Scientific publication with Quarto

COUSIN Workshops Series

Felipe Ortega María Jesús Algar

2024-12-20

Table of contents

Preface

This workshop describes how to use Quarto, software for producing scientific documents and publications, in ecology and plant research.

Quarto is a powerful and versatile tool for researchers implementing **reproducible** workflows. The quest for open-access research, including the final product (manuscripts) and ancillary research materials like source code, datasets, figures, pipelines or setup files, has become a prominent concern among scholars and practitioners in many fields. Prestigious publications require authors to submit these materials alongside manuscript drafts to let other colleagues reproduce and validate the results, replicate studies in new cohorts or improve their interpretability.

Quarto combines formatted text and executable source code chunks into a single document. Code *chunks* can be written in different programming languages such as R, Python, Julia or Observable. As we will see, it is possible to combine different programming languages in the same document or collection of documents, increasing the flexibility of this tool.

This is a **practical guide**, presenting hands-on examples and code to produce your own Quarto documents quickly. In addition, key concepts and best practices are also presented to steer new Quarto apprentices in the right direction.

To learn more about Quarto visit the comprehensive guide. Quarto can produce standalone documents, books like this one, as well as complete websites.

Part I

Quarto

1 Scientific documents

In a typical research scenario, scholars and practitioners producing scientific documents must face a complex task, full of nuances and potentially problematic steps.

Figure X.X depicts an schematic overview of a classical workflow for scientific document creation. Usually, the central asset is a LaTeX or Word master file containing the document (manuscript, technical report, thesis, etc.).

FIXME: INSERT FIGURE HERE

This master file is progressively populated with content from different sources, like:

- Figures and charts resulted from running software code.
- Tables and summaries describing datasets and results.
- Evaluation and results from statistical and/or machine learning models.
- Bibliographic references.

Many of these elements force users to constantly rerun external programs, procedures and tools to update these elements and incorporate the new versions to the master file. Admittedly, this is a mostly manual, tediuos process which is also quite prone to errors and overlooks. "Wait! We forgot to update Figure A". "Are these the latest evaluation results from model M?" "Do we use the last version of dataset D?" These are common inquiries that recurrently surface during this process.

However, it would be great if that manual and at times frustrating process were not necessary. Do we have any alternative? Yes, we do. The answer to our problems is a powerful concept called **literate programming**.

Literate programming

Reproducible research

Quarto for scientific publishing

2 Document types

2.1 Individual documents

2.2 Books

2.3 Manuscripts (scholarly articles)

2.4 Presentations

2.5 Websites

2.6 Dashboards

3 Quarto workflow

3.1 Required tools

3.2 Producing HTML

3.3 Producing PDF

3.3.1 Customising PDF documents

4 Single Documents

4.1 Preamble

4.2 HTML output

4.3 PDF output

4.4 Producing documents

4.4.1 Preview

4.4.2 Selecting output type

4.4.3 Further Customisation

Part II

Quarto books

5 Creating books

5.1 Authoring tools

5.2 Reference management

5.3 Customisation and templates

6 Creting a field guide

6.1 Templates

6.2 Project management

6.3 Publishing

Part III

Publications

7 Scientific articles

7.1 Replicability in scientific publications

7.2 Publication-ready figures

7.3 Journal article templates

7.4 Examples and best practices

8 FAIR data principles

8.1 Overview

8.2 Source code publication

8.3 Dataset publication

8.4 Reference management

9 Additional resources

In summary, this book has no content whatsoever.

See Knuth (1984) for additional discussion of literate programming.

References

Knuth, D. E. (1984). Literate programming. *Comput. J.*, *27*(2), 97–111. https://doi.org/10.1093/comjnl/27.2.97

A Code reference

A.1 Quarto statements

A.2 R statements

B Integrated Development Environments for Quarto

B.1 R Studio

B.2 Visual Studio

B.3 Positron

C Useful R packages

C.1 Ecology

C.2 Data visualisation

C.3 Data processing

C.3.1 Tidyverse

C.3.2 Alternatives to the Tidyverse

C.3.3 Pipelines

C.4 Spatial data

C.4.1 `sf` (Simple Features)

C.4.2 `terra`

C.5 Time series

C.5.1 Tidyverts

C.6 Data visualisation

C.6.1 `ggplot2`

C.7 Data analysis and Machine Learning

C.7.1 Tidymodels

C.7.2 mlr3

D Producing PDF documents

D.1 PDF documents with Quarto

D.2 Quick LaTeX primer

D.3 Available templates

References

Knuth, D. E. (1984). Literate programming. *Comput. J.*, *27*(2), 97–111. https://doi.org/10.1093/comjnl/27.2.97