

# Table of contents

<b>1 Abstract</b>	<b>1</b>
<b>2 Front matter</b>	<b>2</b>
<b>3 Cross-referencing</b>	<b>5</b>
3.1 Amsthm . . . . .	9
3.2 Diagrams . . . . .	11
3.3 Equations . . . . .	14
3.4 Figures . . . . .	15
3.5 Listings . . . . .	17
3.6 Tables . . . . .	18
3.7 Sections . . . . .	19
3.7.1 Section . . . . .	19
Section {-} . . . . .	20
3.7.2 Heading . . . . .	20
3.7.3 Heading + Break . . . . .	20
3.7.4 Section + Break . . . . .	21
3.8 Footnotes . . . . .	22
<b>4 Citations</b>	<b>23</b>
4.1 Multipart Bibliography . . . . .	24
4.2 Basic citations . . . . .	28
4.3 Advanced citations . . . . .	29
4.4 Citation Backlinks . . . . .	31
<b>5 Resources</b>	<b>34</b>
5.1 Callout . . . . .	34
5.2 Layout . . . . .	35
5.3 Generic Divs . . . . .	35
<b>6 References</b>	<b>37</b>
6.1 Primary Sources . . . . .	37
6.2 Secondary Sources . . . . .	37
6.3 Workflow . . . . .	37
6.4 Songs . . . . .	37
<b>Cite Tools Samples</b>	<b>39</b>

# List of Figures

2.1 Instead of using the <b>Title</b> , the YAML key-value pair is usually formed by item's <\\$custom:ID>, and <\\$custom:Value> or <b>Text</b> to allow the of more descriptive titles. The new front matter also makes it a breeze to edit parameters without disturbing YAML's sensitive white space rules, and it makes it much easier to revert to a working configuration after introducing accidental errors. . . . .	3
2.2 This demonstrates how to keep, entirely within Scrivener, a bibliography in CSL-YAML, the format favored by Pandoc and Quarto (being +10x faster to process than BibTeX and RIS). You can find this sample in the templates folder. . . . .	4
3.1 <b>Replacements</b> pane in the <b>Compile...</b> configurations can be used to allow different labels for automatic links. This is purely optional and we recommend limiting this to one rule or two. . . . .	6
3.2 Figure caption . . . . .	11
3.3 A graphviz graph with figure reference and caption, using raw markup. Currently in LaTeX this could overflow the page depending on verso/recto, but renders fine in HTML; see <a href="https://quarto.org/docs/authoring/diagrams.html#sizing">https://quarto.org/docs/authoring/diagrams.html#sizing</a> for more details... . . . . .	12
3.4 A Graphviz-generated state machine diagram . . . . .	12
3.5 Figure caption . . . . .	13
3.6 Figure caption . . . . .	13
3.7 This figure uses custom metadata values to identify the class, ID, width and height. The %CA% ( <b>Caption Attributes</b> ) tag at the start of the caption is replaced with the correct Scrivener placeholders by the compiler; see global replacements for the details... . . . . .	15
3.8 This demonstrates generating a multi-panel figure using a Scrivener Section Type [Multipart Figure] instead of using raw markdown as shown here. ID, Class, and Attributes specific to the block [#fig-elephants2 .column-body layout-ncol=2 layout-align="bottom"] are saved to <u>Custom Metadata-&gt;ID, Class &amp; Attributes</u> , and this is then inserted into the markup for this chunk by the Section Layout at compile time. . . . .	16

## List of Figures

3.9	Quarto allows the creation of figure panels with sub-figures. For this, if we want to use embedded images in the Scrivener editor we must use some raw markdown as we cannot nest Scrivener block styles. Note we can use the Scale Image... Tool in Scrivener and these sizes get exported to Quarto and the output. Here we scale both images to the same height. . . . .	16
4.1	Cite Tool bundles together several lua filters to address complex bibliography demands while keeping the output consistent across formats. If you have suggestions for improvements or bug reports, please <a href="#">open an issue at the citetools repository</a> . Logo image generated with Dall-E using “ <i>Enso-like round black and white painting with ancient greek war-ship with a man tied to the mast as prompt</i> ”. . . . .	23
4.2	This will result in a separate file, whose path will be added to the front matter, with the content of the text. The resulting bibliography will print right where we placed it in our project. . . . .	25
4.3	On the left we have the <b>Table of Contents</b> ; on the top-right, we see the printed bibliography (page 31) and, on the bottom-right, the files are automatically created during the compilation process. . . . .	26
4.4	The shared templates folder in the main Scrivener configuration window. . . . .	26
4.5	The <u>Add new document</u> menu in the Toolbar. . . . .	27
6.1	Multipart bibliography with sections, such as primary sources and secondary sources . . . . .	39
6.2	Cite Field allows the evocation of arbitrary information from the references, such as author, editor, translator (using CSL variables name conventions) . . . . .	39
6.3	The <b>Citation Backlinks</b> filter adds an index of cited references to the bibliography, with links back to all in-text citations. It also allows the user to turn these off globally or in an <i>ad hoc</i> fashion. . . . .	40

# List of Tables

3.1	Cross-referencing amsthm theorems, diagrams, figures, listings, tables, and sections. . . . .	8
3.2	Cross-referencing amsthm elements in ScrivQ. . . . .	9
3.3	Cross-referencing Dot and Mermaid diagrams. . . . .	11
3.4	Cross-referencing equations. . . . .	14
3.5	Cross-referencing figures. . . . .	15
3.6	Cross-referencing listings. . . . .	17
3.7	Cross-referencing tables. . . . .	18
3.8	This table uses <b>Text</b> as the <b>Section Type</b> , and <b>Table Caption</b> as the <b>Paragraph Style</b> for the caption. . . . .	18
3.9	This is an example of Table as <b>Section Type</b> . The caption and the remaining attributes are added as part of the Section Type markup. . . . .	18
3.10	This is a markdown multi-table panel with two sub-tables generated using a Section Type [ <b>Multipart Table</b> ]. Note that Custom Metadata holds the cross-referencing label, layout class, and the attributes for this multipart table, which will be added by the Section Layout by the compiler, using the Scrivener placeholders: <code>&lt;\$custom:Class&gt; &lt;\$custom:Attributes&gt;</code> . . . . .	19
3.11	Cross-referencing sections. . . . .	19
4.1	Citation syntax in Quarto and Pandoc. . . . .	28
4.2	All ready-made <b>Character Styles</b> for the Cite Field lua filter. . . . .	30
4.3	CSL-YAML/CSL-JSON variables alongside corresponding BibTeX fields and RIS tags. Those marked with an asterisk exist and correspond, but, for some reason, Pandoc ignores them instead of converting to CSL. . . . .	31

# 1 Abstract

The **ScrivQ** template programs the Scrivener writing environment – **Compile Formats**, **Section Layouts**, **Section Types**, **Paragraph Styles**, **Character Styles**, text templates with raw markup, and even the icons – for publishing **Quarto Books** (PDF, DOCX, and HTML). It descends from Ian’s excellent [Scrivomatic](#) and [Quarto template for Scrivener](#), making use of a modified version of its original ruby script to perform additional tasks, such as repositioning parts of the markup (cross-references, footnotes), *optionally* adding and committing using the content from Git Commit, and splitting the text into the multiple files needed for a Quarto project – such as the lua filters (`_extensions`), the project metadata (`_quarto`), and the bibliographies (Primary Sources, Secondary Sources, Workflow) (if they already exist in the system, they are overwritten; other files are not modified). The **ScrivQ** template should be able to compile with **zero configuration** if both **Scrivener** and **Quarto** are installed (for PDF, `tinytex` is also required<sup>1</sup>).

---

<sup>1</sup>Install using `quarto install tinytex`. If it doesn’t compile, [open an issue at Github](#) or join [the discussion at the Scrivener user forum](#) with all the relevant information.

## 2 Front matter

A notable feature of the template is the front matter. Instead of using a single binder item for all, we are using one for each YAML parameter, with the idea of having them ready to be added or removed by simply ticking a box. Most options include a bookmarks linking to the relevant section in the official Quarto documentation and a small synopsis. We use this strategy to control a high number of variables, such as the labels involved in cross-referencing (Figure 2.1). Other complex tasks can also be managed, and are demonstrated here as a proof-of-concept, such as keeping a bibliography in CSL-YAML (Figure 2.2) or controlling the behavior of Quarto websites.

### A sworn of parameters

Many parameters are included for completeness and could be erased if they are not in use. Should they become necessary, they can be retrieved again from a newly created project.

Looking for one way to control **Quarto** from **Scrivener**, we find not *one*, but *many* ways of doing so. So much that we are reminded of Socrates addressing Meno, in the homonymous dialogue, saying that, in looking for *the virtue of human excellence*, he had found a sworn of them coming from his interlocutor.

### Binder glitches

In some cases, the sheer number of items can cause the **Binder** to behave in strange ways. If you notice any glitches, collapse and expand the parent item for the children to be properly displayed. Removing unused parameters should alleviate the problem.

Apart from one-click compilation, and facilitated parameter settings, two priorities in **ScrivQ** are cross-referencing (Section 3) and bibliography (Section 4).

Socrates: “*I seem to be in good luck, Meno; for in seeking one virtue I have discovered a whole swarm of virtues there in your keeping.*” ΣΩ. Πολλῇ γέ τινι εύτυχιδι ἔσικα κεχρῆσθαι, ὡς Μένων, εἰ μίαν ζητῶν ἀρετὴν σμῆνός τι ἀνηγόρηκα ἀρετῶν παρὰ σοὶ κειμένων (Plat. *Men.* 72A-B).

Label	In...	Title	Value	Section Type	Locked	Status	>
Metadata	✓	> @ metadata		id: value (0)	✓	✓	⋮
Rendere...	✓	✓ < filters		id: value (0)	✓	⋮	⋮
Rendere...	✓	✓ citetools	citetools	-value (2)	✓	⋮	⋮
Refere...	✓	✓ citation		id: value (0)	✓	⋮	⋮
Crossref	✓	✓ crossref		id: value (0)	✓	✓	⋮
Crossref	✓	✓ Labels	roman I	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Hyperlink	true	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Reset every chapter	false	id: value (2)	✓	✗	⋮
Crossref	✓	✓ Delimiter between Title and c...	—	id: value (2)	✓	✓	⋮
Crossref	✓	✓ Equation Title	Citação	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Figure Title	Figura	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Listing Title	Bloco	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Table Title	Quadro	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Conjuncture Title	Conjuntura	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Corollary Title	Corolário	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Definition Title	Definição	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Example Title	Passagem	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Exercise Title	Exercício	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Lemma Title	Lema	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Proposition Title	Proposição	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Theorem Title	Teorema	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Equation Labels	alpha A	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Figure Labels	arabic	id: value (2)	✓	⋮	⋮
Crossref	✓	✓ Listing Labels	arabic	id: value (2)	✓	⋮	⋮

Figure 2.1: Instead of using the Title, the YAML key-value pair is usually formed by item's `<\$custom: ID>`, and `<\$custom: Value>` or Text to allow the of more descriptive titles. The new front matter also makes it a breeze to edit parameters without disturbing YAML's sensitive white space rules, and it makes it much easier to revert to a working configuration after introducing accidental errors.

Title	Value
> @ ID	Bodnar2018
✓ @ ID	DA1995
📘 TYPE	book
▼ 🌱 AUTHOR	
👤 FAMILY	Aristotelis
▼ 📜 EDITOR	
👤 FAMILY	Biehl
👤 GIVEN	Wilhelm
👤 FAMILY	Apelt
👤 GIVEN	Otto
▼ 📜 TRANSLATOR	
👤 FAMILY	Seidl
👤 GIVEN	Horst
👤 FAMILY	Theiler
👤 GIVEN	Willy
📘 TITLE	De Anima
📘 SUBTITLE	Griechisch - Deutsch
💼 COLLECTION-TITLE	Philosophische Bibliothek
▼ 📅 ISSUED	
📅 YEAR	1995
💼 PUBLISHER	Felix Meiner
📍 PUBLISHER-PLACE	Hamburg
📋 ABSTRACT	See <a href="https://www.amazon.com/Über-die-Seele-Anima-Aristoteles/dp/3787313818">https://www.amazon.com/Über-die-Seele-Anima-Aristoteles/dp/3787313818</a>
🌐 LANG	DE, GRC
▼ 🔍 KEYWORDS	
🔍 KEYWORD	Aristoteles
🔍 KEYWORD	Seele
🔍 KEYWORD	Naturphilosophie
📘 NOTES	

Figure 2.2: This demonstrates how to keep, entirely within Scrivener, a bibliography in CSL-YAML, the format favored by Pandoc and Quarto (being +10x faster to process than BibTeX and RIS). You can find this sample in the templates folder.

# 3 Cross-referencing

With all the affordances of **Scrivener** and **Quarto**, cross-referencing is not a trivial matter, as the options are many.

First, bear in mind that **Section Types** and **Paragraph Styles** are rigged with automatic IDs in the format `scriv<$linkID>`<sup>1</sup> (preceded by the relevant prefix, such as `sec`, `cnj`, `cor`, `def`, `exm`, `exr`, `lem`, `prp`, `thm`, `eq`, `lst`, `fig`, `tbl`) . This way, there is no need to choose an ID each time an element is created, nor to remember any when another needs to be referenced (to create links we will use this same standard identifier, `scriv<$linkID>`, select the text, link to the appropriate document, and apply the style corresponding to the element we want to reference). We leave it to Scrivener to figure out the value of the `<$linkID>` placeholder.

Automatic IDs

## Translating Quarto into Scrivener

In **ScrivQ** we can use **Section Types** or **Paragraph Styles** to create **Sections**, **Tables**, **Equations**, **Figures**, **Listings**, **Callouts** (Caution, Important, Note, Tip, Warning), and **Amsthm** environments (Conjecture, Corollary, Definition, Example, Exercise, Lemma, Proposition, Theorem). We can also use **Character Styles** to easily reference any of them. Keep reading to learn how.

## Choosing your own label for automatic links

In **ScrivQ** one can use different keywords as labels for automatic links. Simply use one of the provided rules for **Replacements** (Figure 3.1) in the Compile settings (or in the Format configurations) to have keywords such as `scriv + link`, `auto + ref`, `%auto + ref%`, `%autoref: + something-random-that-will-be-erased%`, `[autoref]`, converted into `scriv<$linkID>` during compilation.

Add prefix and markup  
using Character Styles

<sup>1</sup>Note that in Scrivener we have to escape the \$, otherwise the placeholder will get expanded into its correct value during compilation.

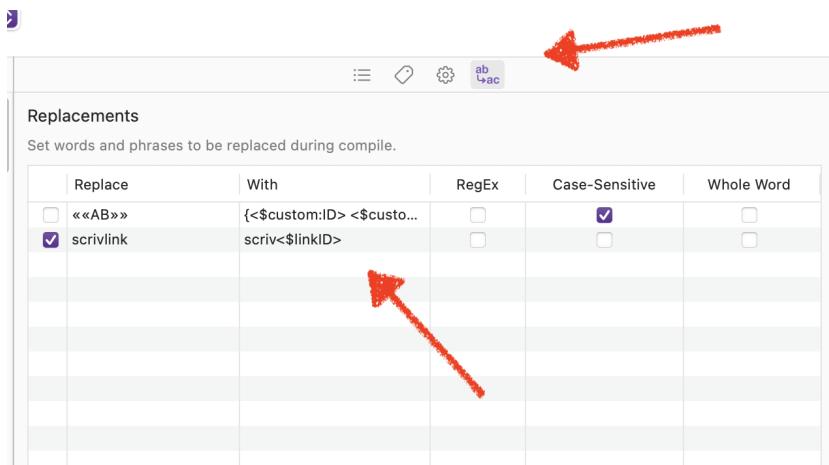


Figure 3.1: **Replacements** pane in the **Compile...** configurations can be used to allow different labels for automatic links. This is purely optional and we recommend limiting this to one rule or two.

To cross-reference a **table**, an **example**, or a **theorem**, one could use tbl-keyword (*i.e.* tbl-scriv<\$linkID>), exm-keyword (*i.e.* exm-scriv<\$linkID>), and thm-keyword (*i.e.* thm-scriv<\$linkID>), respectively. Seeing that the prefixes are not always easy to remember, **Character Styles** are available to inject the correct markup.

The *Crossref Table*, for example, will turn the keyword into [@tbl-keyword]<sup>2</sup>; and the *CrossrefTable\** style will turn it into [@tbl-keyword]<sup>3</sup>.

Likewise, the *Crossref Example* and the *Crossref Example\** will result in [@exm-keyword] and [-@exm-keyword]<sup>4</sup>, and so on. See *scriv4* below for yet more examples.

### 💡 Cross-referencing a table

1. Type your-keyword-of-choice or scriv<\$linkID>, select it, and hit Command + L;
2. Link to the document that contains the table.
3. Apply a **Character Style** called *CrossrefTable* (e.g. Table 3.1).

The asterisk (\*) in the title of Character Styles indicates the suppression of part of the data (as is common in LaTeX).

Below we will see several examples of the same strategy being applied to several different elements. I hope that these examples prove as instructive to consult as they were to prepare.

<sup>2</sup>That is, [@tbl-scriv<\\$linkID>].

<sup>3</sup>That is, [-@tbl-scriv<\\$linkID>].

<sup>4</sup>That is, [@exm-scriv<\\$linkID>] and [-@exm-scriv<\\$linkID>].

Genus	Species	Markdown Source	Rendered Output
Amsthm	Conjecture	[@cnj-scriv4]	Conjecture <a href="#">3.1.1</a>
Amsthm	Conjecture	[@cnj-scriv5]	Conjecture <a href="#">3.1.2</a>
Amsthm	Corollary	[@cor-scriv4]	Corollary <a href="#">3.1.1</a>
Amsthm	Corollary	[@cor-scriv6]	Corollary <a href="#">3.1.2</a>
Amsthm	Definition	[@def-scriv4]	Definition <a href="#">3.1.1</a>
Amsthm	Definition	[@def-scriv7]	Definition <a href="#">3.1.2</a>
Amsthm	Example	[@exm-scriv4]	Example <a href="#">3.1.1</a>
Amsthm	Example	[@exm-scriv8]	Example <a href="#">3.1.2</a>
Amsthm	Exercise	[@exr-scriv4]	Exercise <a href="#">3.1.1</a>
Amsthm	Exercise	[@exr-scriv9]	Exercise <a href="#">3.1.2</a>
Amsthm	Lemma	[@lem-scriv4]	Lemma <a href="#">3.1.1</a>
Amsthm	Lemma	[@lem-scriv10]	Lemma <a href="#">3.1.2</a>
Amsthm	Proposition	[@prp-scriv4]	Proposition <a href="#">3.1.1</a>
Amsthm	Proposition	[@prp-scriv11]	Proposition <a href="#">3.1.2</a>
Amsthm	Theorem	[@thm-scriv4]	Theorem <a href="#">3.1.1</a>
Amsthm	Theorem	[@thm-scriv12]	Theorem <a href="#">3.1.3</a>
Diagram	Dot	[@fig-scriv14]	Figure <a href="#">3.2</a>
Diagram	Dot	[@fig-scriv14B]	Figure <a href="#">3.3</a>
Diagram	Dot	[@fig-scriv15]	Figure <a href="#">3.4</a>
Diagram	Mermaid	[@fig-scriv16]	Figure <a href="#">3.5</a>
Diagram	Mermaid	[@fig-scriv16B]	Figure <a href="#">3.6</a>
Diagram	Mermaid	[@fig-scriv17]	Figure ??
Cross-reference	Equation	[@eq-scriv19]	Equation <a href="#">3.1</a>
Cross-reference	Equation	[@eq-scriv20]	Equation <a href="#">3.2</a>
Cross-reference	Figure	[@fig-scriv22]	Figure <a href="#">3.7</a>
Cross-reference	Listing	[@lst-scriv26]	Listing <a href="#">3.1</a>
Cross-reference	Listing	[@lst-scriv27]	Listing <a href="#">3.2</a>
Cross-reference	Table	[@tbl-scriv30]	Table <a href="#">3.8</a>
Cross-reference	Table	[@tbl-scriv31]	Table <a href="#">3.9</a>
Cross-reference	Section	[@sec-scriv33]	Section <a href="#">3.7</a>
Cross-reference	Section	[@sec-scriv34]	Section <a href="#">3.7.1</a>
Cross-reference	Section	[@sec-scriv36]	Section <a href="#">3.7.2</a>
Cross-reference	Section	[@sec-scriv38]	Section <a href="#">3.7.4</a>
Multipart	Figure	[@fig-scriv23]	Figure <a href="#">3.8</a>
Multipart	Figure	[@fig-scriv23A]	Figure <a href="#">3.8a</a>
Multipart	Figure	[@fig-scriv23B]	Figure <a href="#">3.8b</a>
Multipart	Figure	[@fig-scriv24]	Figure <a href="#">3.9</a>

Genus	Species	Markdown Source	Rendered Output
Multipart	Figure	[@fig-scriv24A]	Figure 3.9a
Multipart	Figure	[@fig-scriv24B]	Figure 3.9b
Multipart	Table	[@tbl-scriv32]	Table 3.10
Multipart	Table	[@tbl-scriv32A]	Table 3.10a
Multipart	Table	[@tbl-scriv32B]	Table 3.10b

Table 3.1: Cross-referencing amsthm theorems, diagrams, figures, listings, tables, and sections.

## 3.1 Amsthm

In this section, we are demonstrating the cross-referencing mechanism working with **Amsthm** theorems. First, we will see all of the theorems created using **Paragraph Styles**, then they will be introduced again as **Section Types**. In the table below, you'll see several Character Styles (labeled as Crossref...) used to reference both.

Element	Markdown Source	Rendered Output
Conjecture	[@cnj-scriv4]	Conjecture <a href="#">3.1.1</a>
Conjecture	[@cnj-scriv5]	Conjecture <a href="#">3.1.2</a>
Corollary	[@cor-scriv4]	Corollary <a href="#">3.1.1</a>
Corollary	[@cor-scriv6]	Corollary <a href="#">3.1.2</a>
Definition	[@def-scriv4]	Definition <a href="#">3.1.1</a>
Definition	[@def-scriv7]	Definition <a href="#">3.1.2</a>
Example	[@exm-scriv4]	Example <a href="#">3.1.1</a>
Example	[@exm-scriv8]	Example <a href="#">3.1.2</a>
Exercise	[@exr-scriv4]	Exercise <a href="#">3.1.1</a>
Exercise	[@exr-scriv9]	Exercise <a href="#">3.1.2</a>
Lemma	[@lem-scriv4]	Lemma <a href="#">3.1.1</a>
Lemma	[@lem-scriv10]	Lemma <a href="#">3.1.2</a>
Proposition	[@prp-scriv4]	Proposition <a href="#">3.1.1</a>
Proposition	[@prp-scriv11]	Proposition <a href="#">3.1.2</a>
Theorem	[@thm-scriv4]	Theorem <a href="#">3.1.1</a>
Theorem	[@thm-scriv12]	Theorem <a href="#">3.1.3</a>

Table 3.2: Cross-referencing amsthm elements in ScrivQ.

### Paragraph Styles

**Conjecture 3.1.1.** *Conjecture*

**Corollary 3.1.1.** *Corollary*

**Definition 3.1.1.** Definition

**Example 3.1.1.** Example

**Exercise 3.1.1.** Exercise

**Lemma 3.1.1.** *Lemma*

**Proposition 3.1.1.** *Proposition*

**Theorem 3.1.1** (Pythagorean theorem).

$$[x^2 + y^2 = z^2]$$

## Section Types

**Conjecture 3.1.2.** *Conjecture*

**Corollary 3.1.2.** *Corollary*

**Definition 3.1.2.** Definition

**Example 3.1.2.** Example

**Exercise 3.1.2.** Exercise

**Lemma 3.1.2.** *Lemma*

**Proposition 3.1.2.** *Proposition*

**Theorem 3.1.2** (Pythagorean theorem).

**Theorem 3.1.3.**

$$[x^2 + y^2 = z^2]$$

## 3.2 Diagrams

Let us see how we can use **raw markup**, **Section Types**, and **Paragraph Styles** to create **Dot** and **Mermaid** diagrams.

Element	Markdown Source	Rendered Output
Diagram Dot	[@fig-scriv14]	Figure 3.2
Diagram Dot	[@fig-scriv14B]	Figure 3.3
Diagram Dot	[@fig-scriv15]	Figure 3.4
Diagram Mermaid	[@fig-scriv16]	Figure 3.5
Diagram Mermaid	[@fig-scriv16B]	Figure 3.6
Diagram Mermaid	[@fig-scriv17]	Figure ??

Table 3.3: Cross-referencing Dot and Mermaid diagrams.

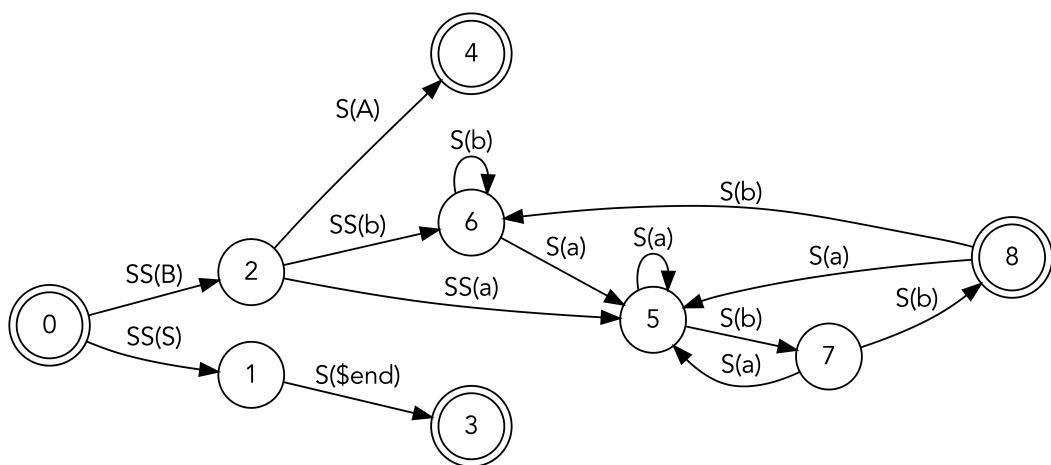


Figure 3.2: Figure caption

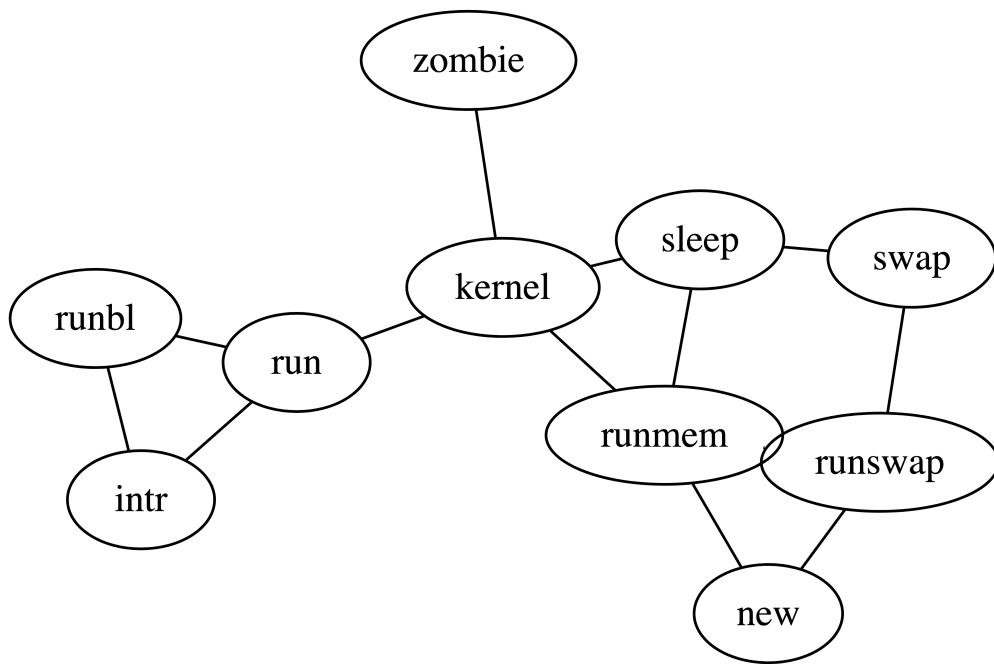


Figure 3.3: A graphviz graph with figure reference and caption, using raw markup. Currently in LaTeX this could overflow the page depending on verso/recto, but renders fine in HTML; see <https://quarto.org/docs/authoring/diagrams.html#sizing> for more details...

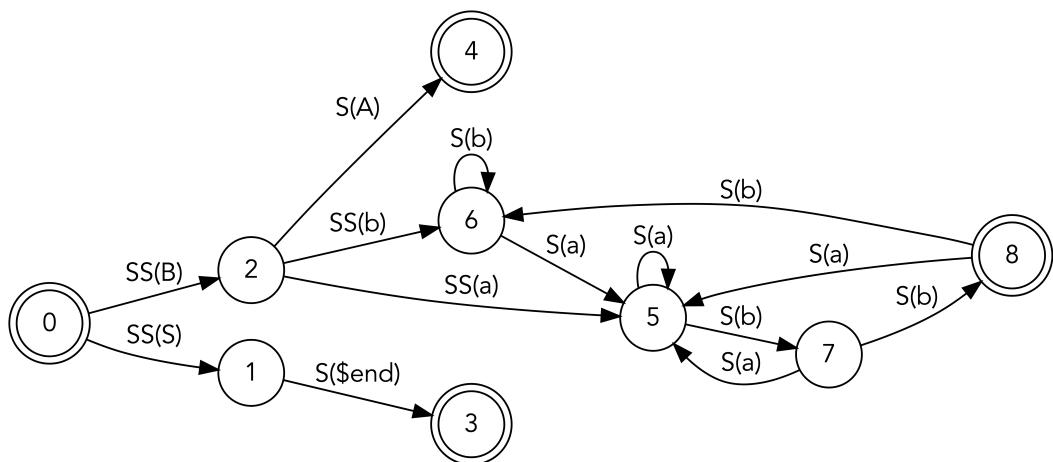


Figure 3.4: A Graphviz-generated state machine diagram

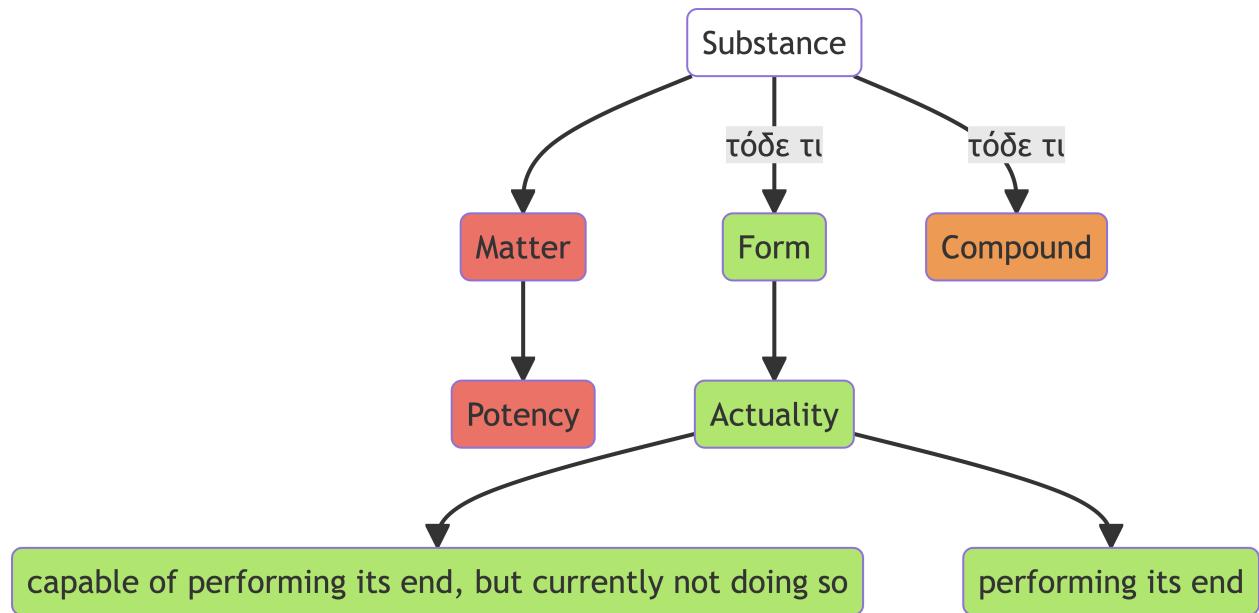


Figure 3.5: Figure caption

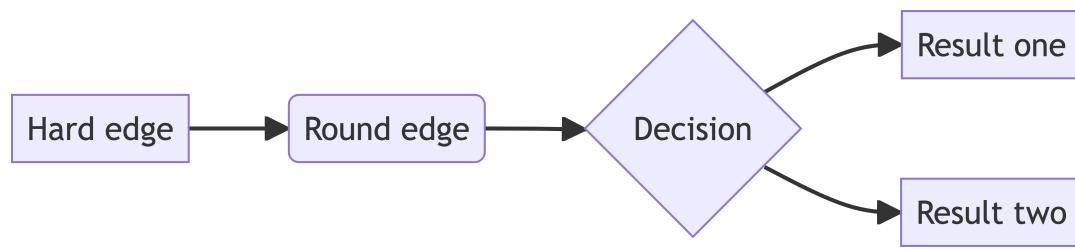
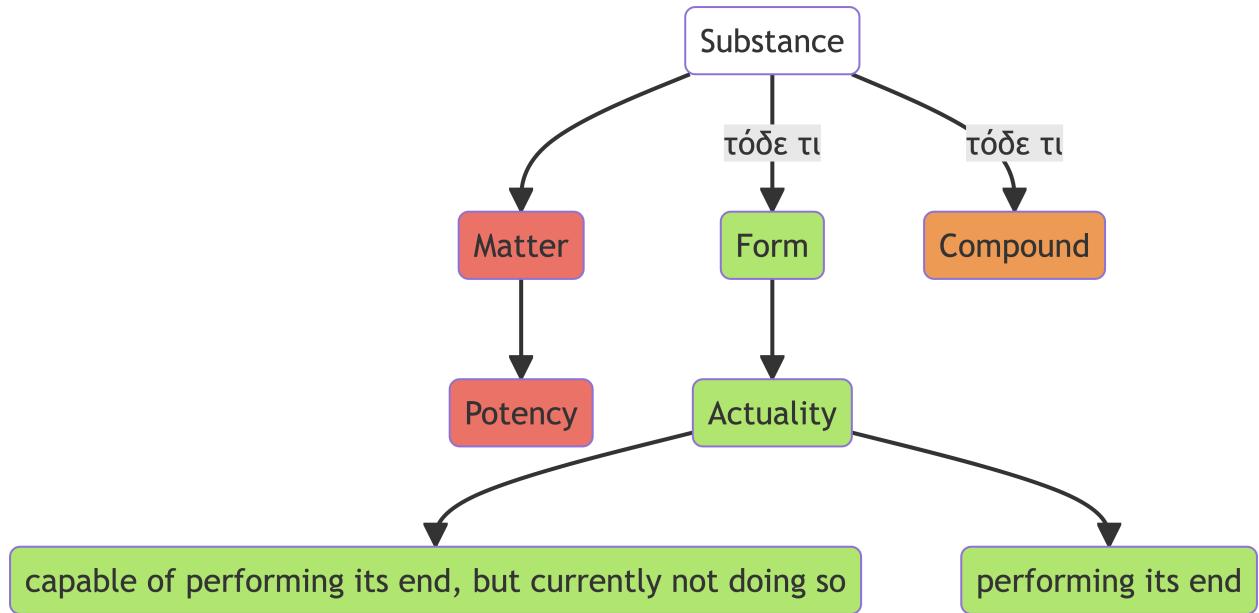


Figure 3.6: Figure caption



A Mermaid figure using a Scrivener Section Type [Diagram Mermaid], see <https://quarto.org/docs/authoring/diagrams.html> for more details

### 3.3 Equations

Element	Markdown Source	Rendered Output
Equation	[@eq-scriv19]	Equation 3.1
Equation	[@eq-scriv20]	Equation 3.2

Table 3.4: Cross-referencing equations.

$$t' = \frac{t - \frac{v}{c^2}x}{\sqrt{1 - \frac{v^2}{c^2}}} \quad (3.1)$$

$$t' = \frac{t - \frac{v}{c^2}x}{\sqrt{1 - \frac{v^2}{c^2}}} \quad (3.2)$$

## 3.4 Figures

Element	Markdown Source	Rendered Output	
Figure	[@fig-scriv22]	Figure 3.7	
Figure (Multipart)	[@fig-scriv23]	Figure 3.8	
Figure (Multipart)	[@fig-scriv23A]	Figure 3.8a	
Figure (Multipart)	[@fig-scriv23B]	Figure 3.8b	
Figure (Multipart)	[@fig-scriv24]	Figure 3.9	
Figure (Multipart)	[@fig-scriv24A]	Figure 3.9a	
Figure (Multipart)	[@fig-scriv24B]	Figure 3.9b	

Table 3.5: Cross-referencing figures.

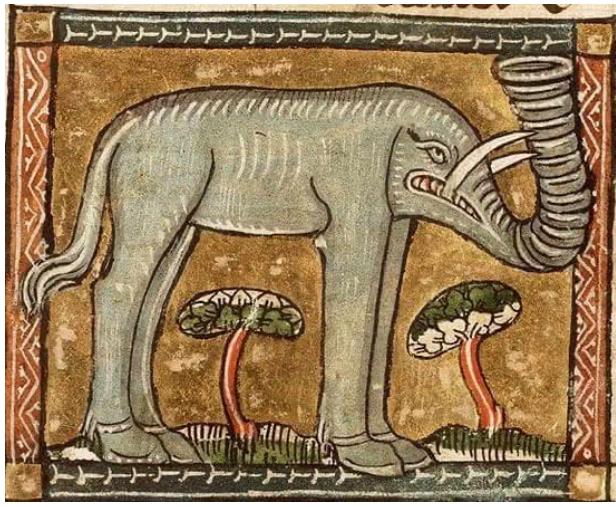


Figure 3.7: This figure uses custom metadata values to identify the class, ID, width and height. The %CA% (Caption Attributes) tag at the start of the caption is replaced with the correct Scrivener placeholders by the compiler; see global replacements for the details...

“I propose a toast, to my self-control. You see it helpless, crawling on the floor.” Morphine, *Cure For Pain* (1993)



(a) Elephant.

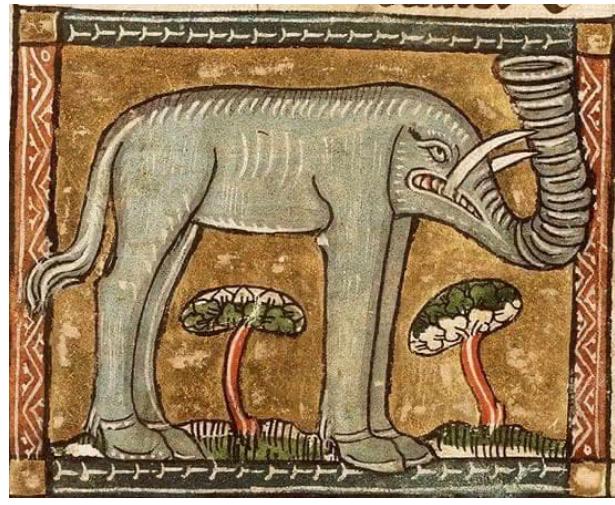


(b) Angry elephant with a big trunk.

Figure 3.8: This demonstrates generating a multi-panel figure using a Scrivener Section Type [Multipart Figure] instead of using raw markdown as shown here. ID, Class, and Attributes specific to the block [`#fig-elephants2 .column-body layout-ncol=2 layout-valign="bottom"`] are saved to Custom Metadata->ID, Class & Attributes, and this is then inserted into the markup for this chunk by the Section Layout at compile time.



(a) Elephant castle.



(b) Angry elephant with a big trunk.

Figure 3.9: Quarto allows the creation of figure panels with sub-figures. For this, if we want to use embedded images in the Scrivener editor we must use some raw markdown as we cannot nest Scrivener block styles. Note we can use the Scale Image... Tool in Scrivener and these sizes get exported to Quarto and the output. Here we scale both images to the same height.

## 3.5 Listings

Element	Markdown Source	Rendered Output
Listing	[@lst-scriv26]	<a href="#">Listing 3.1</a>
Listing	[@lst-scriv27]	<a href="#">Listing 3.2</a>

Table 3.6: Cross-referencing listings.

---

**Listing 3.1** Ruby code block. The listing Paragraph Style uses the custom metadata of the current text document.

---

```

1 require "unicode/name"
2
3 characters = %w(
4   )
5
6 # characters = ''
7 characters.each do |character|
8   puts character.unpack('U*').map { |i|
9     "U+#{i.to_s(16).rjust(4, '0')}.upcase}""
10  }.join
11  puts Unicode::Name.of character
12 end

```

---



---

**Listing 3.2** The caption

---

```

1 #!/usr/bin/env ruby
2 # frozen_string_literal: false
3
4 Encoding.default_external = Encoding::UTF_8
5
6 Dir["#{__dir__}/Ruby/**/*.rb"].each do |file|
7   require_relative file
8 end

```

---

---

**Listing 3.3** My Ruby code block.

```

1 require "unicode/name"
2
3 characters = %w(
4
5 # characters = ''
6 characters.each do |character|
7   puts character.unpack('U*').map { |i|
8     "U+#{i.to_s(16).rjust(4, '0')}.upcase}"}
9   }.join
10  puts Unicode::Name.of character
11 end

```

---

## 3.6 Tables

Element	Markdown Source	Rendered Output
Table	[@tbl-scriv30]	Table <a href="#">3.8</a>
Table	[@tbl-scriv31]	Table <a href="#">3.9</a>
Table (Multipart)	[@tbl-scriv32]	Table <a href="#">3.10</a>
Table (Multipart)	[@tbl-scriv32A]	Table <a href="#">3.10a</a>
Table (Multipart)	[@tbl-scriv32B]	Table <a href="#">3.10b</a>

---

Table 3.7: Cross-referencing tables.

1	2	3
<hr/>		
4	5	6

Table 3.8: This table uses Text as the **Section Type**, and Table Caption as the **Paragraph Style** for the caption.

1	2	3
<hr/>		
4	5	6

Table 3.9: This is an example of Table as **Section Type**. The caption and the remaining attributes are added as part of the Section Type markup.

Element	Prefix	Markdown Source	Rendered Output
Equation A	eq	A	B
Equation A	eq	C	D
Listing A	lst	E	F

(a) First Table

Element	Prefix	Markdown Source	Rendered Output
Equation B	eq	A	B
Equation B	eq	C	D
Listing B	lst	E	F

(b) Second Table

Table 3.10: This is a markdown multi-table panel with two sub-tables generated using a Section Type [Multipart Table]. Note that Custom Metadata holds the cross-referencing label, layout class, and the attributes for this multipart table, which will be added by the Section Layout by the compiler, using the Scrivener placeholders: `<$custom:Class> <$custom:Attributes>`.

## 3.7 Sections

The text sections can be referenced with **Character Styles**, and created with **Paragraph Styles** or **Section Types**. As before, all of these receive automatic IDs.

Element	Markdown Source	Rendered Output
Section	[@sec-scriv33]	Section 3.7
Section	[@sec-scriv34]	Section 3.7.1
Section	[@sec-scriv36]	Section 3.7.2
Section	[@sec-scriv37]	Section 3.7.3
Section	[@sec-scriv38]	Section 3.7.4

Table 3.11: Cross-referencing sections.

Note that the unnumbered section cannot be referenced.

### 3.7.1 Section

This is an example of the Section section type.

## **Section {-}**

This is an example of the Section {-} section type.

### **3.7.2 Heading**

### **3.7.3 Heading + Break**

This is an example of the Heading + Break section type.

### **3.7.4 Section + Break**

This is an example of the Section + Break section type.

## 3.8 Footnotes

We can use also use a **Section Type** to create and a **Character Style** (Footnote) to reference footnotes using the standard identifier.<sup>5</sup>

There is one small caveat: the user has to remember to always add two empty spaces before each new **paragraph** in the footnote environment. One option we have not to worry about this is using the Footnote Text Paragraph Style.

---

<sup>5</sup> This strategy has the outstanding advantage of allowing us to use, that is right, you guessed it, **Paragraph Styles** and **Character Styles** in footnotes.

## 4 Citations

Part of the motivation behind ScrivQ comes from another project I co-developed: the [Cite Tools](#) extension for [Pandoc](#) and [Quarto](#).

As I was first starting to use [Citeproc](#), coming from the jurassic [BibTeX](#), I was exceptionally pleased with its speed and reliability. Apart from being *a lot* faster, it would produce the same output across all supported formats (which amounts to over 60). Out-of-the-box, however, it lacked support for really ordinary **BibTeX functionalities**, such as the ability to split the bibliography into multiple sections, or the ability to cite arbitrary fields of the references (*e.g.* using `citetitle`, `citeauthor`, `citefield`). It also lacked the interesting `backref` option afforded by [BibTeX](#) used in conjunction with [HyperRef](#) to create linked indexes of citations.



Figure 4.1: Cite Tool bundles together several lua filters to address complex bibliography demands while keeping the output consistent across formats. If you have suggestions for improvements or bug reports, please [open an issue at the citetools repository](#). Logo image generated with Dall-E using “*Enso-like round black and white painting with ancient greek war-ship with a man tied to the mast as prompt*”.

To step around these limitations, I started tinkering with existing filters available on GitHub (all of them by Albert Krewinkel), and co-developed Cite Field, to create **Cite Tools**, an extension for Quarto and Pandoc that allows the easy creation of a **Multipart Bibliography** (*e.g.* split in *primary* and *secondary* sources, see Figure 6.1), the citation of arbitrary fields of the references (see Figure 6.2)<sup>1</sup>, and the linking of each bibliography entry back to its in-text occurrences (see Figure 6.3)<sup>2</sup>. The filters are built-in, and the front matter is set up so that the necessary files are automatically created during compilation.

**A** **Deleting Cite Tools from ScrivQ will cause the compilation to fail.**

**💡** If you need to use **Cite Tools** in an ordinary Quarto project, use `quarto install extension bcdavasconcelos/citetools` to install it.

## 4.1 Multipart Bibliography

In many areas of research, the ability to split the bibliography into sections is a condition *sine qua non* for publishing. In the humanities, for example, there are usually *primary* and *secondary* sources. In philosophy, even, they can be very nuanced with sections dedicated to original sources, translations, commentaries, and so on. The **Section Type** titled **Multipart Bibliography** can be used to create as many new bibliography sections as necessary. Add the references that should print there to the text, and let it know in the custom metadata `<\\$custom:Attributes>` the format being used (*e.g.* `bib`, `yml`, `ris`; **no dot, just the extension**).

**💡** **Bibliography formats**

Speaking about formats, the most common bibliography formats are **CSL-YAML**, **CSL-JSON**, **BibTeX**, and **RIS**. Internally, **Pandoc** and **Quarto** use the **CSL (Citation Style Language)** to handle bibli-

---

<sup>1</sup>In the official nomenclature, CSL has variables, BibTeX has fields, and RIS has tags.  
As a general rule, we have stuck to the term fields.

<sup>2</sup>Linked glossaries can also easily be created by dressing them as bibliography.

ography, so **CSL-YAML** and **CSL-JSON** perform much better (up to 10 times faster) than older formats like **BibTeX** or **RIS** that will have to be converted by **Pandoc** before it can be understood.

ScrivQ provides all the data needed for the project to compile. Before you can use Citeproc on your projects, you will need to generate your bibliography data. In principle, nothing stops you from manually, or semi-manually, keeping a bibliography in Scrivener, but this is not very easy to manage if you have many projects sharing the same references. (Luckily, in this regard, Scrivener offers the best text comparison tools I can think of). The best alternative, it seems, is to rely on specialized software such as Zotero, Bookends, Bibdesk (also JabRef, Endnote, and **others**)<sup>3</sup>. These programs allow you to edit your bibliography and easily export it in the desired format, which can be copied and pasted to different Scrivener projects. Zotero even offers an API that can be used to download shared libraries by merely accessing a link, such as <https://api.zotero.org/groups/LibraryID/items?format=bibtex&limit=999> where LibraryID corresponds to the library's 7-digit code (visible in the middle of the library URL).

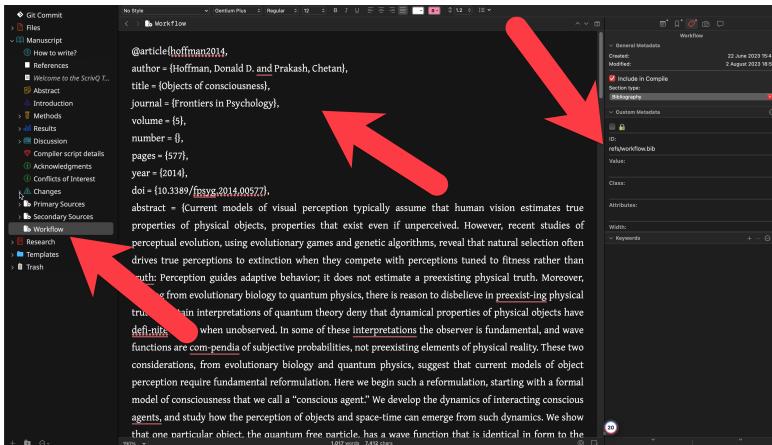


Figure 4.2: This will result in a separate file, whose path will be added to the front matter, with the content of the text. The resulting bibliography will print right where we placed it in our project.

**!** The sky is the limit

You can add as many bibliographies as you want!

<sup>3</sup>If you are using macOS, check Bookends and Bibdesk; and, on all platforms, definitely get Zotero as well.

## 4 Citations

3 Methods	4
3.1 Data Recording	4
3.2 Experimental Perturbations	5
3.3 Stimulus Plotting	6
3.4 Statistical Analysis	8
4 Results	11
4.1 Lunar Cycles	11
4.2 Solar Cycles	13
5 Discussion	15
6 Compiler script details	16
7 Acknowledgments	17
8 Conflicts of Interest	18
9 Changes	19
9.1 Cross-referencing	20
9.1.1 cnj, cor, dxxm, ext, lmm	21
9.1.2 eq, lst, fig, tbl	22
9.1.3 fns	22
9.2 CiteTools Integration	24
9.2.1 Bibliography	24
9.2.2 Cite-Field	24
9.2.3 Believe it or not, another can be an interesting	25
place to store your...	26
9.2.4 What other interesting possibilities to	26
control...	26
9.3 Resources	27
9.4 Shortcodes	27
10 Primary Sources	29
11 Secondary Sources	30
12 Workflow	31

## 12 Workflow

- Barrett, LF, and WK Simmons. 2015. "Interoceptive Predictions in the Brain." *Nature Reviews Neuroscience* 16 (7): 419–29. <https://doi.org/10.1038/nrn3950>. [ $\leftarrow$ , 2, 3, 3, 31, 15]
- Copenhaver, Rebecca. 2014. "Berkeley on the Language of Nature and the Objects of Vision." *Res Philosophica* 91 (1): 29–46. [ $\leftarrow$ , 3]
- Crivellato, Enrico, and Domenico Ribatti. 2007. "Soul, Mind, Brain: Greek Philosophy and the Birth of Neuroscience." *Brain Research Bulletin* 71 (4): 327–36. <https://doi.org/10.1016/j.brainresbull.2006.09.020>. [ $\leftarrow$ , 2, 2, 2, 3, 11, 15]
- Hoffman, Donald D., and Chetan Prakash. 2014. "Objects of Consciousness." *Frontiers in Psychology* 5: 577. <https://doi.org/10.3389/fpsyg.2014.00577>. [ $\leftarrow$ , 3]
- Siegel, Susanna, and Nicholas Silins. 2015. "The Epistemology of Perception." In *Oxford Handbook of Philosophy of Perception*, edited by Mohan Matthen, 1–48. Oxford University Press. [ $\leftarrow$ , 3, 4, 5, 5, 10, 11, 15, 15, 15, 15]
- Simmons, Alison. 2013. "Perception in Early Modern Philosophy." In *The Oxford Handbook of Philosophy of Perception*, edited by Mohan Matthen. Oxford: Oxford University Press. [ $\leftarrow$ , 3]

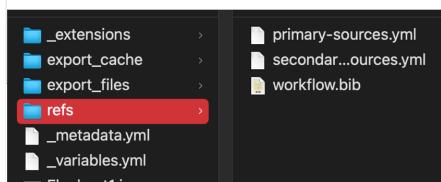


Figure 4.3: On the left we have the **Table of Contents**; on the top-right, we see the printed bibliography (page 31) and, on the bottom-right, the files are automatically created during the compilation process.

### 💡 Super tip: Making use of the Templates folder

Thanks to **ScriQ**, there is no need to keep separate bibliography files in the system, as the data can simply be copied and pasted from the bibliography managers to Scrivener. However, if you already have many bibliography files ready that you would like to use, it could be a good idea to use shared Templates folder to bring them to your fingertips inside Scrivener.

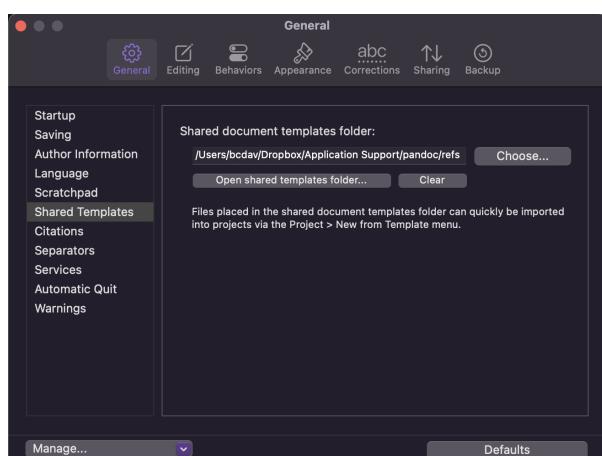


Figure 4.4: The shared templates folder in the main Scrivener configuration window.

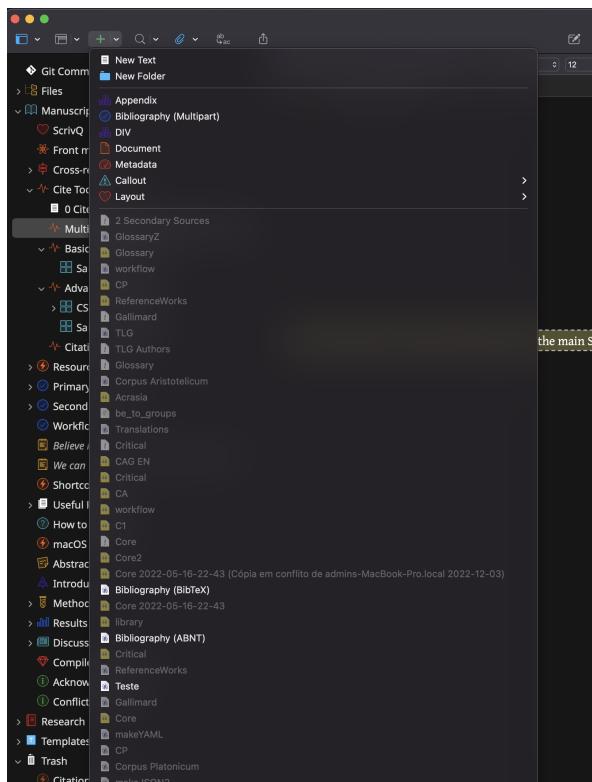


Figure 4.5: The Add new document menu in the Toolbar.

**⚠** For some reason, only .txt files are accepted, so the files must have this extension. Hopefully, L&L will change this to allow other plain text formats, such as md, qmd, yml, tex, ris, and so on.

### **i** From lists of names to the formatted bibliography

We are feeding the bibliography data to the most natural place, that is, precisely where it should be printed. In this case, however, we do it using any structured format we want (bib, ris, yml) unencumbered of the task of formatting the bibliography based on (some random set of) rules (set by some pissed off editor), which Citeproc will do for us using the provided CSL-style file (or the default one, which is Chicago, if none is provided).

## 4.2 Basic citations

### i Official documentation

See the official documentation on citations at [Pandoc](#) and [Quarto](#).

Let us quickly recapitulate the basics of Pandoc Citeproc and how it uses citations.

### i Citekeys

As we know, each citation must have a key, composed of @ + the citation identifier that must begin with a letter, digit, or \_, and may contain alphanumerics, \_, and internal punctuation characters (: . # \$ % & - + ? < > ~ /).

The citation syntax is very simple: @Citekey for **Author (Date)** (an *in-text* citation); [@Citekey] for **(Author, Date)**; and [-@Citekey] for **(Date)**. Multiple citations can be grouped in the same brackets separated by semicolons [@CitekeyA; @CitekeyB]. The citation key is optionally followed by a locator, which can be a page number, a line number, a chapter number, or a section number, preceded by a comma.

Markdown Source	Rendered output
@Long2004	Long ( <a href="#">2004</a> )
[@Long2004]	( <a href="#">Long 2004</a> )
[@Long2004, p. 15]	( <a href="#">Long 2004, 15</a> )
[-@Long2004]	( <a href="#">2004</a> )
[-@Long2004, p.15]	( <a href="#">2004, 15</a> )

Table 4.1: Citation syntax in Quarto and Pandoc.

### i (Date)

...on the deliberations of the prudent person ([2004](#)).

... on the deliberations of the prudent person [-@Long2004].

...on the deliberations of the prudent person ([2004, 17](#)).

... on the deliberations of the prudent person [-@Long2004, p.17].

**i** Author (Date)

Long (2004) says that...  
@Long2004 says that...

**i** (Author, Date)

...on the deliberations of the prudent person (Long 2004).  
... on the deliberations of the prudent person  
[@Long2004].  
...on the deliberations of the prudent person (Long 2004, 17).  
... on the deliberations of the prudent person  
[@Long2004, p.17].

**i** (Author, Date; Author, Date)

...on the deliberations of the prudent person (Long 2004; Hoffman and Prakash 2014).  
... on the deliberations of the prudent person  
[@Long2004; @hoffman2014].  
...on the deliberations of the prudent person (Long 2004, 17; Hoffman and Prakash 2014, 15).  
... on the deliberations of the prudent person  
[@Long2004, p.17; @hoffman2014, p.15].

That is pretty much all there is to it. Now that we have the basics covered, let us see what **Cite Field** can do for us.

### 4.3 Advanced citations

**i** TLDR

Several **Character Styles** are available to inject the correct markup ([@Citekey]{.csl\_field}) to cite specific fields from your references.

In many areas, we are frequently invited to comment on different editions and translations of the same classical works. In such cases, we refer not only to the author and the date issued of a publication, but also to its editor, translator, publisher, and even original-title and

edition. But how to do this? With **Cite Tools** enabled, the answer lies in a small variation of Pandoc's vanilla syntax for citations.

CSL Field	Markdown Source	Output
Author	[@DA]{.author}	Aristotelis
Editor	[@DA]{.editor}	Bekker
Issued	[@DA]{.issued}	1834
Original-title	[@DA]{.original-title}	περὶ ψυχῆς
Publisher	[@DA]{.publisher}	Reimer
Publisher-place	[@DA]{.publisher-place}	Berlin
Title	[@DA]{.title}	<i>De Anima</i>
Title-short	[@DA]{.title-short}	<i>De An.</i>
Translator	[@DA]{.translator}	Tatáxης

Table 4.2: All ready-made **Character Styles** for the Cite Field lua filter.

As we said, internally, Pandoc uses the **Citation Style Language** format for bibliographies. This means that **we must use the CSL variable names** (see Table 4.3), and not necessarily the field name you may see in a **RIS** or **BibTeX** bibliography. The correct way to print the book title, for example, would be `[@Citekey]{.container-title}` (and **not** using the BibTeX alternative which is `booktitle`).

```
The works of [@AristOp]{.author} were first
edited by [@AristOp]{.editor} in [@AristOp]{.issued}.
Later, the [@DA]{.title} was edited by [@DABiehl]{.editor}
in [@DABiehl]{.issued} (reprinted in [@DATheiler]{.translator}'s
[@DATheiler]{.issued} translation).
```

The works of Aristotelis were first edited by Bekker in 1834.  
 Later, the *De Anima* was edited by Biehl in 1896 (reprinted  
 in Theiler's 1995 translation).

CSL variables	BibTeX Fields	RIS Tags
abstract	abstract	AB
author	authors	AU A1

CSL variables	BibTeX Fields	RIS Tags
call-number	library	ID
chapter-number	chapter number issue	IS
collection-number		
number issue		
collection-title	series	-
container-title	booktitle journal	BT T2 JA JF JO
DOI	doi	DO
editor	editors	A2 ED
genre	type	-
ISSN	issn	SN
issued	date	PY Y1
keywords	keywords	KW
language	langid	LA
number-of-volumes	volumes	NV
original-title	origtitle	OR*
page	pages	SP EP
publisher	publisher school	PB
institution		
organization		
howpublished		
publisher-place	address	PP
title	title	TI T1 CT
title-short	shorttitle	ST*
url	URL	UR LK
version	version	-
volume	volume	VL

Table 4.3: CSL-YAML/CSL-JSON variables alongside corresponding BibTeX fields and RIS tags. Those marked with an asterisk exist and correspond, but, for some reason, Pandoc ignores them instead of converting to CSL.

## 4.4 Citation Backlinks

With Pandoc Citeproc, you can use link-citations to control whether citations in the body of the text should be clickable links to the reference in the bibliography (e.g. [ @EN ]). This is a very useful feature, especially

**BibTeX**

```

1 @book{AristOp,
2   author = {Aristotle},
3   editor = {Bekker, Immanuel},
4   title = {Aristotelis opera},
5   publisher = {Reimer},
6   address = {Berlim},
7   volumes = {4},
8   edition = {1},
9   year = {1831}
10 }
```

**RIS**

```

1 TY - BOOK
2 ID - AristOp
3 AU - Aristotle
4 ED - Bekker, Immanuel
5 TI - Aristotelis opera
6 PB - Reimer
7 CY - Berlim
8 ET - 1
9 VL - 4
10 Y1 - 1831
11 ER -
```

**CSL-YAML**

```

1 ---
2 references:
3 - author:
4   - family: Aristotle
5     edition: 1
6     editor:
7       - family: Bekker
8         given: Immanuel
9       id: AristOp
10      issued: 1831
11      number-of-volumes: 4
12      publisher: Reimer
13      publisher-place: Berlim
14      title: Aristotelis opera
15      type: book
16 ---
```

when you want to quickly check the source of a citation without having to scroll through the whole text. **ScrivQ** takes this one step further with **Cite Tools** and adds, in a crescent ordinal fashion<sup>4</sup>, a backlink to each citation an entry has received in the document. This allows the reader to easily arrive at sections of the text where the same reference was discussed, quickly seeing with the array of backlinks, how many times each reference was used in the text.

#### Turning off undesired linking

You can set `link-fields` to false to avoid undesired linking when citing specific fields (Section 4.3).

#### **link-citations**

Hyperlink citations to the corresponding bibliography entries.

Defaults to false.

#### **link-fields**

Hyperlink citations that target specific CSL fields to the corresponding entries in the bibliography. If `link-citations` is true, this defaults to true.

#### **link-bibliography**

Hyperlink DOIs, PMCIDs, PMID, and URLs in bibliographies.

Defaults to true.

#### **lang**

Affects the bibliography tags. Defaults to en-US.

---

<sup>4</sup>In other output formats, such as PDF, the reader will see the page number instead of a crescent ordinal number.

# 5 Resources

There are several other incredible resources in ScrivQ. Seriously. I spent easily over one hundred hours building this template, adding useful, good, and pretty things to it (aren't the icons lovely?). There are still many undocumented developed features that will receive proper treatment in upcoming versions. Just to mention in passing, the next step will probably be documenting some of the already existing possibilities for editing Pandoc templates inside Scrivener. This is already possible and being done, (everything is there and it is already working).

- Bootstrap Icons - <https://icons.getbootstrap.com> - These are available in Quarto documents using the **Shortcode Font Awesome** style as in . (There is also **Shortcode Env**, **Shortcode Meta**, **Shortcode Var**).
- Writing in Scrivener (<https://github.com/iandol/scrivomatic#writing-in-scrivener>) is a must read.
- The Plain Person's Guide to Plain Text Social Science - <https://plain-text.co/index.html#introduction>
- Quarto Reference - <https://quarto.org/docs/reference/>
- The easiest way to publish to Github Pages: Render to docs
- Example of Quarto Book - [https://github.com/jjallaire/hopr/blob/master/\\_quarto.yml](https://github.com/jjallaire/hopr/blob/master/_quarto.yml)
- Quarto with GH Pages - <https://tarleb.com/posts/quarto-with-gh-pages/>

## 5.1 Callout

These sections are divs with hardcoded classes (.callout-caution, .callout-important, .callout-note, .callout-tip, .callout-warning).

### Callout Caution

**!** Callout Important

**i** Callout Note

**?** Callout Tip

**⚠** Callout Warning

## 5.2 Layout

The **Section Type** [Column Page] adds the homonymous class to make the content much wider, though stopping short of extending across the whole document. See <https://quarto.org/docs/authoring/article-layout.html> for details. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

The contents will be assigned the `.column-page-right` class and stretched rightwards across the page, see <https://quarto.org/docs/authoring/article-layout.html> for details. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

The contents will be assigned the `.column-page-left` class and stretched leftwards across the page, see <https://quarto.org/docs/authoring/article-layout.html> for details.

This is an example of the Column Screen section type. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

## 5.3 Generic Divs

Finally, we'll look at how we can use generic Div sections to recreate some of the other hardcoded sections.

Some people might understandably prefer to achieve with fewer **Section Types** the same functionalities afforded by the profusion we saw earlier. To give an example, all Amsthm (Section 3.1) elements, the Multipart

This Marginalia is using a Section Type [Column Margin]. The contents will be assigned the `.column-margin` class and placed in the margin in HTML and LaTeX outputs. See <https://quarto.org/docs/authoring/article-layout.html> for details...

panels (Section 3.4, Section 3.6), the Callouts (Section 5.1), and the Column Layouts (Section 5.2) could be created from a single generic Div section. This is even made easier by the presence of the <\$custom:ID-Prefix>, <\$custom:Class>, and <\$custom:Class<sub>2</sub>> fields, which unen-cumbers the user from remembering the correct prefix for **Amsthm** or **Cross-reference** sections, and the classes for **Callouts** and different **Column Layout** options.

 Tip

For <\$custom:Width> and <\$custom:Height>, you can input the value *directly*. There is no need to add width= or place the value between quotes. This will automatically be done for you.

 Note

If the fields are left empty, the ruby script will remove the empty keys for us.

 Now that you have familiarized yourself with **ScrivQ**, I ask you to consider the following: this template was developed not by ChatGPT, or some other automated method, but by *an obscene amount of time and effort* from a real human being. So, ask yourself, “How much is it worth to me to employ this in my work?”, “How much is it worth it to me to always have an updated working version?”. Then, **I ask you to let the artisan know**. It makes a difference.

# 6 References

## 6.1 Primary Sources

Aristotelis. 1834a. *Aristotelis Opera*. Edited by I. Bekker. Berlin: Reimer.

[]

———. 1834b. “De Anima.” In *Aristotelis Opera*, edited by Immanuel Bekker, translated by B. Τατάκης. Berlin: Reimer. []

Plato. 1903. *Meno*. Edited by John Burnet. *Platonis Opera*. Oxford: OCT.

[]

## 6.2 Secondary Sources

Aristotelis. 1896. *De Anima*. Edited by Wilhelm Biehl. Leipzig: Teubner.

[]

———. 1995. *De Anima*. Edited by Wilhelm Biehl. Translated by Willy Theiler and Horst Seidl. Hamburg: Felix Meiner. []

Long, Christopher. 2004. *Ethics of Ontology*. SUNY Series in Ancient Greek Philosophy. Albany: SUNY. [, , , , , , , , , , , ]

## 6.3 Workflow

Hoffman, Donald D., and Chetan Prakash. 2014. “Objects of Consciousness.” *Frontiers in Psychology* 5: 577. <https://doi.org/10.3389/fpsyg.2014.00577>. [, , ]

## 6.4 Songs

Morphine, Mark Sandman, Dana Colley, and Jerome Deupree. 1993. *Cure For Pain*. CD. Cure For Pain. Rykodisc. <https://open.spotify.com/>

## *6 References*

[track/3hO9gaVixKDoYDrITBrEWf?si=o668baf1aab345d4.](#) []

# Cite Tools Samples

## Primary Sources

- Aristotle. 1831a. *Aristotelis Opera*. Edited by Immanuel Bekker. 4 vols. Berlim: Reimer. [1]  
———. 1831b. “De Anima.” In *Aristotelis Opera*, edited by Immanuel Bekker, 402a01–435b25. Berlim: Reimer.  
———. 1896. *De Anima*. Edited by Wilhelm Biehl. Leipzig: Teubner. [1]  
———. 1912. *Aristotelis de Anima Libri III*. Edited by Aurél Förster. Budapest: Hungarian Academy of Sciences. [1]  
———. 1961. *De Anima*. Edited by William David Ross. Oxford: Clarendon. [1]

## Secondary Sources

- Aristoteles. 2017. *Über Die Seele. De Anima*. Translated by Klaus Corcilius. Hamburg: Felix Meiner. [1, 2]  
Aristotelis. 1995. *De Anima*. Translated by H. Seidl and W. Theiler. Hamburg: Felix Meiner. [1]

Figure 6.1: Multipart bibliography with sections, such as primary sources and secondary sources

The works of [@AristOp]{.author} were first edited by  
[@AristOp]{.editor} in [@AristOp]{.issued}.

The works of Aristotle were first edited by Bekker in 1831.

Figure 6.2: Cite Field allows the evocation of arbitrary information from the references, such as author, editor, translator (using CSL variables name conventions)

## 6 References

manuel Bekker. 4 vols. Berlim: Reimer. [1, 2]  
i, edited by Immanuel Bekker, 402a01–435b25. Berlim:  
iehl. Leipzig: Teubner. [1, 2]  
ed by Aurél Förster. Budapest: Hungarian Academy of  
avid Ross. Oxford: Clarendon. [1, 2]

Figure 6.3: The **Citation Backlinks** filter adds an index of cited references to the bibliography, with links back to all in-text citations. It also allows the user to turn these off globally or in an *ad hoc* fashion.