

Contributing to the AtoM Documentation

An introduction to AtoM's docs,
Sphinx, and reStructuredText



Dan Gillean
Artefactual Systems

Version 2.3 has not been released yet, and its documentation is a work in progress.

AtoM documentation

User manual

User guide to creating, editing, accessing, importing, exporting, and translating content in AtoM; includes administering the system via the user interface.

OVERVIEW

This section provides an overview of the technical architecture and requirements.

- What is AtoM?
- Technical requirements
- Entity types
- Descriptive standards

GETTING STARTED

This section describes how to start using AtoM once it has been installed. It provides a quick orientation to the home page, how to log in and log out, the various user roles in the system, the main page types a user will encounter, and how to switch languages.

- Home page
- Log in
- User roles
- Page types
- Choose language
- Change password

ADD/EDIT CONTENT

This section describes how to add, edit, and delete

Administrator manual

Instructions for installing, upgrading, importing and more; includes administering the system via the command-line interface.

INSTALLATION

- Intended audience
- Requirements
- Linux
- Windows
- Mac OS X
- Using Vagrant
- Multi-node deployment
- Upgrading
- PHP execution limits
- Asynchronous jobs and worker management

MAINTENANCE

- Command line tools
- Clear cache
- Web analytics
- Logging
- Monitoring
- Populate search index
- Data backup
- Tuning server parameters
- Troubleshooting
- Debug mode
- Elasticsearch

CUSTOMIZATION

Version 2.3

[\[Table of contents\]](#)

CONTENTS

AtoM documentation

- User manual
 - Overview
 - Getting started
 - Add/edit content
 - Access content
 - Translate
 - Reports/printing
 - Import/export
 - Administer
 - Data entry / templates
 - Glossary
- Administrator manual
 - Installation
 - Maintenance
 - Customization
 - Security

SEARCH

Open the general index or type your search in the search box.

AVAILABLE DOCUMENTS

- Version 2.0
- Version 2.1
- Version 2.2
- Version 2.3

OVERVIEW

Version 2.3 has not been released yet, and its documentation is a work in progress.

AtoM documentation

Version 2.3

[Table of contents]

CONTENTS

AtoM documentation

- User manual
 - Overview
 - Getting started
 - Add/edit content
 - Access content
 - Troubleshooting
 - Import/export
 - Administrator
 - Data entry / templates
- Administrator manual
 - Installation
 - Maintenance
 - Customization
 - Security

SEARCH

Open the general index or type your search in the search box.

API AND DOCUMENTATION

- Version 2.3
- Version 2.1
- Version 2.2
- Version 2.3

Three Manuals:

- **User manual:** Using the application via the user interface
- **Administrator's manual:** installation, maintenance, upgrades, customization, troubleshooting
- **Developer's manual:** API, development environments

AtoM documentation is **versioned** for each major release

ADD/EDIT CONTENT

This section describes how to add, edit, and delete

CUSTOMIZATION

OVERVIEW

AtoM WIKI DOCUMENTATION RESOURCES

On the [AtoM wiki](https://wiki.accesstomemory.org/):

- Information about the documentation
- How to contribute via GitHub
- Contribution guidelines

<https://wiki.accesstomemory.org/Resources/Documentation/>

The screenshot shows the AtoM Wiki website. At the top is the '@tom' logo. Below it is a navigation bar with links: Home, Downloads, Documentation, Community, Wiki, and Demo. The main content area is titled 'Documentation' and includes a breadcrumb trail: '< Resources' and 'Main Page > Resources > Resources/Documentation'. A paragraph states: 'On this page you'll find some information about our user documentation [🔗], including the User and Administrator Manuals found on our website for each major AtoM release.' Below this is a 'Seealso' section with two bullet points: 'Contribute code' and 'Contribute translations'. Further down are three sections: 'ABOUT OUR DOCUMENTATION [EDIT]' with a paragraph about Sphinx and a bullet point 'About our documentation'; 'HOW TO CONTRIBUTE TO OUR DOCUMENTATION [EDIT]' with a paragraph about basic instructions and a bullet point 'How to contribute to our documentation'; and 'DOCUMENTATION CONTRIBUTION GUIDELINES [EDIT]' with a paragraph about a tutorial and style guide and a bullet point 'Documentation contribution guidelines'. At the bottom, it says 'Category: Resources'.

@tom

Home Downloads Documentation Community Wiki Demo

Documentation

< Resources

Main Page > Resources > Resources/Documentation

On this page you'll find some information about our user documentation [🔗], including the User and Administrator Manuals found on our website for each major AtoM release.

Seealso

- Contribute code
- Contribute translations

ABOUT OUR DOCUMENTATION [EDIT]

Information on why we have chosen the Sphinx documentation platform to create our User and Administrator manuals

- About our documentation

HOW TO CONTRIBUTE TO OUR DOCUMENTATION [EDIT]

Basic instructions on the ways, both large and small, that you can help us maintain our project documentation!

- How to contribute to our documentation

DOCUMENTATION CONTRIBUTION GUIDELINES [EDIT]

A tutorial and style guide on some of the best practices we encourage when working with our documentation. Please check this out before contributing documentation!

- Documentation contribution guidelines

Category: Resources

FIRST, A STORY...

...eggen dat deze vroege, maar
meer lijkt op een...
obo's (of dwergsluipmuis) en de
er op de geweldadige...
eet. De kloof lijkt nauwelijks te...
In ieder geval hoop ik...
leunen bij de bonobo's. Ik doe het...
met iedereen, maar in mijn diepe...
krochten mijner ziel is een...
promiscuïteit toch niet ver...
niet gunstiger van pas komen...
sindsdien hebben bedacht en uitge...
volgens Montaigne. Volgens Montaigne...
volgens Montaigne. Volgens Montaigne...
sociale leven zijn hoogste vervulling...
vriendschap. Niet in de liefde, niet in de...
"Want", zegt hij, "alle banden die vanwege de...
lust, de voordelen of om een particulier...
belang gesmeed en onderhouden worden, zijn...
het algemeen minder mooi en edel, en...
vriendschappen te bestempelen in de mate...
er een ander motief, doel en voordeel dan de...
schap zelf een rol in speelt." Jammer toch dat...



Atom DEVELOPMENT

2008

0.X-BETA

ICAAtom

1.0-BETA

FIRST
NON-BETA
RELEASE

1.1

1.2

1.3

2014

atom
access to memory

2.0

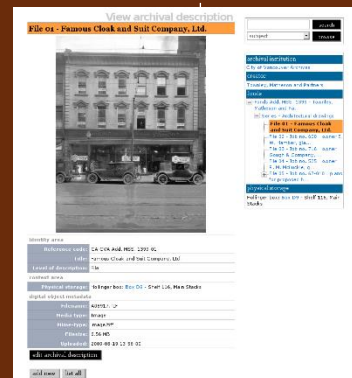
2009

2010

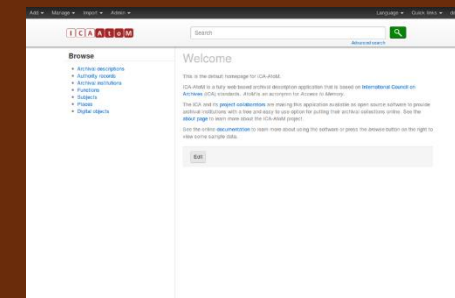
2011

2012

2013



TRILLIUM THEME





AtOM DEVELOPMENT

2008

2014

0.X-BETA

1.0-BETA

FIRST
NON-BETA
RELEASE

1.1

1.2

1.3

atom
access to memory

2.0

2009

2010

2011

2012

2013



Qubit developer's wiki

DCB wiki

ICA-AtOM wiki

TRILLIUM THEME



2013:

1 PROJECT...

3 WIKIS

4 THEMES

And AtoM 2.0 on the way...



<http://imgur.com/32eaxCQ>

WIKI CHALLENGES

No versioning (e.g. ICA-AtoM 1.2, 1.3; AtoM 2.0)

No enforced structure

Too easy to create orphaned pages

No easy output to other formats



ENTER



SPHINX

Python Documentation Generator





SPHINX

Python Documentation Generator

OVERVIEW

The screenshot shows the Sphinx website's 'Welcome' page. It features the Sphinx logo and navigation links (Home, Get it, Docs, Extend/Develop). The main content area includes a 'Welcome' section with a description of Sphinx as a tool for creating intelligent documentation, a 'What users say' quote, and a list of features like output formats, cross-references, and extensions. There are also links to documentation, a search bar, and a quick search section. The footer contains copyright information and a list of contributors.

Welcome

Sphinx is a tool that makes it easy to create intelligent and beautiful documentation, written by Georg Brandl and licensed under the BSD license.

It was originally created for [the new Python documentation](#), and it has excellent facilities for the documentation of Python projects, but C/C++ is already supported as well, and it is planned to add special support for other languages as well. Of course, this site is also created from reStructuredText sources using Sphinx! The following features should be highlighted:

- **Output formats:** HTML (including Windows HTML Help), LaTeX (for printable PDF versions), ePub, Texinfo, manual pages, plain text
- **Extensive cross-references:** semantic markup and automatic links for functions, classes, citations, glossary terms and similar pieces of information
- **Hierarchical structure:** easy definition of a document tree, with automatic links to siblings, parents and children
- **Automatic indices:** general index as well as a language-specific module indices
- **Code handling:** automatic highlighting using the [Pygments](#) highlighter
- **Extensions:** automatic testing of code snippets, inclusion of docstrings from Python modules (API docs), and [more](#)
- **Contributed extensions:** more than 50 extensions [contributed by users](#) in a second repository; most of them installable from PyPI

Sphinx uses [reStructuredText](#) as its markup language, and many of its strengths come from the power and straightforwardness of reStructuredText and its parsing and translating suite, the [Docutils](#).

Documentation

[First steps with Sphinx](#)
overview of basic tasks

[Search page](#)
search the documentation

[Contents](#)
for a complete overview

[General Index](#)
all functions, classes, terms

You can also download PDF versions of the Sphinx documentation: a [version](#) generated from the LaTeX Sphinx produces, and a [version](#) generated by rst2pdf.

Examples

Links to documentation generated with Sphinx can be found on the [Projects using Sphinx](#) page.

For examples of how Sphinx source files look, use the "Show source" links on all pages of the documentation apart from this welcome page.

You may also be interested in the very nice [tutorial](#) on how to create a customized documentation using Sphinx written by the matplotlib developers.

There is a [Japanese translation](#) of this documentation, thanks to the Japanese Sphinx user group.

A Japanese book about Sphinx has been published by O'Reilly: [Sphinx @ はじめての Sphinx Learning Sphinx](#).

Hosting

Need a place to host your Sphinx docs? [readthedocs.org](#) hosts a lot of Sphinx docs already, and integrates well with projects' source control. It also features a powerful built-in search that exceeds the possibilities of Sphinx' JavaScript-based offline search.

Contributor Guide

If you want to contribute to the project, this part of the documentation is for you.

- [Sphinx Developer's Guide](#)
- [Sphinx Authors](#)
- [Sphinx Authors](#)

© Copyright 2007-2015, Georg Brandl and the Sphinx team. Created using Sphinx 1.3.1+rc4746b.

Documentation generator
Open source
Created by Python community
Builds on existing open source projects:

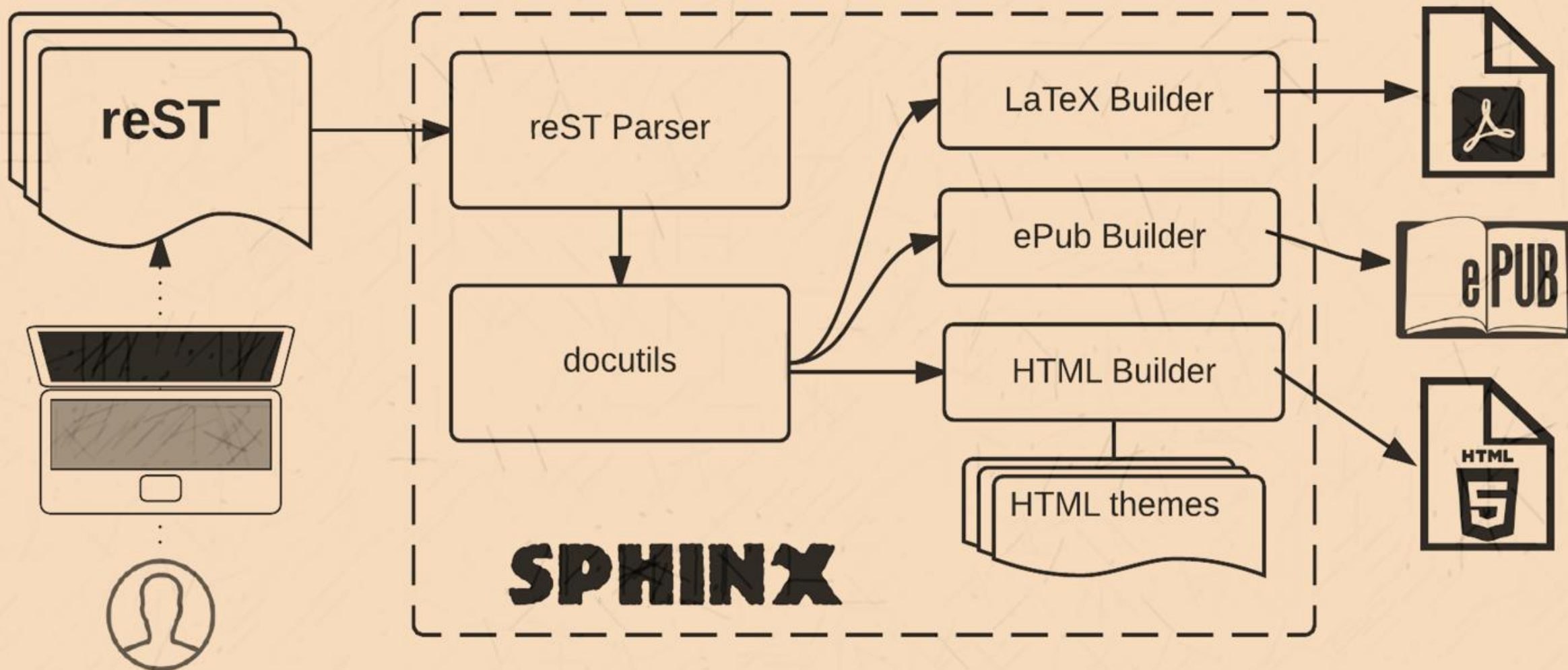
- reStructured Text (markup language)
- Docutils (text processing system)



SPHINX

Python Documentation Generator

OVERVIEW





SPHINX
Python Documentation Generator

ADVANTAGES

- Structured documentation
 - Built around a table of contents you define
- Versioning
 - Easily maintained in a repository w many branches
- Easy output to other formats
- Glossary, footnotes, asides, figures, tables, etc...
- Built-in themes or custom themes
- Automated indices
- Strong support for code documentation



<http://cheezburger.com/6705529088>

reST BASICS



reST BASICS

`.. _section-anchor:`

`=====`

`Section header`

`=====`

`Subsection header`

`=====`

`Sub-subsection`

`-----`

Example of a `:term:` `glossary term` and a `:ref:` `reference <section-anchor>` used in a paragraph

```
.. image:: /images/my-image.png
   :align: center
   :width: 80%
   :alt: my image alt text
```

Text can be **strong** or have *emphasis*

reST BASICS

`.. IMPORTANT::`

This is an admonition. Sphinx also supports NOTE, TIP, SEEALSO, WARNING, CAUTION, HINT, etc.

This is a ``link`_` that can be reused

`.. _link: http://www.example.com`

This is an ``inline link`_`
`<http://www.example2.com>`_`

`.. code-block:: xml`

```
<scopecontent>My scope  
and content</scopecontent>
```

- * This is a bulleted list.
- * It has two items, the second item uses two lines.

1. This is a numbered list.
2. It has two items too.

- #. This is a numbered list.
- #. It has two items too.


```
129 .. _add-function-add-menu:
```

```
131 Add a new function from the main menu
```

```
134 .. image:: images/add-functions.*
```

```
135 :align: right
```

```
136 :width: 20%
```

```
137 :alt: Choosing functions in the Add menu
```

A new function can be added at any time, from anywhere in the application, via the `:term:`main menu``, provided that a user has sufficient `:term:`access privileges <access privilege>`` to create new records. For more information on user permissions, see: `:ref:`user-roles`` and `:ref:`default-user-permissions``.

****To create a new function from the main menu****

1. In the `:term:`main menu`` located in the `:term:`header bar``, click the `|plus| :ref:`"Add" <main-menu-add>`` menu and select "Function" from the `:term:`drop-down menu``.
2. AtoM takes you to a blank `:term:`edit page`` for data entry.
3. On loading, the `:term:`edit page`` displays the record with all `:term:`information areas <information area>`` closed; click on an `:term:`information area`` to access the `:term:`fields <field>`` grouped under it. Enter data as required.

```
155 .. figure:: images/blank-function.*
```

```
156 :align: center
```

```
157 :figwidth: 60%
```

```
158 :width: 100%
```

```
159 :alt: An image of a blank ISDF record in edit mode
```

When creating a new function from the Add menu, all information areas will appear closed on the blank function record - click to open and add data as required.

4. You can quit the process at any time by clicking the "Cancel" button in the `:term:`button block``; any data already entered will not be saved and no new record will be added. Note that simply navigating away from the page by

ADD A NEW FUNCTION FROM THE MAIN MENU

A new function can be added at any time, from anywhere in the application, via the [main menu](#), provided that a user has sufficient [access privileges](#) to create new records. For more information on user permissions, see: [User roles](#) and [Default user permissions settings in AtoM](#).

To create a new function from the main menu

1. In the [main menu](#) located in the [header bar](#), click the **+** "Add" menu and select "Function" from the [drop-down menu](#).
2. AtoM takes you to a blank [edit page](#) for data entry.
3. On loading, the [edit page](#) displays the record with all [information areas](#) closed; click on an [information area](#) to access the [fields](#) grouped under it. Enter data as required.

Edit Function - ISDF

Untitled

Identity area

Context area

Relationships area

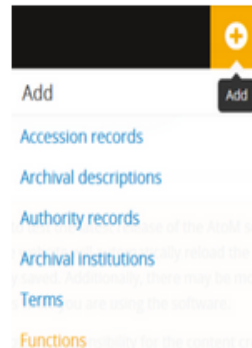
Control area

Cancel

Create

When creating a new function from the Add menu, all information areas will appear closed on the blank function record - click to open and add data as required.

4. You can quit the process at any time by clicking the "Cancel" button in the [button block](#); any data already entered will not be saved and no new record will be added. Note that simply navigating away from the page by any other means, **without first clicking "Create"** will also result in no new record being created.
5. To save the new function record, click the **"Create" button** in the [button block](#) at the bottom of the record. AtoM will redirect you to the function's [view page](#), so you can review your work.



Section anchor

(so you can link to this heading)

Section header

Image aligned right

```

129 .. _add-function-add-menu:
130
131 Add a new function from the main menu
132 -----
133
134 .. image:: images/add-functions.*
135    :align: right
136    :width: 20%
137    :alt: Choosing functions in the Add menu

```

A new function can be added at any time, from anywhere in the application, via the `:term:`main menu``, provided that a user has sufficient `:term:`access privileges <access privilege>`` to create new records. For more information on user permissions, see: `:ref:`user-roles`` and `:ref:`default-user-permissions``.

To create a new function from the main menu

1. In the `:term:`main menu`` located in the `:term:`header bar``, click the `|plus|` `:ref:`"Add" <main-menu-add>`` menu and select "Function" from the `:term:`drop-down menu``.
2. AtoM takes you to a blank `:term:`edit page`` for data entry.
3. On loading, the `:term:`edit page`` displays the record with all `:term:`information areas <information area>`` closed; click on an `:term:`information area`` to access the `:term:`fields <field>`` grouped under it. Enter data as required.

```

155 .. figure:: images/blank-function.*
156    :align: center
157    :figwidth: 60%
158    :width: 100%
159    :alt: An image of a blank ISDF record in edit mode

```

When creating a new function from the Add menu, all information areas will appear closed on the blank function record - click to open and add data as required.

4. You can quit the process at any time by clicking the "Cancel" button in the `:term:`button block``; any data already entered will not be saved and no new record will be added. Note that simply navigating away from the page by any other means, **without first clicking "Create"** will also result in no new record being created.

ADD A NEW FUNCTION FROM THE MAIN MENU

A new function can be added at any time, from anywhere in the application, via the **main menu**, provided that a user has sufficient **access privileges** to create new records. For more information on user permissions, see: [User roles](#) and [Default user permissions settings in AtoM](#).

To create a new function from the main menu

1. In the **main menu** located in the **header bar**, click the **+** "Add" menu and select "Function" from the **drop-down menu**.
2. AtoM takes you to a blank **edit page** for data entry.
3. On loading, the **edit page** displays the record with all **information areas** closed; click on an **information area** to access the **fields** grouped under it. Enter data as required.

Edit Function - ISDF

Untitled

Identity area

Context area

Relationships area

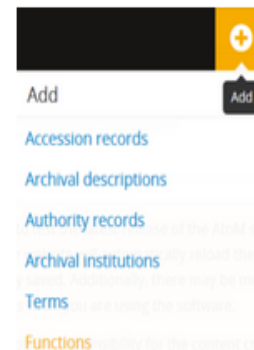
Control area

Cancel

Create

When creating a new function from the Add menu, all information areas will appear closed on the blank function record - click to open and add data as required.

4. You can quit the process at any time by clicking the "Cancel" button in the **button block**; any data already entered will not be saved and no new record will be added. Note that simply navigating away from the page by any other means, **without first clicking "Create"** will also result in no new record being created.
5. To save the new function record, click the **"Create" button** in the **button block** at the bottom of the record. AtoM will redirect you to the function's **view page**, so you can review your work.



```

129 .. _add-function-add-menu:
130
131 Add a new function from the main menu
132 -----
133
134 .. image:: images/add-functions.*
135    :align: right
136    :width: 20%
137    :alt: Choosing functions in the Add menu

```

Glossary term

A new function can be added at any time, from anywhere in the application, via the `:term:`main menu``, provided that a user has sufficient `:term:`access privileges`` to create new records. For more information on user permissions, see: `:ref:`user-roles`` and `:ref:`default-user-permissions``.

****To create a new function from the main menu****

1. In the `:term:`main menu`` located in the `:term:`header bar``, click the `|plus|` `:ref:`"Add" <main-menu-add>`` menu and select "Function" from the `:term:`drop-down menu``.
2. AtoM takes you to a blank `:term:`edit page`` for data entry.
3. On loading, the `:term:`edit page`` displays the record with all `:term:`information areas <information area>`` closed; click on an `:term:`information area`` to access the `:term:`fields <field>`` grouped under it. Enter data as required.

```

155 .. figure:: images/blank-function.*
156    :align: center
157    :figwidth: 60%
158    :width: 100%
159    :alt: An image of a blank ISDF record in edit mode

```

Figure (image with text underneath) centered

When creating a new function from the Add menu, all information areas will appear closed on the blank function record - click to open and add data as required.

4. You can quit the process at any time by clicking the "Cancel" button in the `:term:`button block``; any data already entered will not be saved and no new record will be added. Note that simply navigating away from the page by any other means, **without first clicking "Create"** will also result in no new record being created.
5. To save the new function record, click the `"Create" button` in the `button block` at the bottom of the record. AtoM will redirect you to the function's `view page`, so you can review your work.

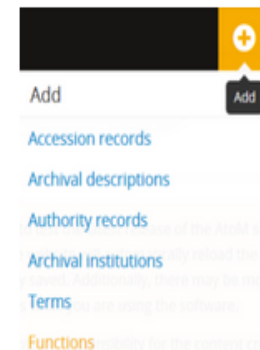
ADD A NEW FUNCTION FROM THE MAIN MENU

A new function can be added at any time, from anywhere in the application, via the `main menu`, provided that a user has sufficient `access privileges` to create new records. For more information on user permissions, see: `User roles` and `Default user permissions settings in AtoM`.

To create a new function from the main menu

1. In the `main menu` located in the `header bar`, click the `+` "Add" menu and select "Function" from the `drop-down menu`.
2. AtoM takes you to a blank `edit page` for data entry.
3. On loading, the `edit page` displays the record with all `information areas` closed; click on an `information area` to access the `fields` grouped under it. Enter data as required.

4. You can quit the process at any time by clicking the "Cancel" button in the `button block`; any data already entered will not be saved and no new record will be added. Note that simply navigating away from the page by any other means, **without first clicking "Create"** will also result in no new record being created.
5. To save the new function record, click the `"Create" button` in the `button block` at the bottom of the record. AtoM will redirect you to the function's `view page`, so you can review your work.




```
129 .. _add-function-add-menu:
```

```
131 Add a new function from the main menu
```

```
134 .. image:: images/add-functions.*
```

```
135 :align: right
```

```
136 :width: 20%
```

```
137 :alt: Choosing functions in the Add menu
```

```
139 A new function can be added at any time, from anywhere in the application,
140 via the :term:`main menu`, provided that a user has sufficient :term:`access
141 privileges` <access privilege> to create new records. For more information on
142 user permissions, see: :ref:`user-roles` and :ref:`default-user-permissions`.
```

```
144 **To create a new function from the main menu**
```

1. In the :term:`main menu` located in the :term:`header bar`, click the **+** "Add" menu and select "Function" from the :term:`drop-down menu`.
2. AtoM takes you to a blank :term:`edit page` for data entry.
3. On loading, the :term:`edit page` displays the record with all :term:`information areas` <information area> closed; click on an :term:`information area` to access the :term:`fields` <field> grouped under it. Enter data as required.

```
155 .. figure:: images/blank-function.*
```

```
156 :align: center
```

```
157 :figwidth: 60%
```

```
158 :width: 100%
```

```
159 :alt: An image of a blank ISDF record in edit mode
```

When creating a new function from the Add menu, all information areas will appear closed on the blank function record - click to open and add data as required.

4. You can quit the process at any time by clicking the "Cancel" button in the :term:`button block`; any data already entered will not be saved and no new record will be added. Note that simply navigating away from the page by any other means, **without first clicking "Create"** will also result in no new record being created.
5. To save the new function record, click the **"Create" button** in the :term:`button block` at the bottom of the record. AtoM will redirect you to the function's **view page**, so you can review your work.

ADD A NEW FUNCTION FROM THE MAIN MENU

A new function can be added at any time, from anywhere in the application, via the **main menu**, provided that a user has sufficient **access privileges** to create new records. For more information on user permissions, see: [User roles](#) and [Default user permissions settings in AtoM](#).

To create a new function from the main menu

1. In the **main menu** located in the **header bar**, click the **+** "Add" menu and select "Function" from the **drop-down menu**.
2. AtoM takes you to a blank **edit page** for data entry.
3. On loading, the **edit page** displays the record with all **information areas** closed; click on an **information area** to access the **fields** grouped under it. Enter data as required.

Edit Function - ISDF

Untitled

Identity area

Context area

Relationships area

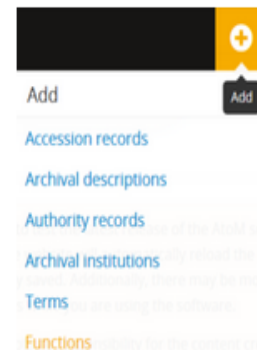
Control area

Cancel

Create

When creating a new function from the Add menu, all information areas will appear closed on the blank function record - click to open and add data as required.

4. You can quit the process at any time by clicking the "Cancel" button in the **button block**; any data already entered will not be saved and no new record will be added. Note that simply navigating away from the page by any other means, **without first clicking "Create"** will also result in no new record being created.
5. To save the new function record, click the **"Create" button** in the **button block** at the bottom of the record. AtoM will redirect you to the function's **view page**, so you can review your work.



DETAILED CONTRIBUTION GUIDELINES

Outlined on the [Atom wiki](https://wiki.accesstomemory.org/Atom/wiki)
Notes on getting started and
elements of style for consistency
across the project

https://wiki.accesstomemory.org/Resources/Documentation/Contribution_guidelines

@atom

Home

Downloads

Documentation

Community

Wiki

Demo

Documentation contribution guidelines

< Resources | Documentation

Main Page > Resources > Resources/Documentation > Resources/Documentation/Contribution guidelines

This page offers a overview of best-practice suggestions for contributing documentation. We strongly recommend reviewing this page before contributing to the AtoM documentation - thanks! This document covers how to structure and format contributions to meet the established conventions of Sphinx, and of our documentation structure. For information on how you can contribute, see: [Contribute documentation](#).

Senaho

• Contribute code

• Contribute translations

Contents

(hide)

1 Contribution guidelines

1.1 Before contributing

1.2 Contributing new documentation

1.3 Questions?

Contribution guidelines

(edit)

So you're thinking of contributing to our documentation? See something that needs fixing, or something that could be improved? Great! Below you'll find some helpful guidelines to get you started, and to familiarize you with the approach we have been trying to follow.

In general, there are two main types of contributions:

• **General improvements:** typo corrections, fixing broken refs or links, correcting inaccurate or out-of-date information, and offering better explanations through clearer writing and additional examples.

• **New features or new pages:** Adding a page of documentation that we haven't yet covered in our ongoing rewrite attempts as we move platforms, or documenting a new feature that has been added to AtoM since the last release.

We welcome both kinds of contributions from our users, and are always aiming to improve our documentation as much and as often as possible. To maintain a sense of consistency and quality over time, here are a few tips.

BEFORE CONTRIBUTING

(edit)

1. Understand that our documents are all released under a Creative Commons Attribution-ShareAlike 4.0 license (CC BY SA#).

This means that your contributions are being released to Artefactual and the AtoM community to use, share, remix, and distribute as they see fit, provided any distribution takes place under the same licensing. Your work will be a gift to the AtoM user community, and we thank you for it!

2. Read through some of the documents we've already created.

You'll notice that there is a consistent structure to most of them, which we will describe in greater detail below. By reading through some of our existing documentation, you will get a feeling for the overall structure and style. By looking at the source code (there's always a link to the source code in the sidebar), you can see how we've used Sphinx# to style and structure the page.

3. Familiarize yourself with Sphinx and reStructuredText before starting.

As outlined above in our About our documentation section of the wiki, we are using the Sphinx Python Documentation Generator# to produce the AtoM documentation. Sphinx employs reStructured Text # as its markup language, and you'll need to understand how to use its syntax if you want to contribute. The "First Steps with Sphinx" (here #) is a great introduction to some of the key elements of Sphinx, and the reStructuredText Primer (here #) will introduce you to the key characteristics of reST.

4. Examine the AtoM documentation glossary.

Over the years, we have been adding to the glossary in our User Manual in an attempt to be consistent with how we refer to different parts of the application.

ACCOUNT

• Dan

• Talk

• Preferences

• Watchlist

• Contributions

• Log out

PAGE

• Discussion

• Edit

• History

• Delete

• Move

• Protect


• Unwatch

SEARCH

Search

LEARN MORE ABOUT SPHINX AND reST

- Sphinx documentation: <http://www.sphinx-doc.org/>
- Sphinx reST primer: <http://www.sphinx-doc.org/en/stable/rest.html#rst-primer>
- Full reStructuredText docs: <http://docutils.sourceforge.net/rst.html>

**SPHINX**
Python Documentation Generator

HomeGet itDocsExtend/Develop

Welcome

Sphinx is a tool that makes it easy to create intelligent and beautiful documentation, written by Georg Brandl and licensed under the BSD license.


It was originally created for [the Python documentation](#), and it has excellent facilities for the documentation of software projects in a range of languages. Of course, this site is also created from reStructuredText sources using Sphinx!

The following features should be highlighted:

- **Output formats:** HTML (including Windows HTML Help), LaTeX (for printable PDF versions), ePub, Texinfo, manual pages, plain text
- **Extensive cross-references:** semantic markup and automatic links for functions, classes, citations, glossary terms and similar pieces of information
- **Hierarchical structure:** easy definition of a document tree, with automatic links to siblings, parents and children
- **Automatic indices:** general index as well as a language-specific module indices
- **Code handling:** automatic highlighting using the [Pygments](#) highlighter
- **Extensions:** automatic testing of code snippets, inclusion of docstrings from Python modules (API docs), and [more](#)
- **Contributed extensions:** more than 50 extensions [contributed by users](#) in a second repository; most of them installable from PyPI

What users say:

"Cheers for a great tool that actually makes programmers **want** to write documentation!"

A  project

Download

Current version: **1.5.4**

Get Sphinx from the [Python Package Index](#), or install it with:

```
pip install -U Sphinx
```

Questions? Suggestions?

Join the [sphinx-users](#) mailing list on Google Groups:

CONTRIBUTING DOCUMENTATION CHANGES WITH VAGRANT:

Using Vagrant, git, and a text editor to view, edit, and add documentation from your local computer



CLI 101 REMINDERS

Autocomplete a directory/location/filename	> TAB
Show last command	> UP ARROW
Abort a process	> CTRL+C
Change to a directory	> cd <path/to/dir>
Go up one directory	> cd ..
Go to home directory	> cd
Clear current commands out of view	> clear

There are some great cheat sheets and tutorials out there – a simple internet search will uncover many options

SETTING UP THE DOCS IN VAGRANT

Change to the **atom-docs** directory

```
> cd atom-docs
```

Check what branch you are on, and what others are available

```
> git branch -avv
```

```
vagrant$ git branch -avv
 2.3          6cdb33b [origin/2.3: behind 7] update publication permissions notes, refs #9304
* 2.4          973f7ea [origin/2.4] Xenial FOP install procedure change
```

Branch

last commit

last commit message

* = **current branch**

SETTING UP THE DOCS IN VAGRANT

Check out the branch you want

```
> git checkout <branch name>
```

```
vagrant$ git checkout 2.3
Switched to branch '2.3'
Your branch is behind 'origin/2.3' by 7 commits, and can be fast-forwarded.
  (use "git pull" to update your local branch)
[~/atom-docs] (2.3)
vagrant$
```

Pull in the latest changes and rebase your work on top

```
> git pull --rebase
```

WAIT... WHAT THE HECK IS GIT??

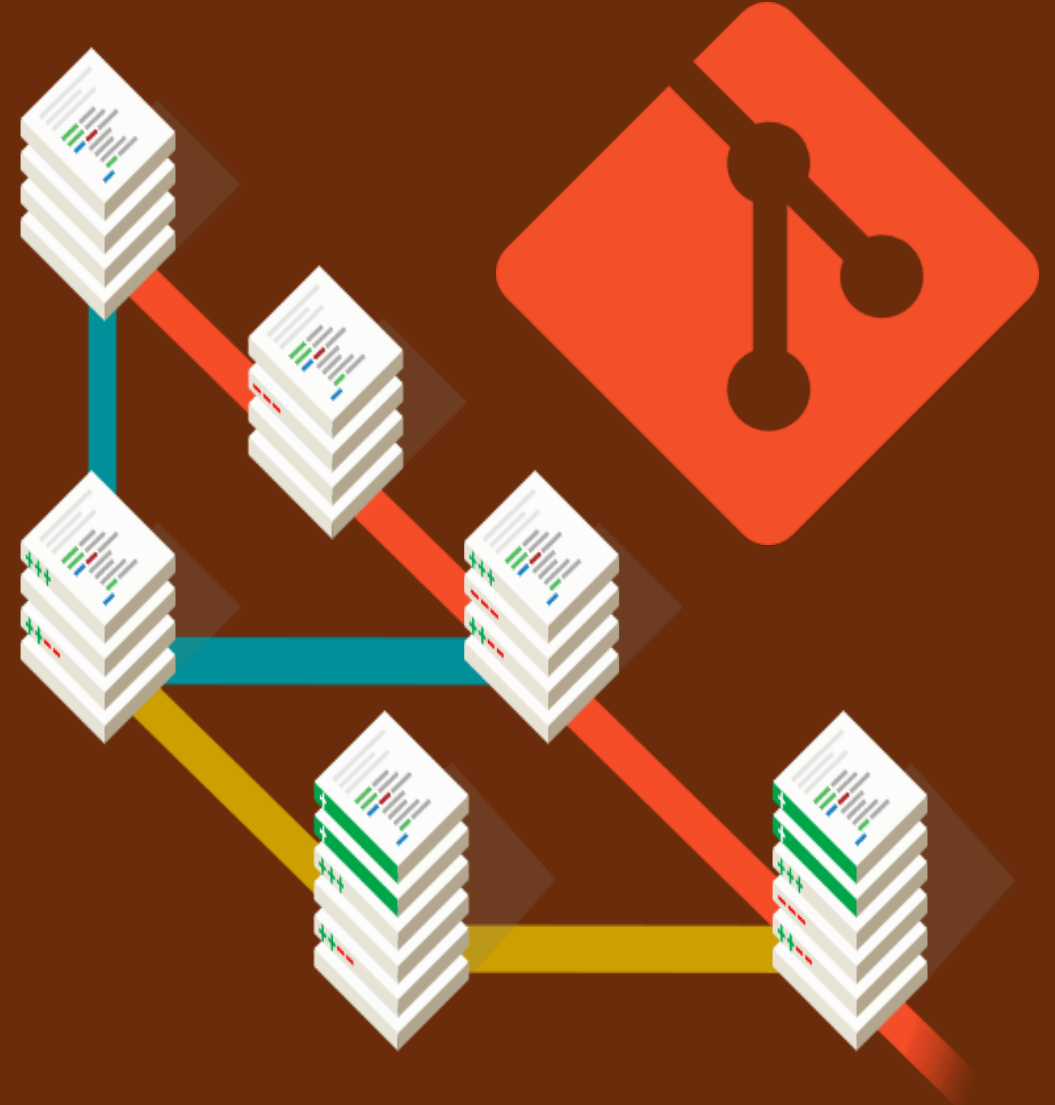


Distributed version control and git

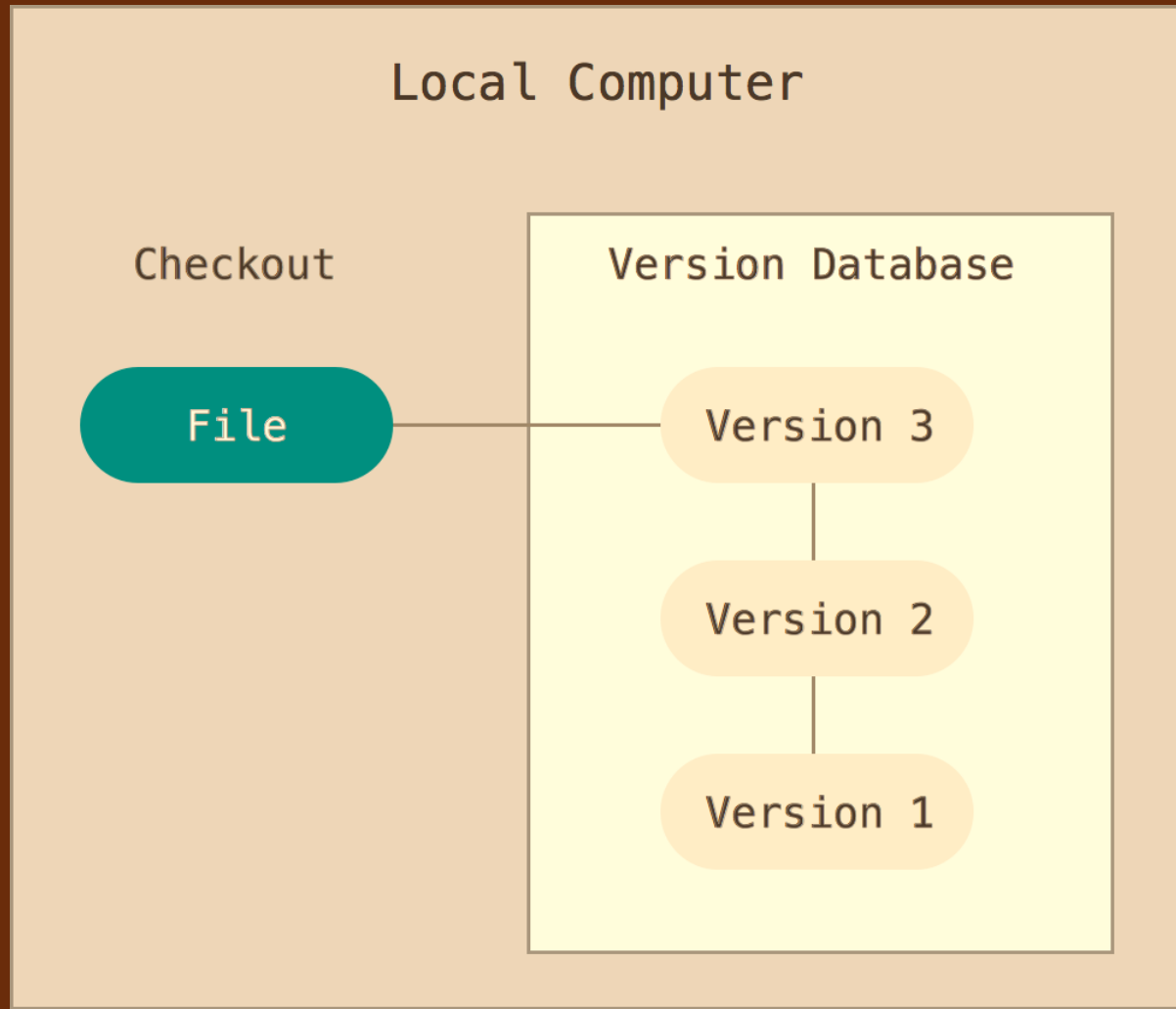
Version control:

A system for managing changes to source data (documents, code, etc) over time.

Version control systems help you track who made what change when, and why. They also allow you to roll back changes to your data to a previous state.

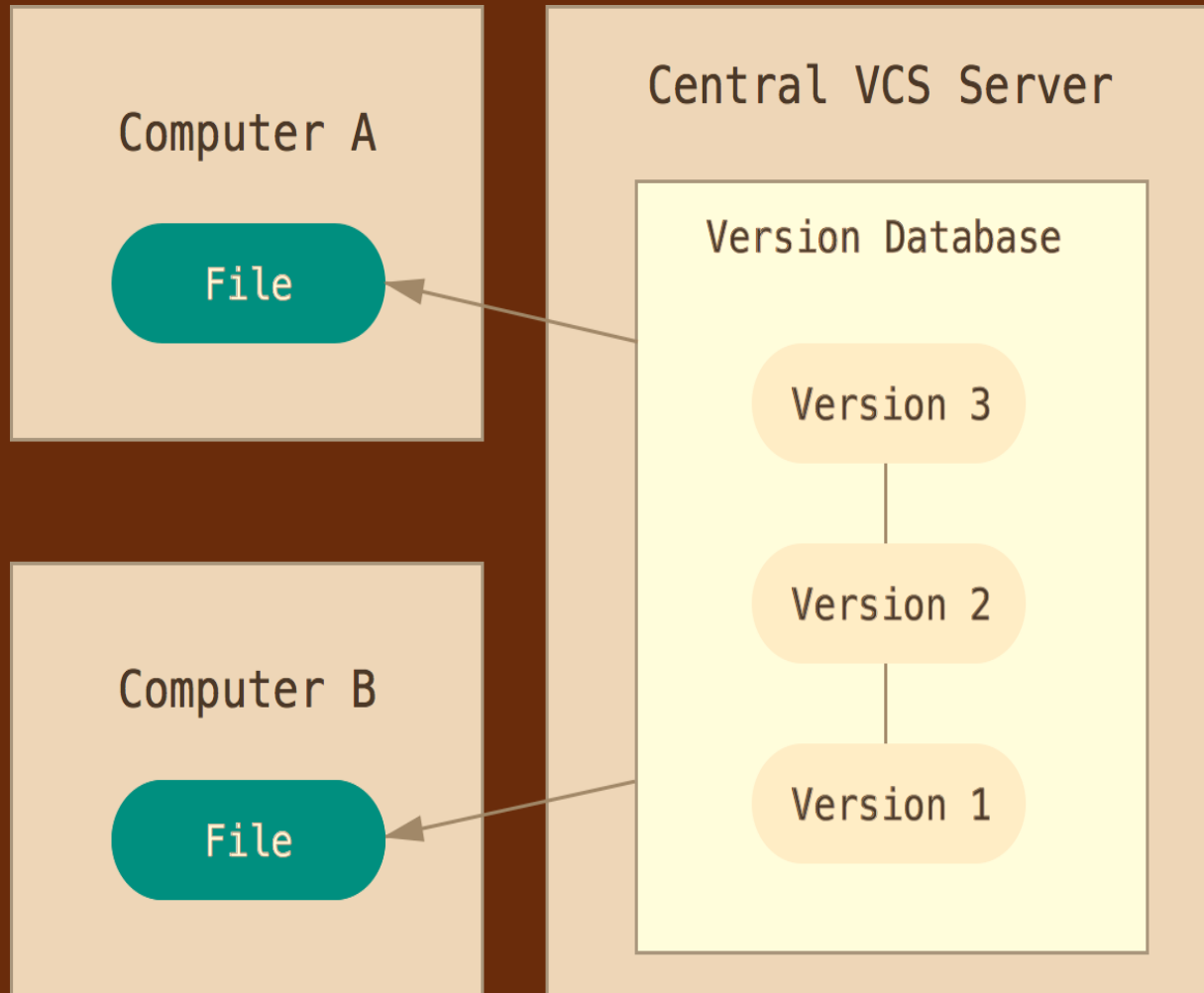


Local version control



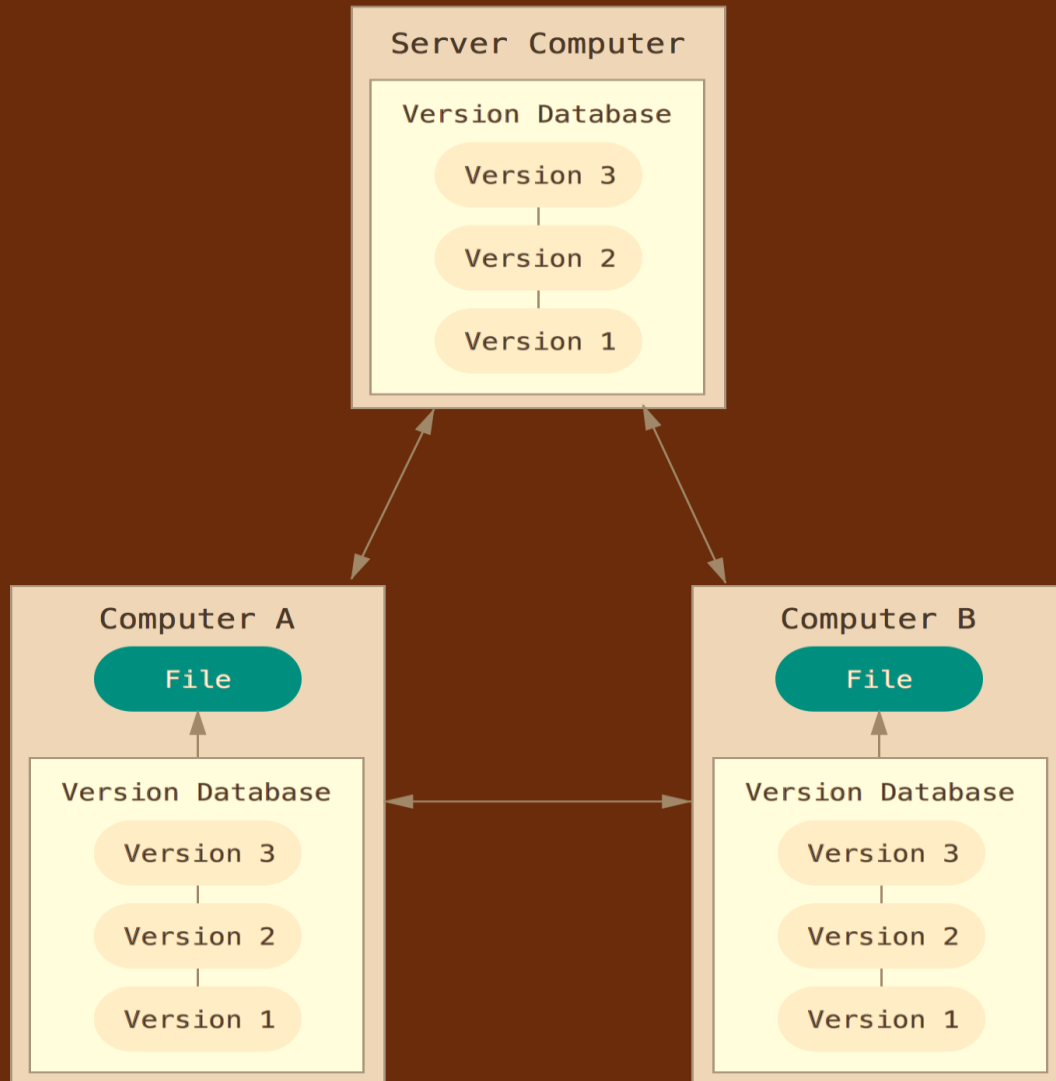
- Changes are tracked on a local computer
- Simple – great for working alone
- Not effective for collaboration

Centralized version control



- Single server with all versioned files; users “check out” files
- Allows for collaboration
- Single point of failure

Distributed version control



- Full mirroring of source with each user
- Broadest potential for collaboration
- Complex merging, branching, etc. possible



- Open source
- Distributed model
- Strong User Community, lots of free documentation
- Widely used and supported across platforms

THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.

COOL. HOW DO WE USE IT?

NO IDEA. JUST MEMORIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP. IF YOU GET ERRORS, SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.





- Hashes all files in the repository
- Produces a diff of changes →
- Commit messages explain who did what when and why
- Supports parallel workflows with a method for resolving conflicts

```
584 - * AtoM looks for creator names in the *creators* column in the RAD and ISAD CSV
585 - import templates, as well as :term:`access point` names (used as subjects) in
586 - the *nameAccessPoints* column during a CSV import of
622 + * AtoM looks for creator names in the *eventActors* column in the RAD and ISAD
623 + CSV import templates, as well as :term:`access point` names (used as subjects)
624 + in the *nameAccessPoints* column during a CSV import of
587 625 :term:`archival descriptions <archival description>`.
588 626 * Similarly, any Administrative / biographical history data in an archival
589 - description CSV import (i.e. data contained in the *creatorHistories* CSV
627 + description CSV import (i.e. data contained in the *eventActorHistories* CSV
590 628 column will be mapped to the "History" :term:`field` (ISAAR-CPF 5.2.2) in the
591 629 related :term:`authority record` (generated from the data contained in the
592 630 *creators* column of the CSV), and then is presented in AtoM in any related
593 631 descriptions where the entity is listed as a creator.
594 632 * Where multiple creator names and histories are included in an import,
595 - *creators* and *creatorHistories* elements are matched 1:1 in the order they
596 - appear in the CSV, divided by pipe elements (e.g. ``|``). For example, if the
597 - *creators* column contains ``name 1|name 2``, the *creatorHistories* should
598 - also include ``history 1|history 2`` to match on import.
599 - * If a creator history element is included in a CSV import, but no creator
633 + *eventActors* and *eventActorHistories* elements are matched 1:1 in the
634 + order they appear in the CSV, divided by pipe elements (e.g. ``|``). For
635 + example, if the *eventActors* column contains ``name 1|name 2``, the
636 + *eventActorHistories* should also include ``history 1|history 2`` to match on
637 + import. If there is *no* history for the first actor, you can include
638 + ``NULL``, and AtoM will ignore the input - e.g. ``name 1|name 2`` should be
639 + matched with ``NULL|history 2`` to include only a history for name 2.
640 + * This same ``NULL`` approach can be used for any matched date values where
641 + multiple actor names are included for import - *eventDates*,
642 + *eventStartDates*, *eventEndDates* can all include ``NULL`` if you wish
643 + to leave these blank when associating multiple actors with an event. An
644 + example, using the RAD template:
645 +
646 +.. image:: images/csv-creatorDates-2.*
647 + :align: center
```



GitHub is a web-based Git or version control repository and Internet hosting service. It offers all of the distributed version control and source code management (SCM) functionality of Git as well as adding its own features. It provides access control and several collaboration features such as bug tracking, feature requests, task management, and wikis for every project.

GitHub offers both plans for private and free repositories on the same account which are commonly used to host open-source software projects ([Wikipedia](https://en.wikipedia.org/wiki/GitHub))

The screenshot shows the GitHub interface for the repository 'artefactual / atom-docs'. At the top, there are navigation links for 'Features', 'Explore', and 'Pricing', along with a search bar and 'Sign in or Sign up' buttons. The repository name is displayed, followed by statistics: 16 watches, 8 stars, and 10 forks. Below this, there are tabs for 'Code', 'Pull requests', 'Projects', 'Pulse', and 'Graphs'. The 'Code' tab is selected, showing the repository's URL: 'https://www.accesstomemory.org/docs/'. A summary bar indicates 948 commits, 7 branches, 0 releases, 14 contributors, and a CC-BY-SA-4.0 license. Below the summary, there are buttons for 'Branch: 2.3', 'New pull request', 'Find file', and 'Clone or download'. The main content area shows a list of files and their commit history, including '_static', 'admin-manual', 'dev-manual', 'images', 'user-manual', '.editorconfig', '.gitignore', 'LICENCE', 'Makefile', 'index.rst', and 'index.rst'. The footer contains copyright information, links to 'Terms', 'Privacy', 'Security', 'Status', and 'Help', and a 'Contact GitHub' link.

<https://github.com/artefactual/atom-docs/>



GitHub

Note: You will need a GitHub account to be able to submit changes from the Vagrant box. It's free and really easy to set up. Head over to: <https://github.com/> to create one.



Features Explore Pricing

Search GitHub

Sign in or Sign up

Built for developers

GitHub is a development platform inspired by the way you work.
Host code, manage projects, and build software alongside
millions of other developers.

Pick a username

Your email address

Create a password

Use at least one letter, one numeral, and seven characters.

Sign up for GitHub


By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We'll occasionally send you account related emails.

Learn all things git (for free)

<https://git-scm.com/book/>

The entire Pro Git book, written by Scott Chacon and Ben Straub and published by Apress, is available here. All content is licensed under the Creative Commons Attribution Non Commercial Share Alike 3.0 license. Print versions of the book are available on Amazon.com.

There are a lot of other free books and tutorials out there as well!

 **git** --distributed-even-if-your-workflow-isnt

Search entire site...

About

- Documentation**
 - Reference
 - Book
 - Videos
 - External Links
- Blog**
- Downloads**
- Community**

This book is available in [English](#).

Full translation available in [Deutsch](#), [简体中文](#), [正體中文](#), [Français](#), [日本語](#), [Nederlands](#), [Русский](#), [한국어](#), [Português \(Brasil\)](#) and [Čeština](#).


Partial translations available in [Arabic](#), [Español](#), [Indonesian](#), [Italiano](#), [Suomi](#), [Македонски](#), [Ελληνικά](#), [Polski](#) and [Türkçe](#).

Translations started for [Azərbaycan dili](#), [Беларуская](#), [Català](#), [Esperanto](#), [Español \(Nicaragua\)](#), [فارسی](#), [हिन्दी](#), [Magyar](#), [Norwegian Bokmål](#), [Română](#), [Српски](#), [ភាសាខ្មែរ](#), [Tiếng Việt](#), [Українська](#) and [Ўзбекча](#).

The source of this book is [hosted on GitHub](#).

Book

The entire Pro Git book, written by Scott Chacon and Ben Straub and published by Apress, is available here. All content is licensed under the [Creative Commons Attribution Non Commercial Share Alike 3.0 license](#). Print versions of the book are available on [Amazon.com](#).



2nd Edition (2014)
[Switch to 1st Edition](#)

1. Getting Started

- 1.1 [About Version Control](#)
- 1.2 [A Short History of Git](#)
- 1.3 [Git Basics](#)
- 1.4 [The Command Line](#)
- 1.5 [Installing Git](#)
- 1.6 [First-Time Git Setup](#)
- 1.7 [Getting Help](#)
- 1.8 [Summary](#)

2. Git Basics

- 2.1 [Getting a Git Repository](#)
- 2.2 [Recording Changes to the Repository](#)
- 2.3 [Viewing the Commit History](#)
- 2.4 [Undoing Things](#)
- 2.5 [Working with Remotes](#)
- 2.6 [Tagging](#)
- 2.7 [Git Aliases](#)
- 2.8 [Summary](#)

OKAY COOL THX KEEP GOING



SETTING UP THE DOCS IN VAGRANT

Clean out the HTML working directory

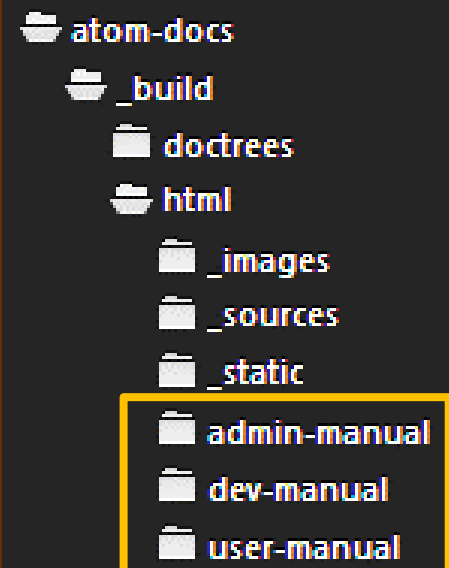
```
> make clean
```

Rebuild the local HTML files based on your local copy

```
> make html
```

HTML versions of your files will be generated and placed in:

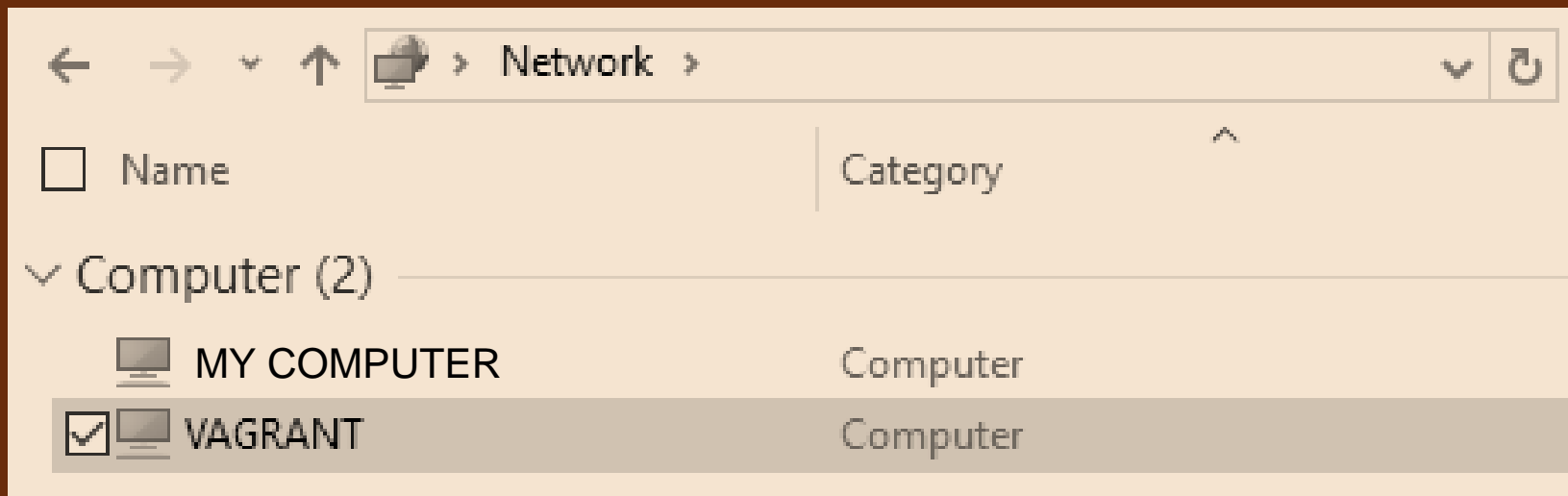
atom-docs/_build/html/



You can use these local HTML files to preview and proof your work in your web browser before submitting it

SETTING UP THE DOCS IN VAGRANT

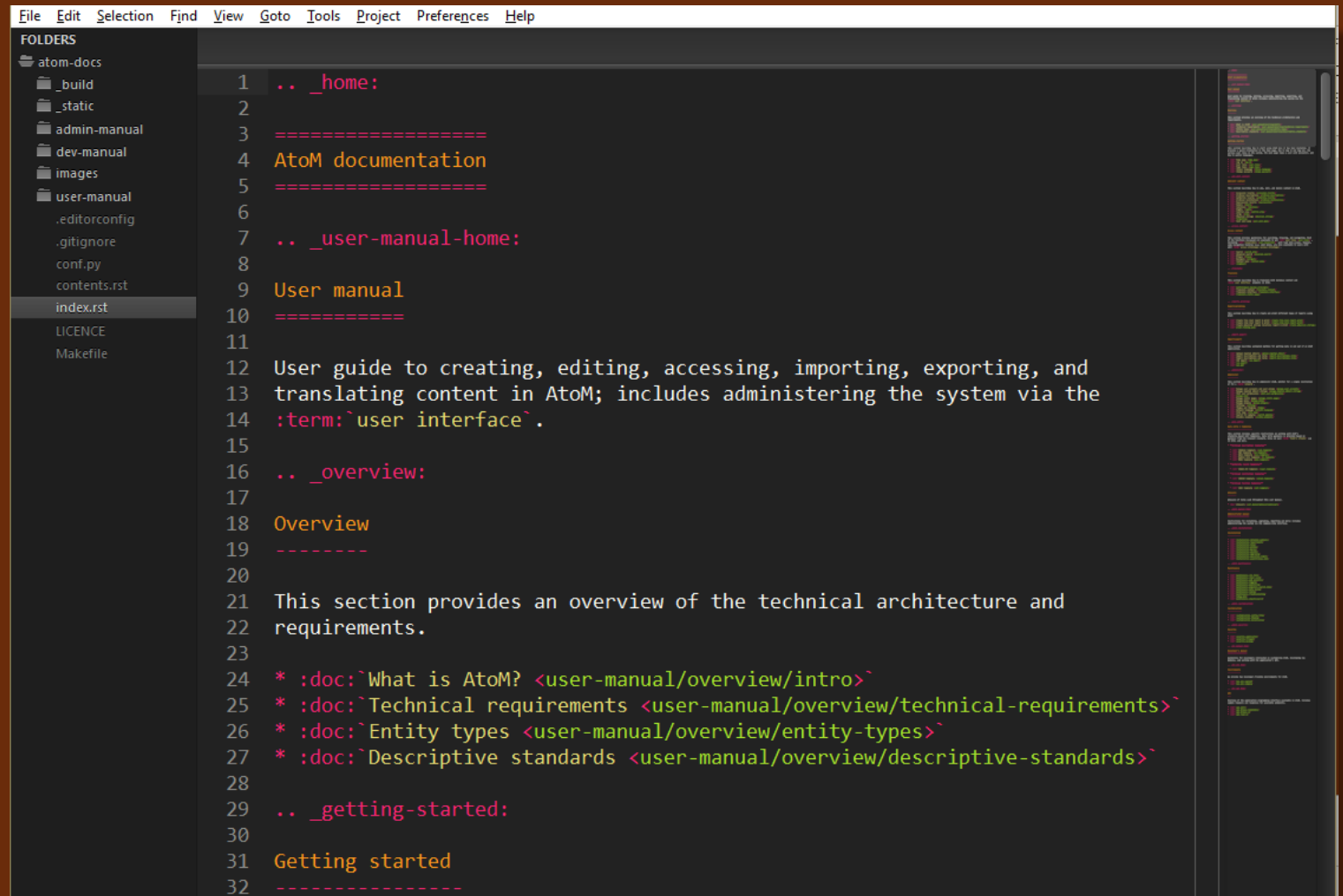
Turn on your network discovery and find Vagrant. Vagrant must be running for this to work. Add a shortcut to the atom-docs folder for easy reference in the future.



An example in Windows Explorer - double-click to drill down until you find the **atom-docs** folder

SETTING UP THE DOCS IN VAGRANT

Open a text editor (like [Notepad++](#), [Sublime](#), [Atom.io](#), etc) and drag the atom-docs folder over to it. If your text editor has the option, you can save this as a project to make things easier in the future.



The screenshot shows a text editor with a dark theme. On the left, a sidebar displays the 'FOLDERS' list for the 'atom-docs' project, including subfolders like '_build', '_static', 'admin-manual', 'dev-manual', 'images', and 'user-manual', along with files like '.editorconfig', '.gitignore', 'conf.py', 'contents.rst', 'index.rst', 'LICENCE', and 'Makefile'. The 'index.rst' file is selected and its content is displayed in the main editor area. The content is a ReStructuredText (RST) document for the AtoM documentation, featuring sections for home, user manual, overview, and getting started, with interspersed docstrings for Sphinx documentation.

```
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
atom-docs
├── _build
├── _static
├── admin-manual
├── dev-manual
├── images
├── user-manual
├── .editorconfig
├── .gitignore
├── conf.py
├── contents.rst
├── index.rst
├── LICENCE
└── Makefile

1  .. _home:
2
3  =====
4  AtoM documentation
5  =====
6
7  .. _user-manual-home:
8
9  User manual
10 =====
11
12 User guide to creating, editing, accessing, importing, exporting, and
13 translating content in AtoM; includes administering the system via the
14 :term:`user interface`.
15
16 .. _overview:
17
18 Overview
19 -----
20
21 This section provides an overview of the technical architecture and
22 requirements.
23
24 * :doc:`What is AtoM? <user-manual/overview/intro>`
25 * :doc:`Technical requirements <user-manual/overview/technical-requirements>`
26 * :doc:`Entity types <user-manual/overview/entity-types>`
27 * :doc:`Descriptive standards <user-manual/overview/descriptive-standards>`
28
29 .. _getting-started:
30
31 Getting started
32 -----
```


CONFIGURE YOUR GIT IDENTITY

Configure your global username and email

```
> git config --global user.name "Jane Doe"
```

```
> git config --global user.email janedoe@example.com
```

Use the same username and email you signed up with on GitHub. You can check these later with the following (along with a bunch of other settings):

```
> git config --list
```

Read more about setting up git: <https://git-scm.com/book/en/v2/Getting-Started-First-Time-Git-Setup>

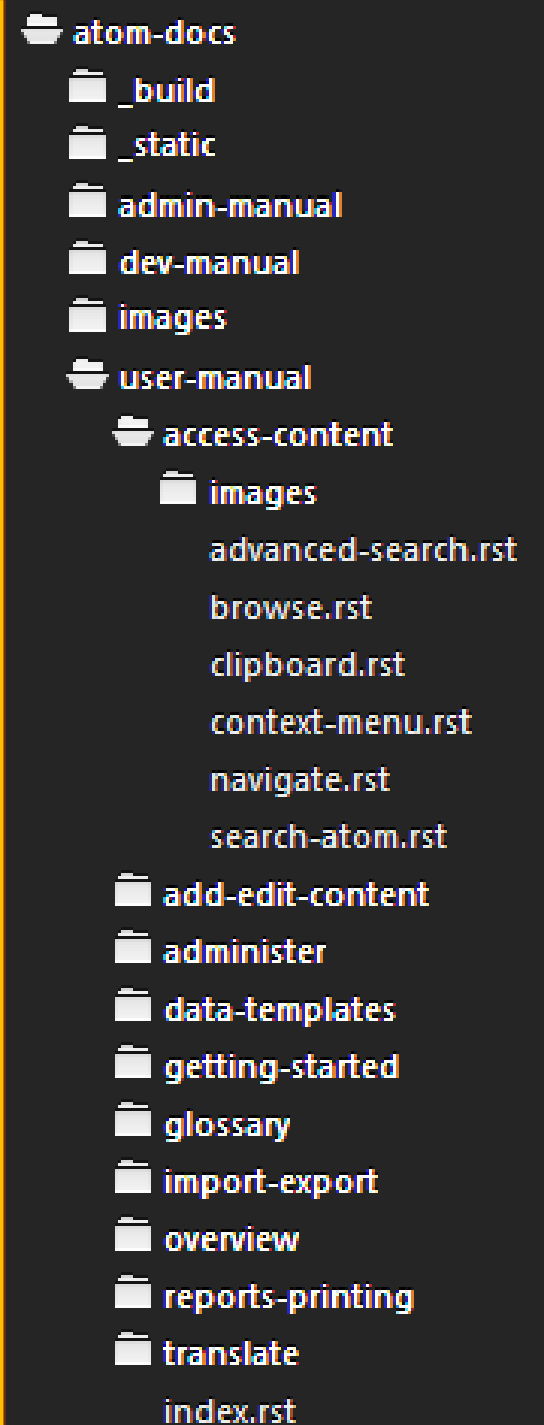
MAKING A NEW DOC PAGE

In your text editor, add a new file. Save it with an **.rst** extension.

Each manual (User, Admin, Dev) is organized into sections - these are represented as folders in the project. Each section also has its own subdirectory for the images used in that section of the manual.

Make and save your file in the section to which you want to add it. Use lower case, and dashes instead of spaces.

Add content to your new file! Make sure there is a section anchor at the top of the page so we can link to it later.



Now we need to add it to the **toctree** (table of contents tree) of the AtOM docs project - this tells Sphinx how to order the documents.

Each manual (User, Admin, Dev) has its own **index.rst** file. Open the one belonging to the manual to which you want to add the new page.

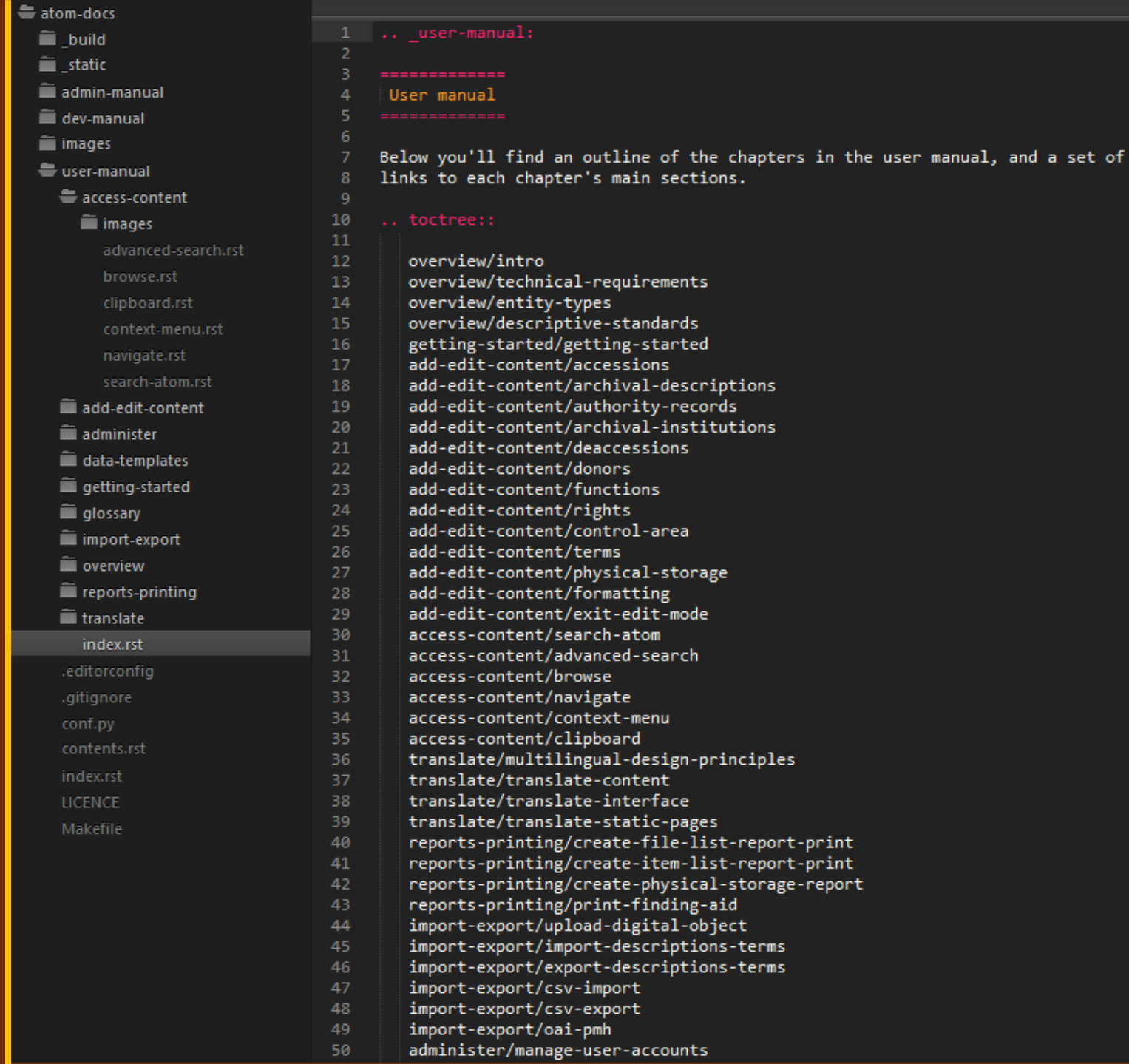
Depending on what subdirectory you placed your new file in, add an entry, based on how the files should be ordered if you were reading the docs like a book. Format:

subdirectory/filename

Remember to save this file after!

More about the [toctree](http://www.sphinx-doc.org/en/stable/markup/toctree.html#toctree-directive):

<http://www.sphinx-doc.org/en/stable/markup/toctree.html#toctree-directive>



```
1 .. _user-manual:
2
3 =====
4 User manual
5 =====
6
7 Below you'll find an outline of the chapters in the user manual, and a set of
8 links to each chapter's main sections.
9
10 .. toctree::
11
12     overview/intro
13     overview/technical-requirements
14     overview/entity-types
15     overview/descriptive-standards
16     getting-started/getting-started
17     add-edit-content/accessions
18     add-edit-content/archival-descriptions
19     add-edit-content/authority-records
20     add-edit-content/archival-institutions
21     add-edit-content/deaccessions
22     add-edit-content/donors
23     add-edit-content/functions
24     add-edit-content/rights
25     add-edit-content/control-area
26     add-edit-content/terms
27     add-edit-content/physical-storage
28     add-edit-content/formatting
29     add-edit-content/exit-edit-mode
30     access-content/search-atom
31     access-content/advanced-search
32     access-content/browse
33     access-content/navigate
34     access-content/context-menu
35     access-content/clipboard
36     translate/multilingual-design-principles
37     translate/translate-content
38     translate/translate-interface
39     translate/translate-static-pages
40     reports-printing/create-file-list-report-print
41     reports-printing/create-item-list-report-print
42     reports-printing/create-physical-storage-report
43     reports-printing/print-finding-aid
44     import-export/upload-digital-object
45     import-export/import-descriptions-terms
46     import-export/export-descriptions-terms
47     import-export/csv-import
48     import-export/csv-export
49     import-export/oai-pmh
50     administer/manage-user-accounts
```

FOLDERS

atom-docs

_build

_static

admin-manual

dev-manual

images

user-manual

access-content

images

advanced-search.rst

browse.rst

clipboard.rst

context-menu.rst

navigate.rst

search-atom.rst

add-edit-content

administer

data-templates

getting-started

glossary

import-export

overview

reports-printing

translate

index.rst

.editorconfig

.gitignore

conf.py

contents.rst

index.rst

LICENCE

Makefile

```
1 |.. _home:
2
3 |=====
4 |AtoM documentation
5 |=====
6
7 |.. _user-manual-home:
8
9 |User manual
10 |=====
11
12 |User guide to creating, editing, accessing, importing, exporting, and
13 |translating content in AtoM; includes administering the system via the
14 |:term:`user interface`.
15
16 |.. _overview:
17
18 |Overview
19 |-----
20
21 |This section provides an overview of the technical architecture and
22 |requirements.
23
24 |* :doc:`What is AtoM? <user-manual/overview/intro>`
25 |* :doc:`Technical requirements <user-manual/overview/technical-requirements>`
26 |* :doc:`Entity types <user-manual/overview/entity-types>`
27 |* :doc:`Descriptive standards <user-manual/overview/descriptive-standards>`
28
29 |.. _getting-started:
30
31 |Getting started
32 |-----
33
34 |This section describes how to start using AtoM once it has been installed. It
35 |provides a quick orientation to the home page, how to log in and log out, the
36 |various user roles in the system, the main page types a user will encounter, and
37 |how to switch languages.
38
39 |* :ref:`Home page <home-page>`
40 |* :ref:`Log in <log-in>`
41 |* :ref:`User roles <user-roles>`
42 |* :ref:`Page types <page-types>`
43 |* :ref:`Choose language <choose-language>`
44 |* :ref:`Change password <change-password>`
45
46 |.. _add-edit-content:
47
48 |Add/edit content
49 |-----
50
```

Finally, we need to add a link to the documentation homepage, so people can find your new page when they arrive at the AtoM documentation.

There is **an index.rst** file in the root atom-docs directory - you can add the link in the appropriate section there, and save.

REVIEW YOUR WORK LOCALLY

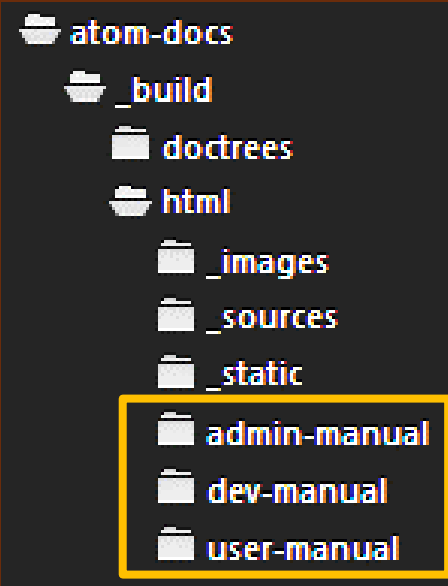
Hey remember this?

Rebuild the local HTML files based on your local copy

```
> make html
```

HTML versions of your files will be generated and placed in:

atom-docs/_build/html/



You can use these local HTML files to preview and proof your work in your web browser before submitting it

REVIEW YOUR WORK LOCALLY

When you run the make command, Sphinx will spit out warning or errors if you have messed up the syntax or structure - review these carefully and resolve any issues that relate to your work before proceeding!

```
/home/vagrant/atom-docs/user-manual/data-templates/rad-template.rst:2323: WARNING: undefined label: premis-template (if the link has no caption the label must precede a section header)
```

An example of an error - a ref link has been made with no related title and section anchor. The docs throw a warning in this instance (they will still compile). Fix any warnings or errors you cause. The path tells you which file, followed by the line number.

Review → Fix → Save → make HTML → Review

COMMIT AND PUSH YOUR WORK

Review what files you have changed

```
> git status -s
```

```
[~/atom-docs] (2.4)
vagrant$ git status -s
  M index.rst
  M user-manual/index.rst
?? user-manual/data-templates/images/premis-act-area.png
?? user-manual/data-templates/images/premis-rights-basis-area.png
?? user-manual/data-templates/premis-template.rst
[~/atom-docs] (2.4)
vagrant$
```

Shows new, modified, and deleted files in your local working area. **M** = modified.
?? = new file.

This example shows a new page, **premis-template.rst**, and 2 new images, being added to the documentation, along with updates to the **index** files.

COMMIT AND PUSH YOUR WORK

Add the changes in your working tree to the index

```
> git add <file>
```

Repeat as needed for each file changed. Running **git status -s** again will show you if you have added everything: **green** means it is now in the index and ready to be committed.

```
vagrant$ git add index.rst
[~/atom-docs] (2.4)
vagrant$ git add user-manual/index.rst
[~/atom-docs] (2.4)
vagrant$ git add user-manual/data-templates/premis-template.rst
[~/atom-docs] (2.4)
vagrant$ git add user-manual/data-templates/images/
[~/atom-docs] (2.4)
vagrant$ git status -s
M index.rst
A user-manual/data-templates/images/premis-act-area.png
A user-manual/data-templates/images/premis-rights-basis-area.png
A user-manual/data-templates/premis-template.rst
M user-manual/index.rst
[~/atom-docs] (2.4)
vagrant$
```

TIP: remember, using TAB in the command-line will help you autocomplete the file paths and names, so you don't have to type them out fully

COMMIT AND PUSH YOUR WORK

Ready to commit! Add a short message explaining your work

```
> git commit -m "Message about what I've done"
```

```
vagrant$ git commit -m "Add premis rights data entry template page"
[2.4 7194f77] Add premis rights data entry template page
5 files changed, 93 insertions(+), 2 deletions(-)
create mode 100644 user-manual/data-templates/images/premis-act-area.png
create mode 100644 user-manual/data-templates/images/premis-rights-basis-area.png
create mode 100644 user-manual/data-templates/premis-template.rst
[~/atom-docs] (2.4)
vagrant$
```

In this example, I have committed my new PREMIS rights data entry page. The command line will show a summary of git's actions, including creating new files

COMMIT AND PUSH YOUR WORK

To include a longer explanation, don't use the message option:

```
> git commit
```

The editor in your command line will open a page where you add the subject (e.g. short message) at the top, and optionally, a longer description below. Read more:

<https://chris.beams.io/posts/git-commit/>

Tips on writing the commit message subject:

1. Separate subject from body with a blank line
2. Limit the subject line to 50 characters
3. Capitalize the subject line
4. Do not end the subject line with a period
5. Use the imperative mood in the subject line
6. Wrap the body at 72 characters
7. Use the body to explain what and why vs. how

COMMIT AND PUSH YOUR WORK

Run the pull rebase command one more time

```
> git pull --rebase
```

This is done in case someone else is working on the docs at the same time, and has committed changes - this way, you will pull the newest work in first and rebase your work on top of it. If nothing has changed since you last ran the pull --rebase command, then you will be told your branch is up to date.

```
vagrant$ git pull --rebase
Current branch 2.4 is up to date.
[~/atom-docs] (2.4)
vagrant$
```

COMMIT AND PUSH YOUR WORK

Ready to push your work!

```
> git push
```

As long as no one has been working on the exact same files as you, then the **push** should go through without any problems.

```
vagrant$ git push
Counting objects: 10, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (10/10), done.
Writing objects: 100% (10/10), 38.36 KiB | 0 bytes/s, done.
Total 10 (delta 6), reused 0 (delta 0)
remote: To git@github.com:artefactual/atom-docs.git
remote:    973f7ea..7194f77  2.4 -> 2.4
To git@git.artefactual.com:atom-docs.git
    973f7ea..7194f77  2.4 -> 2.4
[~/atom-docs] (2.4)
vagrant$
```

WIKIS STILL HAVE THEIR USES!

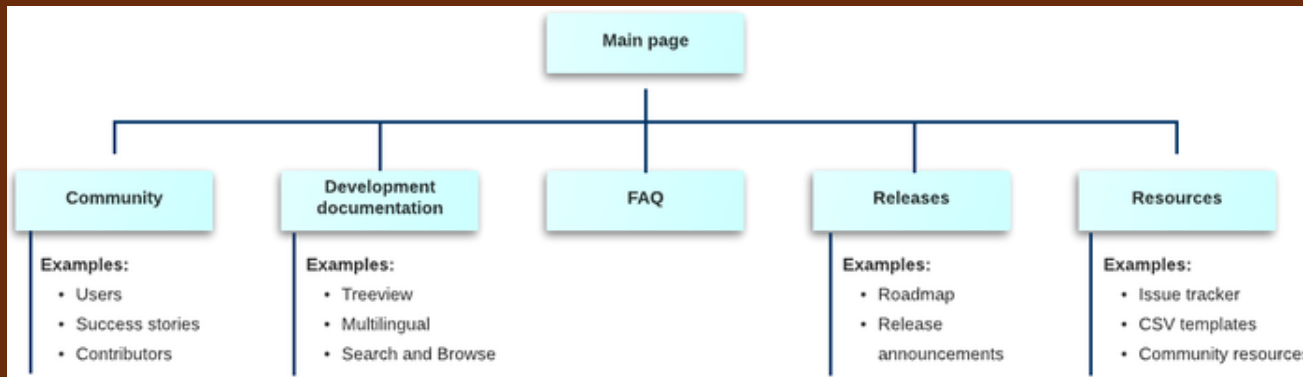
- Web-based editing
- Popular uses (e.g. Wikipedia) have made it familiar and easy to understand for users
- Good for rapidly changing content

AtoM Sphinx docs:

- User manual
- Admin manual

AtoM Wiki:

- Release notes
- User list
- Community resources
- Development documentation



THANKS!



AtoM docs: <https://www.accesstomemory.org/docs>

Documentation repo: <https://github.com/artefactual/atom-docs>

Sphinx documentation: <http://sphinx-doc.org/>

[Home](#)[Downloads](#)[Documentation](#)[Community](#)[News](#)[Wiki](#)[Demo](#)

Version 2.3 has not been released yet, and its documentation is a work in progress.

AtoM documentation

User manual

User guide to creating, editing, accessing,

Administrator manual

Instructions for installing, upgrading, importing and

Version 2.3

[\[Table of contents\]](#)

CONTENTS

AtoM documentation

- User manual