

# Evaluating FAIR Digital Object as a distributed object system

This manuscript ([permalink](#)) was automatically generated from [stain/2022-fdo-paper@80c6399](#) on May 4, 2022.

## Authors

---

- **Stian Soiland-Reyes**

 [0000-0001-9842-9718](#) ·  [stain](#) ·  [soilandreyes](#)

Department of Computer Science, The University of Manchester, UK; Informatics Institute, Faculty of Science, University of Amsterdam, NL · Funded by [BioExcel-2](#) (European Commission [H2020-INFRAEDI-2018-1 823830](#)); [BY-COVID](#) (European Commission [HORIZON-INFRA-2021-EMERGENCY-01 101046203](#))

- **Carole Goble**

 [0000-0003-1219-2137](#) ·  [carolegoble](#) ·  [CaroleAnneGoble](#)

Department of Computer Science, The University of Manchester, UK · Funded by [BioExcel-2](#) (European Commission [H2020-INFRAEDI-2018-1 823830](#)); [EOSC-Life](#) (European Commission [H2020-INFRAEOSC-2018-2 824087](#)); [BY-COVID](#) (European Commission [HORIZON-INFRA-2021-EMERGENCY-01 101046203](#))

- **Paul Groth**

 [0000-0003-0183-6910](#) ·  [pgroth](#) ·  [pgroth](#)

Informatics Institute, Faculty of Science, University of Amsterdam, NL

# Abstract

---

This manuscript is a template (aka “rootstock”) for [Manubot](#), a tool for writing scholarly manuscripts. Use this template as a starting point for your manuscript.

The rest of this document is a full list of formatting elements/features supported by Manubot. Compare the input ( `.md` files in the `/content` directory) to the output you see below.

## Basic formatting

---

### Bold text

### Semi-bold text

Centered text

Right-aligned text

*Italic text*

Combined *italics* and **bold**

~~Strikethrough~~

1. Ordered list item
2. Ordered list item
  - a. Sub-item
  - b. Sub-item
    - i. Sub-sub-item
3. Ordered list item
  - a. Sub-item

- List item
- List item
- List item

subscript: H<sub>2</sub>O is a liquid

superscript: 2<sup>10</sup> is 1024.

[unicode superscripts](#)<sup>0123456789</sup>

[unicode subscripts](#)<sub>0123456789</sub>

A long paragraph of text. Lorem ipsum dolor sit amet, consectetur adipiscing 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. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Putting each sentence on its own line has numerous benefits with regard to [editing](#) and [version control](#).

Line break without starting a new paragraph by putting two spaces at end of line.

## Document organization

---

Document section headings:

# Heading 1

## Heading 2

---

### Heading 3

#### Heading 4

##### Heading 5

###### Heading 6

**A heading centered on its own printed page**

Horizontal rule:

---

Heading 1's are recommended to be reserved for the title of the manuscript.

Heading 2's are recommended for broad sections such as *Abstract*, *Methods*, *Conclusion*, etc.

Heading 3's and Heading 4's are recommended for sub-sections.

## Links

---

Bare URL link: <https://manubot.org>

[Long link with lots of words and stuff and junk and bleep and blah and stuff and other stuff and more stuff yeah](#)

[Link with text](#)

[Link with hover text](#)

[Link by reference](#)

## Citations

---

Citation by DOI [[1](#)].

Citation by PubMed Central ID [[2](#)].

Citation by PubMed ID [[3](#)].

Citation by Wikidata ID [[4](#)].

Citation by ISBN [[5](#)].

Citation by URL [[6](#)].

Citation by alias [[7](#)].

Multiple citations can be put inside the same set of brackets [[1](#),[5](#),[7](#)]. Manubot plugins provide easier, more convenient visualization of and navigation between citations [[2](#),[3](#),[7](#),[8](#)].

Citation tags (i.e. aliases) can be defined in their own paragraphs using Markdown's reference link syntax:

## Referencing figures, tables, equations

---

Figure [1](#)

Figure [2](#)

Figure [3](#)

Figure [4](#)

Table [1](#)

Equation [1](#)

Equation [2](#)

## Quotes and code

---

Quoted text

Quoted block of text

Two roads diverged in a wood, and I—  
I took the one less traveled by,  
And that has made all the difference.

Code `in the middle` of normal text, aka `inline code`.

Code block with Python syntax highlighting:

```
from manubot.cite.doi import expand_short_doi

def test_expand_short_doi():
    doi = expand_short_doi("10/c3bp")
    # a string too long to fit within page:
    assert doi == "10.25313/2524-2695-2018-3-vliyanie-enhansera-copia-i-
        insulyatora-gypsy-na-sintez-ernk-modifikatsii-hromatina-i-
        svyazyvanie-insulyatornyh-belkov-vtransfetsirovannyh-geneticheskikh-
        konstruktsiyah"
```

Code block with no syntax highlighting:

```
Exporting HTML manuscript
Exporting DOCX manuscript
Exporting PDF manuscript
```

## Figures

---



**Figure 1: A square image at actual size and with a bottom caption.** Loaded from the latest version of image on GitHub.



**Figure 2: An image too wide to fit within page at full size.** Loaded from a specific (hashed) version of the image on GitHub.



**Figure 3: A tall image with a specified height.** Loaded from a specific (hashed) version of the image on GitHub.



**Figure 4: A vector `.svg` image loaded from GitHub.** The parameter `sanitize=true` is necessary to properly load SVGs hosted via GitHub URLs. White background specified to serve as a backdrop for transparent sections of the image.

## Tables

**Table 1:** A table with a top caption and specified relative column widths.

<i>Bowling Scores</i>	Jane	John	Alice	Bob
Game 1	150	187	210	105
Game 2	98	202	197	102
Game 3	123	180	238	134

**Table 2:** A table too wide to fit within page.

	Digits 1-33	Digits 34-66	Digits 67-99	Ref.
pi	3.14159265358979323846264338327950	288419716939937510582097494459230	781640628620899862803482534211706	<a href="#">piday.org</a>
e	2.71828182845904523536028747135266	249775724709369995957496696762772	407663035354759457138217852516642	<a href="#">nasa.gov</a>

**Table 3:** A table with merged cells using the `attributes` plugin.



	Colors	
Size	Text Color	Background Color
big	blue	orange
small	black	white

## Equations

A LaTeX equation:

$$\int_0^\infty e^{-x^2} dx = \frac{\sqrt{\pi}}{2} \tag{1}$$

An equation too long to fit within page:

$$x = a + b + c + d + e + f + g + h + i + j + k + l + m + n + o + p + q + r + s + t + u + v + w + x + y + z + 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 \tag{2}$$

## Special

**⚠ WARNING** *The following features are only supported and intended for `.html` and `.pdf` exports. Journals are not likely to support them, and they may not display correctly when converted to other formats such as `.docx`.*

LINK STYLED AS A BUTTON

Adding arbitrary HTML attributes to an element using Pandoc's attribute syntax:

Manubot Manubot Manubot Manubot Manubot. Manubot Manubot Manubot Manubot. Manubot Manubot Manubot. Manubot Manubot. Manubot.

Adding arbitrary HTML attributes to an element with the Manubot `attributes` plugin (more flexible than Pandoc's method in terms of which elements you can add attributes to):

Manubot Manubot Manubot Manubot Manubot. Manubot Manubot Manubot Manubot. Manubot Manubot Manubot. Manubot Manubot. Manubot.

Available background colors for text, images, code, banners, etc:

white

lightgrey

grey

darkgrey

black

lightred

lightyellow

lightgreen

lightblue

lightpurple

red

orange

yellow

green

blue

purple

Using the [Font Awesome](#) icon set:

✓

?

★

🔔

✖

⋮

### Light Grey Banner

useful for *general information* - [manubot.org](http://manubot.org)

### Blue Banner

useful for *important information* - [manubot.org](http://manubot.org)

### Light Red Banner

useful for *warnings* - [manubot.org](http://manubot.org)

## Interoperability Framework for Fast Data

---

Quotes from [9]:

- **Symbiotic** (purpose and intent): Motivations to have the interaction, with varying levels of mutual knowledge of governance, strategy and goals.
- **Pragmatic** (reaction and effects): Management of the effects of the interaction at the levels of choreography, process and service e
- **Semantic** (meaning of content): *Inference Rule base*. Interpretation of a message in context, at the levels of rule, known application components and relations and definition of concepts
- **Syntactic** (notation of representation): \_Representation of application components, in terms of composition, primitive components and their serialization format in messages
- **Connective** (transfer protocol): Lower-level formats and network protocols involved in transferring a message from the context of the sender to that of the receiver
- **Environmental** (deployment and migration) Environment in which each application is deployed and managed, including the portability problems raised by migrations

**Symbiotic:** Expresses the purpose and intent of two interacting applications to engage in a mutually beneficial agreement. Enterprise engineering is usually the topmost level in application interaction complexity, since it goes up to the human level, with governance and strategy heavily involved. Therefore, it maps mainly onto the symbiotic category, although the same principles apply (in a more rudimentary fashion) to simpler subsystems. This can entail a tight coordination under a common governance (if the applications are controlled by the same entity), a joint venture agreement (if the two applications are substantially aligned), a collaboration involving a partnership agreement (if some goals are shared) or a mere value chain cooperation (an outsourcing contract).

**Pragmatic:** The effect of an interaction between a consumer and a provider is the outcome of a contract, which is implemented by a choreography that coordinates processes, which in turn implement workflow behaviour by orchestrating service invocations. Languages such as Business Process Execution Language (BPEL ) [31] support the implementation of processes and Web Services Choreography Description Language (WS-CDL) is an example of a language that allows choreographies to be specified.

**Semantic:** Both interacting applications must be able to understand the meaning of the content of the messages exchanged: both requests and responses. This implies interoperability in rules, knowledge and ontologies, so that meaning is not lost when transferring a message from the context of the sender to that of the receiver. Semantic languages and specifications such as Web Ontology Language (OWL ) and Resource Description Framework (RDF), map onto this category.

**Syntactic:** This deals mainly with form, rather than content. Each message has a structure composed of data (primitive applications) according to some structural definition (its schema). Data need to be serialized to be sent over the network as messages using representations such as XML or JSON .

**Connective:** The main objective is to transfer a message from the context of one application to the other regardless of its content. This usually involves enclosing that content in another message with control information and implementing a message protocol (such as SOAP or HTTP) over a communications network according to its own protocol (such as TCP/IP) and possibly resorting to routing gateways if different networks are involved.

**Environmental:** Each application also interacts with the environment (e.g. a cloud or a server) in which it is deployed, anewed or by migration. The environment's management application programming interface (API) and the infrastructure level that the application requires will most likely have impact on the way applications interact, particularly if they are deployed in (or migrate between) different environments, from different vendors. Interoperability between an application and the environment in which it is deployed usually known as portability.

**Metamodel:** Resource, Service, Transaction, Process, Response, Operation, Request, Channel, Protocol, Link

## A comparison framework for middleware infrastructures

---

Quotes from [\[10\]](#):

- **Openness:** The middleware infrastructure should enable extending the applications built on top of it in various ways. (e.g., adding, removing, upgrading, composing services, etc.).
- **Scalability:** The middleware infrastructure should facilitate the effective operation of the applications at many different scales.
- **Performance:** The middleware infrastructure should enable the efficient and predictable, if needed, execution of the applications that are built on top of it.
- **Distribution transparency:** is the property that determines if the application is perceived by users, or developers as a whole rather than as a collection of independent constituent elements. The requirement for distribution transparency is quite generic and it is usually refined into a number of more specific transparencies including:
  - **Access transparency:** the infrastructure should enable accessing local and remote application elements in the same way.
  - **Location transparency:** the infrastructure should enable accessing the application elements without knowledge of their physical location.
  - **Concurrency transparency:** the infrastructure should allow concurrent processing on resources, without interference.
  - **Failure transparency:** the infrastructure should enable service provisioning despite the occurrence of failures.
  - **Migration transparency:** the infrastructure should provide means for changing the location of elements of the application without compromising the application's correct operation, i.e. without affecting the elements that depend on the migrated elements.
  - **Persistence transparency:** the infrastructure should provide means for concealing the deactivation and reactivation of elements from other elements that are using them.
  - **Transaction transparency:** the infrastructure should provide means for coordinating the execution of atomic and isolated transactions.

**Modularity:** The application should consist of a collection of elements, each one providing services, used by the others. Modularity enables the identification of dependencies between the elements that

make up the system. Consequently, it allows determining, which elements are affected by the eventual addition, removal or upgrade of services.

**Encapsulation:** For each constituent element, there is a clear separation between the element's interface and implementation. The interface is a well-defined specification of the provided services, the contract between the element and the entities using it. The implementation is the realization of the provided services. In general, it is safe to change the implementation of an element as long as the element's interface is preserved. Changing an element's interface without compromising the overall application integrity requires that the rest of the application does not depend on this particular interface, at the time of the change.

**Inheritance:** An interface specification (resp. implementation) may be derived from another one. The derived interface (resp. implementation) provides at least the services of the base interface (resp. implementation). Inheritance enables the vertical and horizontal composition of services.

**Signal interfaces:** defining asynchronous stimuli that can be handled by instances of engineering objects, providing these interfaces.

**Operation interfaces:** defining operations that can be invoked on instances, providing these interfaces. Invoking an operation causes a request message to be sent by the invoker to the invoked instance. Invoking an operation may further result in a reply sent from the invoked instance to the invoking instance.

**Stream interfaces:** defining operations that can be invoked on instances, providing these interfaces. The result of invoking a stream operation is the continuous conveyance of information from the invoked instance to the invoking instance.

# References

---

1. **Sci-Hub provides access to nearly all scholarly literature**  
Daniel S Himmelstein, Ariel Rodriguez Romero, Jacob G Levernier, Thomas Anthony Munro, Stephen Reid McLaughlin, Bastian Greshake Tzovaras, Casey S Greene  
*eLife* (2018-03-01) <https://doi.org/ckcj>  
DOI: [10.7554/elife.32822](https://doi.org/10.7554/elife.32822) · PMID: [29424689](https://pubmed.ncbi.nlm.nih.gov/29424689/) · PMCID: [PMC5832410](https://pubmed.ncbi.nlm.nih.gov/PMC5832410/)
2. **Reproducibility of computational workflows is automated using continuous analysis**  
Brett K Beaulieu-Jones, Casey S Greene  
*Nature biotechnology* (2017-04) <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6103790/>  
DOI: [10.1038/nbt.3780](https://doi.org/10.1038/nbt.3780) · PMID: [28288103](https://pubmed.ncbi.nlm.nih.gov/28288103/) · PMCID: [PMC6103790](https://pubmed.ncbi.nlm.nih.gov/PMC6103790/)
3. **Bitcoin for the biological literature.**  
Douglas Heaven  
*Nature* (2019-02) <https://www.ncbi.nlm.nih.gov/pubmed/30718888>  
DOI: [10.1038/d41586-019-00447-9](https://doi.org/10.1038/d41586-019-00447-9) · PMID: [30718888](https://pubmed.ncbi.nlm.nih.gov/30718888/)
4. **Plan S: Accelerating the transition to full and immediate Open Access to scientific publications**  
cOAlition S  
(2018-09-04) <https://www.wikidata.org/wiki/Q56458321>
5. **Open access**  
Peter Suber  
*MIT Press* (2012)  
ISBN: 9780262517638
6. **Open collaborative writing with Manubot**  
Daniel S Himmelstein, Vincent Rubinetti, David R Slochower, Dongbo Hu, Venkat S Malladi, Casey S Greene, Anthony Gitter  
*Manubot* (2020-05-25) <https://greenelab.github.io/meta-review/>
7. **Opportunities and obstacles for deep learning in biology and medicine**  
Travers Ching, Daniel S Himmelstein, Brett K Beaulieu-Jones, Alexandr A Kalinin, Brian T Do, Gregory P Way, Enrico Ferrero, Paul-Michael Agapow, Michael Zietz, Michael M Hoffman, ... Casey S Greene  
*Journal of The Royal Society Interface* (2018-04) <https://doi.org/gddkhn>  
DOI: [10.1098/rsif.2017.0387](https://doi.org/10.1098/rsif.2017.0387) · PMID: [29618526](https://pubmed.ncbi.nlm.nih.gov/29618526/) · PMCID: [PMC5938574](https://pubmed.ncbi.nlm.nih.gov/PMC5938574/)
8. **Open collaborative writing with Manubot**  
Daniel S Himmelstein, Vincent Rubinetti, David R Slochower, Dongbo Hu, Venkat S Malladi, Casey S Greene, Anthony Gitter  
*PLOS Computational Biology* (2019-06-24) <https://doi.org/c7np>  
DOI: [10.1371/journal.pcbi.1007128](https://doi.org/10.1371/journal.pcbi.1007128) · PMID: [31233491](https://pubmed.ncbi.nlm.nih.gov/31233491/) · PMCID: [PMC6611653](https://pubmed.ncbi.nlm.nih.gov/PMC6611653/)
9. **An Interoperability Framework and Distributed Platform for Fast Data Applications**  
José Carlos Martins Delgado  
*Data Science and Big Data Computing* (2016) <https://doi.org/gp3rds>  
DOI: [10.1007/978-3-319-31861-5\\_1](https://doi.org/10.1007/978-3-319-31861-5_1)
10. **A Comparison Framework for Middleware Infrastructures.**  
Apostolos Zarras  
*The Journal of Object Technology* (2004) <https://doi.org/cj5q8r>

DOI: [10.5381/jot.2004.3.5.a2](https://doi.org/10.5381/jot.2004.3.5.a2)