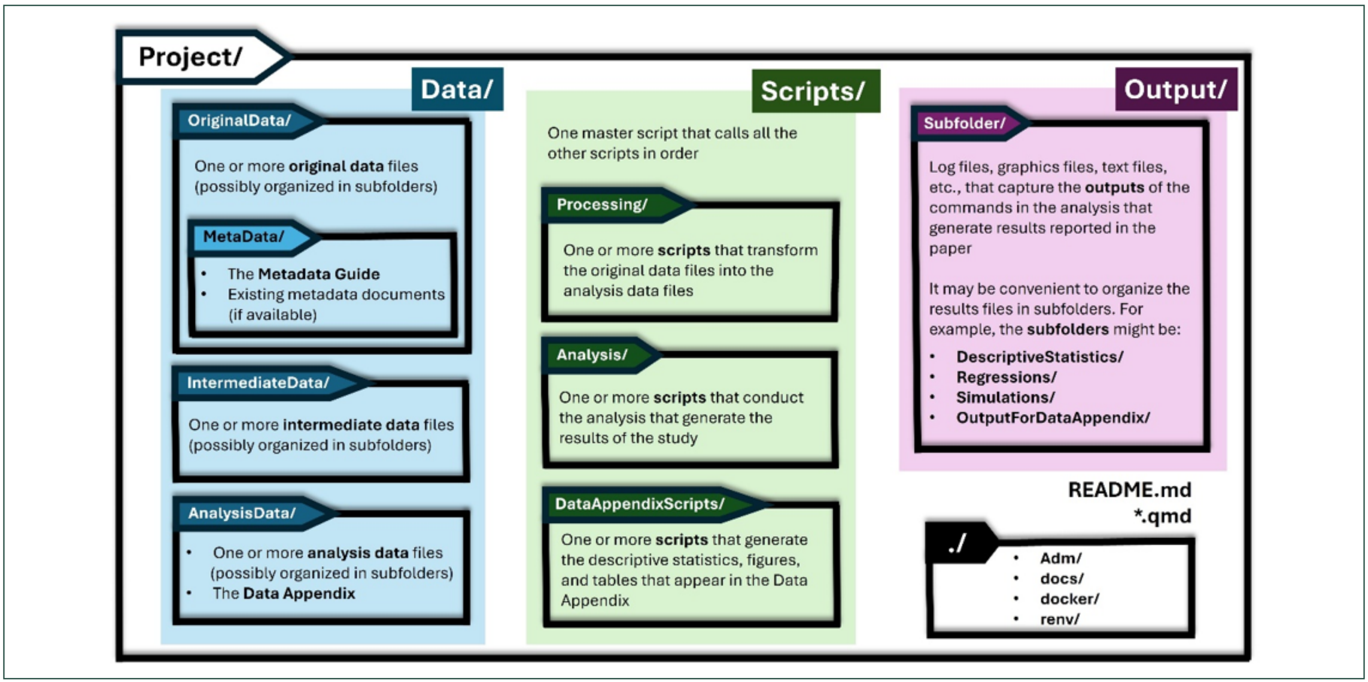
A pre-configured template can help solve not only a technical problem, but also a behavioral one: it makes best practices the most convenient option by establishing an organized structure and an automated workflow as the initial standard, which increases the likelihood of adoption.

THE ARTE WORKFLOW ARCHITECTURE

The ARTE (https://github.com/phdpablo/article-template) is designed to be an ecosystem that integrates dedicated tools, community standards, and meets the five pillars of computational reproducibility (Ziemann et al., 2023). Its architecture is designed to directly address the identified friction points, providing a solid foundation on which the researcher can build their research efficiently. More importantly, the template is flexible, allowing the researcher to choose the level of rigor they want to achieve, from 'minimum' to 'complete' reproducibility.

At the base of the template is the adoption of the TIER protocol (https://www.projecttier.org/), a community standard for organizing research projects (Domingos & Batista, 2021). The protocol logic is embodied in the main folder structure:

- Data/: Repository for all project data. The framework encourages separation between raw and processed data. The original data files must be placed in *Data/InputData* and treated as immutable. Cleanup scripts should read this data and save the analysis-ready versions to *Data/AnalysisData*. This practice ensures complete traceability.
- Scripts/: Contains all the computational code that performs the analysis, making the research logic transparent and verifiable.
- Output/: Intended for all files generated by the scripts (figures, tables, etc.). The contents of this folder must be 'disposable' (Domingos & Batista, 2021), i.e., fully regenerable at any time by running the code in the *Scripts/* folder, which constitutes the definitive proof of computational reproducibility.



Source: Ilustration available at: https://doi.org/10.5281/zenodo.16748763.

Figure 2. ARTE (Article Reproducibility Template & Environment) workflow folder structure.

For a detailed exploration of the guidelines for each folder, the reader is encouraged to consult the README.md files in each directory in the project repository (https://github.com/phdpablo/article-template).

Quarto is an essential tool of this architecture: a scientific publishing system that transforms the manuscript of a static report into a dynamic document. It integrates narrative (in Markdown), code blocks (in R, Python, or Julia), and metadata (in YAML) into a single source file (.qmd). The benefit is that the results (tables, figures, statistics) are generated directly from the code when the document is rendered, eliminating the decoupling between analysis and reporting, and ensuring perfect consistency.

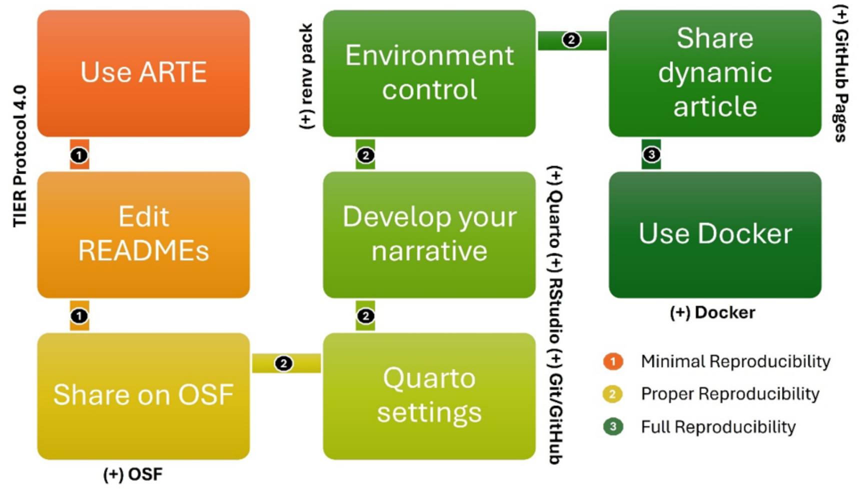
REPRODUCIBILITY SPECTRUM

ARTE (https://github.com/phdpablo/article-template) is designed to accommodate a spectrum of reproducibility practices, allowing researchers to adopt a level of rigor appropriate to their project and their competencies:

- Minimal reproducibility: Focused on the researcher with the least familiarity with command-line tools, this level is achieved by organizing the project with the TIER protocol and sharing the materials (data, scripts, manuscript) in the OSF. It ensures that all components of the survey are available and organized in a logical way. This option of this adoption would be the minimum necessary to achieve reproducibility of methods (Goodman et al., 2016).
- Adequate reproducibility: For the researcher who already uses version control, this level adds the use of Git for strict change tracking, GitHub for live document publishing, and the use of a dependency management tool such as R's renv package (https://rstudio.github.io/renv/), which records the exact versions of software packages. Together, these tools ensure that the package environment can be recreated.

- Complete reproducibility: The gold standard of reproducibility uses containerization technology, such as Docker, to encapsulate the entire computing environment — operating system, software versions, and all dependencies. This ensures that the analysis will run the same on any machine, at any time. The adoption of this option ensures computational reproducibility (Sawchuk & Khair, 2021).

Figure 3 summarizes the path that the researcher can follow in this spectrum and the related tools. In the following section, we present more details on how to implement ARTE in practice.



Source: Ilustration available at: https://doi.org/10.5281/zenodo.16883797.

Figure 3. Roadmap for developing a dynamic and reproducible research article with ARTE workflow.

There are already some proposals in the literature for solutions that fulfill a similar role to ARTE, such as WORCS — Workflow for Open Reproducible Code in Science (Van Lissa et al., 2021), REPRO — Reproducible Workflow (Peikert et al., 2021), and REPUB — Reproducible Publishing (Mang et al., 2023). However, a comparative analysis of these approaches reveals different philosophies and allows us to position the contribution of ARTE.

WORCS (https://cjvanlissa.github.io/worcs/) and REPRO (https://github.com/aaronpeikert/repro) are R packages; thus, their implementations are intended for R users. If the researcher has some knowledge of the tools used in ARTE, they can adapt the appropriate reproducibility for use in other scenarios, such as using Python with another IDE (Jupyter, for example) and another dependency management package (conda, for example). With little knowledge of containerization, the researcher will also be able to adapt another environment (Python + Jupyter + conda, for example), starting from the folder structure and rationale of ARTE, to achieve complete reproducibility.

REPUB (https://github.com/joundso/repub) resembles ARTE's proposal of complete reproducibility and combines Quarto, RStudio, and Docker to create a containerized, yet user-oriented environment that allows users to develop their work directly within the Docker container. ARTE offers flexibility that adapts to various research scenarios, and this is what we believe to be its main contribution: its nature enables gradual adoption.

ARTE is distinguished by its fundamentally pedagogical contribution. Its central innovation is the structuring of a reproducibility spectrum, which transforms the adoption of good practices from a binary choice (adopting the full flow, or not) into an incremental development journey. This scalable framework allows a researcher to start a project at the minimum level (even without technical expertise) and, as their competencies and confidence increase, advance to higher levels of rigor within the same project framework. ARTE, therefore, in addition to being just another template, works as a structured bridge between the accessible philosophy of practice and the training of new scientists.

ARTE IMPLEMENTATION GUIDE

The adoption of ARTE is designed to be scalable. For a more detailed visual guide, the reader can refer to the working model demonstration (https://phdpablo.github.io/article-template/).

Level 1: Minimum reproducibility (no Git)

This path is ideal for those who are just starting out and are not yet comfortable with Git/GitHub and/or Docker.

(1) Download the template: Navigate to the template repository (https://github.com/phdpablo/article-template) and under the green 'Code' button, select 'Download ZIP'. Unzip the file to your computer.
(2) Organize and document: Use the TIER protocol folder structure to organize your data and scripts. It is critical to edit the README.md files (or README.txt, if you prefer to rename them) in each folder to document your project's content and guidelines.
(3) Share in OSF: Create a project in OSF and upload your entire project folder. OSF allows you to version files and create a transparent record of your work. For added transparency and convenience, you can integrate your OSF project with cloud storage services like Google Drive, OneDrive, and other similar services, where files can be synced.

Level 2: Adequate reproducibility (with Git/GitHub)

This level assumes that the researcher has Git installed and configured (on the operating system and/or RStudio) and a GitHub account.

(1) Create and clone the repository: On GitHub, use the 'Use this template' option to create a personal copy of the repository (https://github.com/phdpablo/article-template). Then clone this new repository to your computer using RStudio (File > New Project > Version Control > Git).
(2) Develop the project: Follow the workflow described earlier: manage the data in the *Data/ folder*, develop the analysis in the *Scripts/ folder*, and write the narrative in the .qmd files.
(3) Write the dynamic narrative: To keep the manuscript clean, the main analysis should reside in the scripts. Instead of copying code, use the {{< embed >}} feature in Quarto (https://quarto.org/docs/authoring/notebook-embed.html) to embed analysis notebooks or other .qmd files directly into your document. This allows the analysis to be executed and its results (figures, tables) to be displayed in the desired location, combining organization and dynamism. You can see this dynamic unfolds on the template page: https://phdpablo.github.io/article-template/04-results.html.
(4) Integrate with Zotero: The template integrates with Zotero. Using the 'Better BibTeX' add-on, set up the automatic export of your library to a .bib file within the project. The Quarto configuration file (_quarto.yml) already points to this file. The template also includes a citation style file (apa7ed.csl), which can be commented out or replaced with other styles available in the official Zotero repository (https://www.zotero.org/styles).
(5) Render, version, and share: Render the article in RStudio, commit your changes with descriptive messages, and push them to GitHub. This will trigger automatic publishing (via GitHub Actions, pre-configured to upload the *docs/ folder* to the gh-pages branch) of your article as a website via GitHub Pages, as shown in the template deployment: https://phdpablo.github.io/article-template/.

Level 3: Complete reproducibility (with Docker)

This is the most advanced level and assumes that Docker Desktop is installed.

(1) Start the environment: The template simplifies the use of Docker. Instead of complex commands, simply navigate to the project's *docker/ folder* and run the appropriate startup script: *start.bat* for Windows or *start.sh* for macOS/Linux. These scripts build and start the container automatically with the name of the root folder.
(2) *Work in the containerized environment:* After running the script, access RStudio through your browser (at the address 127.0.0.1:8787, for example). The workflow can be identical to Level 2, but with the security that everything is running in an isolated and perfectly controlled environment.
(3) Shut down the environment: At the end of the process, run the *stop.bat* or *stop.sh* script in the same *docker/ folder* to stop the container safely.

For more detailed instructions and troubleshooting, the README.md in the repository root (https://github.com/phdpablo/article-template/) and the docker folder (https://github.com/phdpablo/article-template/tree/main/docker) of GitHub serve as a complete user manual. It is also worth emphasizing that all important files in the repository (_quarto.yml, Dockerfile, docker-compose.yml, etc.) are properly commented, for better user guidance.

ARTE USAGE SCENARIOS

Using Docker (full reproducibility) offers flexibility that adapts to a variety of research scenarios:

- Solo researcher: The container ensures that a project's environment remains consistent over time, protecting the researcher from their 'future self' (Rodrigues, 2023) and problems caused by software updates.

- Team research: The container eliminates the classic 'works on my machine' problem. By sharing Docker configuration files, all team members can recreate an identical development environment, simplifying collaboration and code integration.
- One project at a time: The researcher can start the active project container, work on it, and stop it at the end of the day. This keeps your computer's resources organized and dedicated.
- Multiple concurrent projects: It is possible to run multiple containers from different projects at the same time, if they are configured to use different ports (local IPs) (the *start.bat/.sh* script does this automatically). This allows you to switch between projects without the risk of dependency conflicts.
- End-use for encapsulation: A researcher can develop their entire project locally (with RStudio on their host computer, for example) and, finally, use the container to 'encapsulate' the final version of the article. This creates a replayability package (image in Docker Hub, e.g., https://hub.docker.com/) for submission to a journal or archive.
- Exclusive use in daily life: The researcher can choose to carry out all his work exclusively inside the container. Although this may reduce some computational efficiency and require additional configurations to optimize the workflow (such as configuring Git and its persistence in the container/host), it ensures the highest level of isolation and reproducibility from the project's outset.
- Hybrid use: It is possible to use the container for the main computational analyses (ensuring reproducibility) while using other tools on the host computer, such as a preferred text editor or a desktop Git client.

The advance toward a more open and transparent science depends, above all, on the consolidation of consistent practices, and not only on the incorporation of sophisticated technological resources. The reproducibility of the research is linked not only to the use of advanced tools, but also to the adoption of simple and well-structured routines (Alston & Rick, 2021).

In addition, the incessant search for perfection should not become a barrier to progress. It is preferable to share at least part of the information, even if incomplete, than to make nothing available, because each advance, no matter how small, contributes to the whole (Kathawalla et al., 2021; Klein et al., 2018). Documenting and sharing data from the earliest stages helps prevent future difficulties and optimizes the use of time during research (Klein et al., 2018). In this sense, our first recommendation is: just start, even if it is with the minimum reproducibility.

However, if you want to start with an adequate level of reproducibility to maximize the successful adoption of ARTE, a pragmatic and scaled approach is recommended rather than an immediate and exclusive immersion in the containerized environment.

The 'End-Use for Encapsulation' scenario is the most recommended starting point for researchers who are beginning with containerization. In this model, the development of the project takes place entirely in the researcher's local environment, using the workflow of adequate reproducibility (RStudio Desktop, Git/GitHub, renv). Docker is employed only in the final phase of the project, as a tool to package and validate the final version of the article and its analysis for submission or archiving.

Once the researcher has gained familiarity and confidence through the encapsulation model, 'Hybrid Use' emerges as the next step. In this scenario, the researcher uses their computer environment (host) for tasks such as text editing in .qmd files and file management, while reserving the Docker container for the execution of the most critical or sensitive computational analyses. This approach acts as a bridge to advanced proficiency.

A purist approach with the exclusive use of Docker from the beginning of the project, although technically correct, can risk non-adoption by the target audience. In this sense, the recommended trajectory — starting with 'End-Use for Encapsulation' and progressing to 'Hybrid Use' — offers a more pragmatic path and is aligned with the pedagogical philosophy of ARTE.

CHALLENGES FOR THE USE OF ARTE

Transitioning to an OS workflow, even with a proposition like ARTE, is not without its challenges. The learning curve, while attenuated, remains a constraint. The transition from a graphical interface-based environment to a script-centric workflow represents a paradigm shift. Tools such as JASP (https://jasp-stats.org/) and Jamovi (https://www.jamovi.org/) can serve as important bridges. Both are open-source, R-based, and offer user-friendly interfaces. A notable feature is the ability to save, in a single file, both data and analysis, promoting a form of self-contained reproducibility. However, this approach may have limitations in terms of computational efficiency, traceability of raw data preprocessing, and scalability for projects with large volumes of data or very complex analyses, where a script-based workflow is recommended.

In addition, the norms of a field of research exert a strong influence, including the inertia of 'house methods' (Alessandroni & Byers-Heinlein, 2022), the fear of sharing code shaming (Crüwell et al., 2019; Kathawalla et al., 2021), and a reluctance to share work-in-progress (Blischak et al., 2016; Gilroy & Kaplan, 2019). However, perhaps the most impactful barrier relates to career evaluation systems, which often prioritize the quantity of publications (product) over quality and transparency (process), representing a fundamental disincentive. The absence of formal training in reproducible computational methods in graduate curricula aggravates this problem.

Overcoming these barriers requires effort. For researchers, adopting new practices brings efficiency gains and fewer errors. For the community, the formation of study groups and workshops can create a support network. For institutions, it is necessary to make a direct appeal to academic leaders to integrate these workflows into methodology courses, and for journals and funding agencies to value and demand transparency. ARTE can be an important pedagogical tool, teaching good practices from the beginning of one's career as the standard way of doing science.

In order to help our community overcome these barriers and reduce the effort, the authors of this editorial have dedicated themselves to offering the course "Future-Proof Article: Open Science Journey in Practice" (https://phdpablo.github.io/curso-open-science/). It takes place remotely for researchers from all over Brazil, with continuous registration flow and open material constantly evolving, including installation and configuration videos of the tools/solutions proposed for ARTE (https://phdpablo.github.io/curso-open-science/00-prework.html).

CONCLUSION

With this editorial, the curtain falls on our three-act play on open science. In Act 1, we defined the problem and the principles. In Act 2, we mapped out the technological arsenal. In Act 3, we assembled the pieces, presenting an integrated workflow in the form of ARTE (https://github.com/phdpablo/article-template).

It is essential to reinforce that ARTE and the tools it integrates are not an end in themselves. Technologies are, by nature, ephemeral. Quarto can be replaced, just as R Markdown, once widely used by the scientific community, is being replaced by Quarto; Docker may evolve, and new tools will emerge. Lasting value lies not in the specific tool, but in the culture and workflow that its adoption helps to build. Learning how to structure a project logically, how to version the work, how to document each step, and how to ensure that the computing environment is controllable are perennial methodological skills.

The adoption of a system like this is an investment that reduces the future learning curve. By internalizing the principles of a reproducible workflow, the researcher will be better prepared to evaluate and incorporate new technologies critically and efficiently. The template, therefore, is a starting point, a scaffolding to build a more lasting scientific practice.

We conclude with a direct invitation to the community: for your next project, try this workflow. The transition to more open science is an incremental evolution. By adopting these practices, we are not only improving our own research, we are collectively contributing to strengthening the foundations of science, ensuring its credibility and relevance in the decades to come. The play is over, but the practice is just beginning.

REFERENCES

Alessandroni, N., & Byers-Heinlein, K. (2022). Ten strategies to foster open science in psychology and beyond. *Collabra: Psychology*, *8*(1), 57545. https://doi.org/10.1525/collabra.57545

Alston, J. M., & Rick, J. A. (2021). A beginner's guide to conducting reproducible research. *The Bulletin of the Ecological Society of America*, *102*(2), e01801. https://doi.org/10.1002/bes2.1801

Blischak, J. D., Davenport, E. R., & Wilson, G. (2016). A quick introduction to version control with git and github. *PLOS Computational Biology*, *12*(1), e1004668. https://doi.org/10.1371/journal.pcbi.1004668

Crüwell, S., Van Doorn, J., Etz, A., Makel, M. C., Moshontz, H., Niebaum, J. C., Orben, A., Parsons, S., & Schulte-Mecklenbeck, M. (2019). Seven easy steps to open science: An annotated reading list. *Zeitschrift für Psychologie*, *227*(4), 237–248. https://doi.org/10.1027/2151-2604/a000387

Domingos, A., & Batista, I. R. (2021). A map for transparency and replicability in empirical social science: The TIER Protocol. *Revista Política Hoje*, 40–86. https://doi.org/10.51359/1808-8708.2021.245776

Gilroy, S. P., & Kaplan, B. A. (2019). Furthering Open Science in Behavior Analysis: An Introduction and Tutorial for Using GitHub in Research. *Perspectives on Behavior Science*, *42*(3), 565–581. https://doi.org/10.1007/s40614-019-00202-5

Goodman, S. N., Fanelli, D., & Ioannidis, J. P. A. (2016). What does research reproducibility mean? *Science Translational Medicine*, *8*(341), 341ps12-341ps12. https://doi.org/10.1126/scitranslmed.aaf5027

Kathawalla, U.-K., Silverstein, P., & Syed, M. (2021). Easing into open science: A guide for graduate students and their advisors. *Collabra: Psychology*, *7*(1), 18684. https://doi.org/10.1525/collabra.18684

Klein, O., Hardwicke, T. E., Aust, F., Breuer, J., Danielsson, H., Mohr, A. H., I. J. zerman, H., Nilsonne, G., Vanpaemel, W., & Frank, M. C. (2018). A practical guide for transparency in psychological science. *Collabra: Psychology*, *4*(1), 20. https://doi.org/10.1525/collabra.158

Limongi, R., & Rogers, P. (2025a). Open science in three acts: Foundations, practice, and implementation - First act. *Brazilian Administration Review*, *22*(1), e250079. https://doi.org/10.1590/1807-7692bar2025250079

Limongi, R., & Rogers, P. (2025b). Open science in three acts: Foundations, practice, and implementation - Second act. *Brazilian Administration Review*, *22*(2), e250116. https://doi.org/10.1590/1807-7692bar2025250116

Mang, J. M., Prokosch, H.-U., & Kapsner, L. A. (2023). Reproducibility in 2023 – An end-to-end template for analysis and manuscript writing. In M. Hägglund, M. Blusi, S. Bonacina, L. Nilsson, I. Cort Madsen, S. Pelayo, A. Moen, A. Benis, L. Lindsköld, & P. Gallos (eds.), *Studies in Health Technology and Informatics*. IOS Press. https://doi.org/10.3233/SHTI230064

Peikert, A., Van Lissa, C. J., & Brandmaier, A. M. (2021). Reproducible research in R: A tutorial on how to do the same thing more than once. *Psych*, *3*(4), 836–867. https://doi.org/10.3390/psych3040053

Rodrigues, B. (2023). *Building reproducible analytical pipelines with R*. https://raps-with-r.dev/

Sawchuk, S. L., & Khair, S. (2021). Computational reproducibility: A practical framework for data curators. *Journal of eScience Librarianship*, *10*(3), 1206. https://doi.org/10.7191/jeslib.2021.1206

Van Lissa, C. J., Brandmaier, A. M., Brinkman, L., Lamprecht, A.-L., Peikert, A., Struiksma, M. E., & Vreede, B. M. I. (2021). WORCS: A workflow for open reproducible code in science. *Data Science*, *4*(1), 29–49. https://doi.org/10.3233/DS-210031

Ziemann, M., Poulain, P., & Bora, A. (2023). The five pillars of computational reproducibility: Bioinformatics and beyond. *Briefings in Bioinformatics*, *24*(6), bbad375. https://doi.org/10.1093/bib/bbad375

Authors

Pablo Rogers
Universidade Federal de Uberlândia, Faculdade de Gestão e Negócios
Av. João Naves de Ávila, n. 2121, Santa Mônica, CEP 38400902, Uberlândia, MG, Brazil
pablorogers@ufu.br

Ricardo Limongi
Universidade Federal de Goiás, Faculdade de Contabilidade Economia e Administração
Rua Samambaia, s/n, Chácaras Califórnia, CEP 74001-970, Goiânia, GO, Brazil
ricardolimongi@ufg.br