

<toc>

# Baroque



Workshop Programme: Publishing  
from Collections: Introducing  
Computational Publishing for Culture

</toc>



# Workshop Programme

## Publishing from Collections

by Simon Worthington

Draft 0.1



Published by: Open Science Lab

Last updated: 2023-02-18

Created: 2023-02-15

Language: English (United States)

Created by: Simon Worthington

# Contents

<b>Publishing from Collections: Introducing Computational Publishing for Culture</b>	<b>6</b>
Learning points	7
Sample publications	8
Prototype exhibition catalogue: <toc>Baroque</toc>	8
Prototype publication catalogue: ScholarlyLed Catalogue	9
Preparation needed by participants	9
Schedule	10
Activities	11
About: Example 'publishing model' and prototype publications	11
Activity A: Create a Wikidata query	12
Activity B. Painting catalogue	13
Activity C. Editing a Jupyter Notebooks and accessing video	14
(Optional) Activity D. Collaborative editing of Colophon with HedgeDoc	15
Software	16

---

# Publishing from Collections: Introducing Computational Publishing for Culture

---

Hands-on Workshop with Simon Worthington (NFDI4Culture @Open Science Lab, TIB, Hannover)

Monday, 20 February 2023

Workshop URL: <https://experimentalbooks.pubpub.org/programme-overview>

Open Notes: <https://demo.HedgeDoc.org/s/4gr9JvUS7>

Document DOI: [10.5281/zenodo.7652524](https://doi.org/10.5281/zenodo.7652524) | Author: Simon Worthington  
<https://orcid.org/0000-0002-8579-9717> | CC BY-SA 4.0 International.

*To edit [this document](#) - request access by emailing [simon.worthington@tib.eu](mailto:simon.worthington@tib.eu).*

All software used is open-source [OSI](#) licence compliant. All content and other resources are open access with open licenced.

Computational publishing was developed in the life sciences and STEM subjects to allow publishers and authors to embed executable code, visualisations and advanced media objects alongside conventional text in

a document. This hands-on workshop demonstrates one way how humanities scholars might use computational publishing.

During the workshop, we will auto-compile catalogue publications for exhibitions or publication listings from multiple open data sources; and show how such compilations can be published multi-format: web, PDF, ebook, etc. A series of exercises, using Jupyter Notebooks for code and the Quarto platform to wrap up the notebooks for multi-format outputting, will give participants a practical introduction to some of the tools, possibilities and concepts of computational publishing.

## Learning points

In the workshop two demonstration catalogue publications will be shown *<toc>Baroque</toc>* and *Publication Catalogue*. A bundle of short exercises and demonstrations have been put together to cover:

1. Rendering a multi-format publication,
2. Asynchronous collective working and making a publication from multiple remote Linked Open Data (LOD) sources,
3. Creating a Wikidata query,
4. Displaying a painting catalogue sample collection from Wikidata LOD query for a multi-format publication,
5. Editing a Jupyter Notebook in MyBinder,
6. Embedding media objects: Video - TIB AV Portal, and; 3D – Kompakkt, and
7. Real-time collaborative editing with Hedge Doc and Fidus Writer.

## Sample publications

Prototype exhibition catalogue: `<toc>Baroque</toc>`

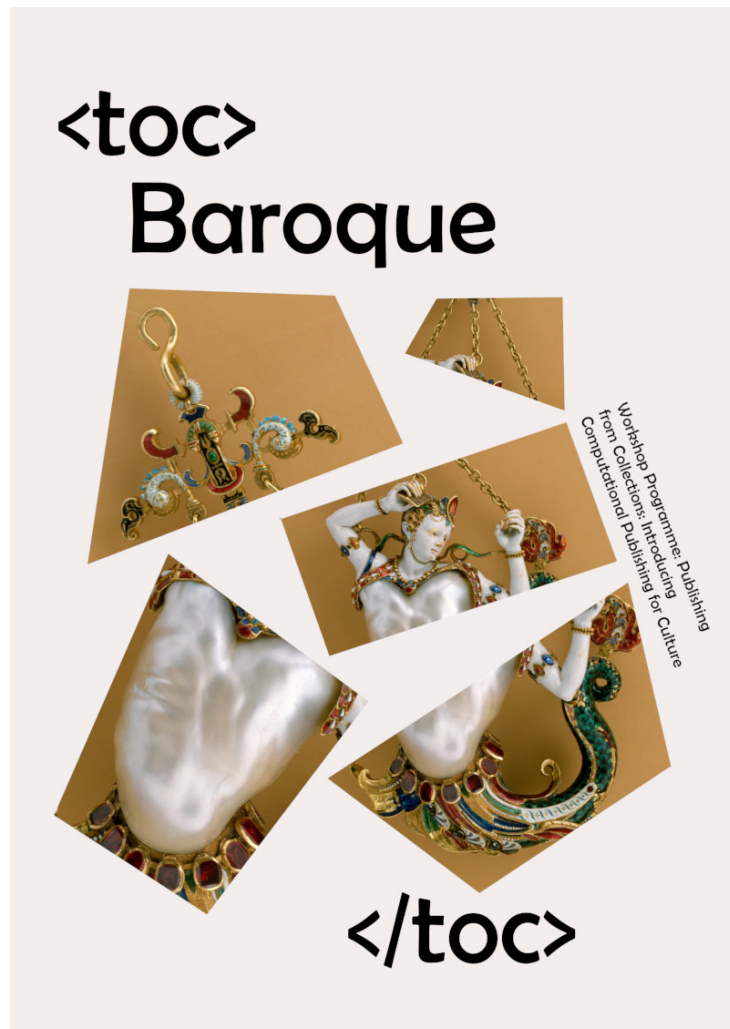


Image: [Baroque pearl with enamelled gold mounts set with rubies](#). Creative Commons CC0 1.0 Universal Public Domain Dedication. This file was donated to Wikimedia Commons as part of a project by the [Metropolitan Museum of Art](#).

A prototype framework publication for an exhibition catalogue.

The catalogue uses a Wikidata based collection of Bavarian collections of Baroque paintings. See: [17C Bavarian painting](#)

Publication

[https://simonxix.github.io/Experimental\\_Books\\_workshop/](https://simonxix.github.io/Experimental_Books_workshop/)

URL:



# Prototype publication catalogue: *ScholarLed Catalogue*

Sample output of publishers titles from the Thoth single source book metadata service API.

Publication URL: [https://simonix.github.io/scholarled\\_catalogue/](https://simonix.github.io/scholarled_catalogue/)

The screenshot displays the ScholarLed catalogue interface. On the left, there is a search bar and a navigation menu with links such as 'Introduction', 'All ScholarLed presses', and 'Open Book Publishers'. The main content area is titled 'Open Book Publishers' and includes a description of the page's purpose, a Creative Commons license notice, and the last update date. Below this, the current month 'February 2023' is highlighted, and a book titled 'Dire Straits-Education Reforms: Ideology, Vested Interests and Evidence' is featured with its cover image. On the right side, a 'Table of contents' sidebar lists months from February 2023 down to November 2020.

Screenshot: [ScholarLed catalogue](https://simonix.github.io/scholarled_catalogue/)

## Preparation needed by participants

No account logins are needed.

The workshop will involve carrying out three activities aimed at giving some familiarity with using computational publishing to retrieve and assemble linked open data and its associated media into a publication.

The activities are organised into short fifteen minute time blocks, with a ten-minute review and/or publication render.

- Activity A. Wikidata query
- Activity B. Painting catalogue
- Activity C. Editing a Jupyter Notebooks and accessing video and 3D models (Semantic Kompakkt)
- *Optional* Activity D. Collaborative editing of Colophon with HedgeDoc

You might want to familiarize yourself with some of the platforms being used in the workshop, but this is not necessary to carry out the activities.

- Wikidata query – <https://query.wikidata.org/>

- Wikimedia Commons Query Service (beta) – <https://commons-query.wikimedia.org/>
- Painting collection on Wikidata and the specific collection we'll be using – [Sum of all paintings](#) | [All painting collections](#) | [17C Bavarian painting](#).
- Jupyter Notebooks – <https://jupyter.org/>
- Quarto – <https://quarto.org/>
- Semantic Kompakkt – <https://semantic-kompakkt.de/>
- TIB AV Portal – <https://av.tib.eu/>
- ADA Semantic Publishing Pipeline – <https://github.com/NFDI4Culture/ada>
- HedgeDoc – <https://HedgeDoc.org/>
- Thoth – <https://thoth.pub/>

## Schedule

15:00-17:00, Monday, 20 February 2023.

Two hours, 120 minutes.

1. Introduction by all workshop members: **10 minutes**
  2. Overview and orientation: **5 minutes**
  3. Activity A: Wikidata query building
    1. Task: **15 minutes**
      1. Whole group review: **10 minutes**
  2. Activity B. Painting catalogue
    1. Task: **15 minutes**
      1. Render and whole group review: **10 minutes**
    2. Bio-break (stretch and refresh time): **10 minutes**
    3. Activity C. Editing a Jupyter Notebooks and accessing video and 3D models (Semantic Kompakkt)
      1. Task: **15 minutes**
        1. Render and whole group review: **10 minutes**
  2. Conclusion and demos – **20 minutes**
- 
8. Optional Activity D. Collaborative editing of Colophon with HedgeDoc

1. Task

1. Render and whole group review

END

## Activities

### About: Example 'publishing model' and prototype publications

#### Publishing model (use case)

Preparation of a multi-format exhibition catalogue from a digital collection. The workflow enables the following:

- Contributors can work on separate documents asynchronously: cover, colophon, essay, catalogue, back matter, and metadata.
- Contributors can use different platforms independently to work on documents: Jupyter Notebooks, Fidus Writer, HedgeDoc, Google Docs, etc.
- Content and media used in publication is taken directly from sources so no copying and pasting, or errors in collation
- Visualisations and data analysis can be used in publications
- Multi-format outputs: web, PDF, e-book, markdown, JATS, BITS, etc
- Automatic templated typesetting

#### Prototype publications

1. Exhibition catalogue: `<toc>Baroque</toc>`, and
2. Publication catalogue: *ScholarlyLed Catalogue*.

#### Prototype exhibition catalogue: `<toc>Baroque</toc>`

A prototype framework publication for an exhibition catalogue.

The catalogue uses a Wikidata based collection of Bavarian collections of Baroque paintings. See: [17C Bavarian painting](#)

Publication

URL:

[https://simonxix.github.io/Experimental\\_Books\\_workshop/](https://simonxix.github.io/Experimental_Books_workshop/)

#### Prototype publication catalogue: ScholarlyLed Catalogue

Sample output of publishers titles from the Thoth single source book metadata service API.

The publication uses data from <https://thoth.pub/>

Publication URL: [https://simonxix.github.io/scholarled\\_catalogue/](https://simonxix.github.io/scholarled_catalogue/)

## Activity A: Create a Wikidata query

**Objective:** User builds a Wikidata query. See example query: paintings, Bavarian Collections, 1590 - 1750 - [query link](#)

**External LOD and media used:** Wikidata LOD, and Wiki Commons, Web Gallery of Art (images) - <https://www.wga.hu/>

**Notes:** Wikidata Query ([help](#))

- Allows for non-expert query building with plain language
- View query as plain language and as code
- Experience of building a query
- Contact with some basic building blocks of Wikidata
- View and export SPARQL query

### Steps

1. Go to <https://query.wikidata.org/>
2. Build a query around the [17C Bavarian painting](#) collection to replicate the catalogue selection to be used in Activity B. Example:
  1. Code:  
[https://github.com/SimonXIX/Experimental\\_Books\\_workshop/blob/main/paintings.ipynb](https://github.com/SimonXIX/Experimental_Books_workshop/blob/main/paintings.ipynb)
  2. Rendering:  
[https://simonxix.github.io/Experimental\\_Books\\_workshop/paintings.html](https://simonxix.github.io/Experimental_Books_workshop/paintings.html)
  3. Example: Paintings; in collection; Bavarian Collections; 1600 - 1700 - [query link](#)
3. **Step-by-step instructions** to replicate parts of this [query link](#) base on this collection [17C Bavarian painting](#):
  1. Go to <https://query.wikidata.org/>
  2. Enable split view with **i info** button top left.
  3. Filter: **instance of** [P31](#), **painting** [Q3305213](#) - wdt:P31 wd:Q3305213.
  4. Filter: **collection** [P195](#), **Bavarian State Painting Collection** [Q812285](#) - wdt:P195 wd:Q812285.
  5. Play button - bottom left - renders query below
  6. Show: **creator** [P170](#); **image** [P18](#); **copyright status** [P6216](#); **inception** [P571](#).

7. Play button - bottom left - renders query below
8. Image grid view :-)
9. Limit
10. Dates from to 1590-1750 (code only) BIND(YEAR(?inception) AS ?inceptionyear) FILTER((1590 <= ?inceptionyear) && (?inceptionyear < 1750 ))
11. Link query: <https://w.wiki/6MGX> - results: <https://w.wiki/6MGY>
4. Participants can change the selection criteria around the available criteria: artists, dates, etc., as in collection [17C Bavarian painting](#)
5. Completion: Paste your query link into the HedgeDoc link provided. <https://demo.HedgeDoc.org/s/4gr9JvUS7> - END of activity.

## Activity B. Painting catalogue

*NB: This is a guided demonstration activity due to the large number of steps involved. Participants will be able to make suggestions and requests for edits.*

**Objective:** Make a section of nine paintings for the exhibition catalogue to be selected from Wikidata and rendered multi-format in Quarto.

**External LOD and media used:** Wikidata LOD, and Wiki commons, Web Gallery of Art - <https://www.wga.hu/>

**Notes:** Jupyter Notebooks retrieval of content from Wikidata and rendering in Quarto.

- Review a query being used in Jupyter Notebooks
- Inclusion of media in a publication
- Editing a Jupyter Notebook
- Multi-format outputs
- Basics of Quarto rendering and publishing

### Links:

1. Code: [https://github.com/SimonXIX/Experimental\\_Books\\_workshop/blob/main/paintings.ipynb](https://github.com/SimonXIX/Experimental_Books_workshop/blob/main/paintings.ipynb)
2. Rendering: [https://simonxix.github.io/Experimental\\_Books\\_workshop/paintings.html](https://simonxix.github.io/Experimental_Books_workshop/paintings.html)

### Steps

1. View code and rendering and explain steps

2. Demonstrate a change in the Notebook code configuration and output, simplest is the increase in the LIMIT of items.
3. Show and talk through full items as they exist on Wikidata.
4. Depending on current Notebook configuration invite suggestions for changes to the configurations of items being retrieved by the Notebook processing. If this is possible – re-render the publication.
5. Show the TOC feature of Quarto of how documents are combined!!! On GitHub the [TOC](#) can be seen. At some point W3C table of contents concepts ([doc\\_toc](#)) and [Manifest](#) standards could be used which would extend the interoperability.

End of activity: Bio break!

## Activity C. Editing a Jupyter Notebooks and accessing video

**Objective:** Running and editing Jupyter Notebooks in MyBinder and retrieving video and 3D models as embeds.

**External LOD and media used:** TIB AV Portal, and Semantic Kompakkt

**Notes:** Jupyter Notebooks editing in MyBinder

- Run a Jupyter Notebook in MyBinder
- Edit a Jupyter Notebook
- Render a Jupyter Notebooks

### Links:

- Sample Jupyter Notebook: [Video and 3D Notebook embeds](#)
- TIB AV Portal: <https://av.tib.eu/>
- Semantic Kompakkt demo site: <https://kompakkt.wbworkshop.tibwiki.io/explore>
- View a model, copy the iframe embed from the folder icon, top right. In the Notebook paste in the complete iframe cover replacing the existing iframe: `<iframe name="Doric Column" src="https://kompakkt.wbworkshop.tibwiki.io/viewer/?entity=63e8c22910e4f555d1f656ca&mode=open" allowfullscreen loading="lazy" ></iframe>`

### Steps

1. Open Notebook in the browser using MyBinder - [Video and 3D Notebook embeds](#) - click the 'launch binder' button to run the Notebook in MyBinder.

2. Add new videos and 3D models to the Notebook from TIB AV Portal and Semantic Kompakkt.
  1. Open a second browser tab and load [TIB AV Portal](https://av.tib.eu/)
  2. Choose a video and copy across the video ID from the URL <https://av.tib.eu/media/60729>
  3. Paste the video ID into the video iframe field and run the cell to render
  4. Open Semantic Kompakkt demo site: <https://kompakkt.wbworkshop.tibwiki.io/explore>
  5. View a model, copy the iframe embed from the folder icon, top right. In the Notebook paste in the complete iframe cover replacing the existing iframe: `<iframe name="Doric Column" src="https://kompakkt.wbworkshop.tibwiki.io/viewer/?entity=63e8c22910e4f555d1f656ca&mode=open" allowfullscreen loading="lazy" ></iframe>`
3. Run the Notebook
4. 3D view size, we can make the initial view bigger, add: `<iframe width="1200" height="630"`
5. Download Notebook
6. Render some videos and 3D models in the Quarto book. Pass along **video id codes** and **3d models** using a hedge doc and chat to the Quarto render. The rendering and final display will take less than 10 minutes (hopefully):
  - a. The code needs to be added to the main repo;
  - b. Rendered locally;
  - c. Uploaded to GitHub;
  - d. Time for GitHub Pages to finish loading.
  1. Code: [https://github.com/SimonXIX/Experimental\\_Books\\_workshop/blob/main/paintings.ipynb](https://github.com/SimonXIX/Experimental_Books_workshop/blob/main/paintings.ipynb)
  2. Rendering: [https://simonxix.github.io/Experimental\\_Books\\_workshop/paintings.html](https://simonxix.github.io/Experimental_Books_workshop/paintings.html)

## (Optional) Activity D. Collaborative editing of Colophon with HedgeDoc

**Objective:** Editing 'Colophon' HedgeDoc to be included in publication. Quarto TOC function allows documented to be edited in different remote applications – which is one approach to asynchronous editing.

**External editing platform:** HedgeDoc

**Notes:** Remote async editing

- Editing with HedgeDoc
- Multi-user real-time editing
- Editing a Markdown document
- Saving as a Gist Git document (temporary file)
- Saving to Quarto

### Links:

- Colophon HedgeDoc: <https://demo.hedgedoc.org/s/Xzv9WrAjq>

### Steps

1. Colophon HedgeDoc: <https://demo.hedgedoc.org/s/Xzv9WrAjq>
2. Save as Gist
3. Make commit to Git Repo
4. Render Quarto

## Software

Over 2023/24 the computational components will be added to the **ADA Semantic Publishing Pipeline** as well as introducing **Vivliostyle Create Book** markdown renderer and swapping to **Jupyter Book** computational book platform away from Quarto – <https://github.com/NFDI4Culture/ada>

- Wikidata – <https://www.wikidata.org/>
- Jupyter Notebooks – <https://jupyter.org/>
- Jupyter Book – <https://jupyterbook.org/>
- Quarto – <https://quarto.org/>
- Semantic Kompakkt – <https://semantic-kompakkt.de/>
- TIB AV Portal – <https://av.tib.eu/>
- HedgeDoc – <https://HedgeDoc.org/>
- Thoth – <https://thoth.pub/>
- Vivliostyle – <https://vivliostyle.org/>
  - [Create Book](#) – Markdown renderer
- Wikibase – <https://wikiba.se/>

---

*To edit [this document](#) - request access by emailing [simon.worthington@tib.eu](mailto:simon.worthington@tib.eu).*