Open Geospatial Consortium

Training Session on Authoring OGC Standards with Metanorma

Part 1

Introduction to Metanorma for OGC

Meeting sponsor

MPA SINGAPORE

SLA SINGAPORE LAND AUTHORITY

GeoWorks SINGAPORE

2022-09-28

Overview

- Introduction
- Motivation
- Supported OGC document types
- Data models
- Metanorma Asciidoc

Introduction

- Metanorma is an open-source framework for writing & publishing standardization documents with the focus on semantic authoring and flexible output support.

- Maintained by Ribose in the metanorma GitHub organization
  - https://github.com/metanorma

- Supports the authoring of standards by several Standards Development Organizations (see https://www.metanorma.org), enabling cross-publishing

Motivation: Some of the problems solved by Metarnoma

- Single source of content for multiple output styles
- Long reviewing times of documents written in typical word processing software
- Documents written in typical word processing software are human error prone
- Conversion of documents to different formats can result in formatting errors
- Duplication of effort in maintenance of bibliographies

Metanorma for OGC

- “Metanorma for OGC” is the implementation of Metanorma for OGC.

- It has been approved as an official way to publish new OGC Standard documents since 2021-09-17, with Metanorma-based document templates subsequently approved by the OGC Document Team (DocTeam) SubCommittee on 2022-02-25.

- Metanorma for OGC documents are created in the Metanorma AsciiDoc format. Metanorma AsciiDoc is a textual syntax for preparing an ISO/AWI 36100 compliant document model tree which can be rendered in a variety of presentation formats.

- In OGC, the supported rendering formats are PDF, HTML, Microsoft Word, and ISO/AWI 36100 XML.

Difference between Metanorma for OGC and other Metanorma flavours

- Supports specification of OGC Standards metadata, including:
  - Document types
  - Stages
  - Identifiers
  - Authorship

- Supports specification of OGC ModSpec (OGC 08-131r3) elements through a specialized syntax
  - Requirements Classes
  - Requirements
  - Conformance Classes
  - Conformance Tests

Including **validation** of linkages between Requirements Classes and Requirements etc.

Difference between Asciidoctor and Metanorma for OGC

Feature	Ascidoctor	Metanorma for OGC
Document metadata (e.g. docnumber, keywords, etc)	Added as values in table cells	Added as document attributes (More info)
Requirements Classes	Presentation and content specified as tables	Created using a definition list, and then automatically rendered as tables. (More info)
Requirements	Presentation and content specified as tables	Created using a definition list, and then automatically rendered as tables. (More info)
Conformance Classes	Presentation and content specified as tables	Created using a definition list, and then automatically rendered as tables. (More info)
Conformance Tests	Presentation and content specified as tables	Content added as a definition list, and then automatically rendered as tables for Presentation. (More info)

https://github.com/opengeospatial/templates/tree/master/cheatsheet

OGC document types supported by Metanorma

- Abstract Specification Topic
- Best Practice
- Community Standard
- Community Practice
- Discussion Paper
- Policy
- Release Notes
- Whitepaper
- Standard
- change-request-supporting-document
- Engineering Report
- Reference Model
- Test Suite
- User Guide

Components of an OGC document

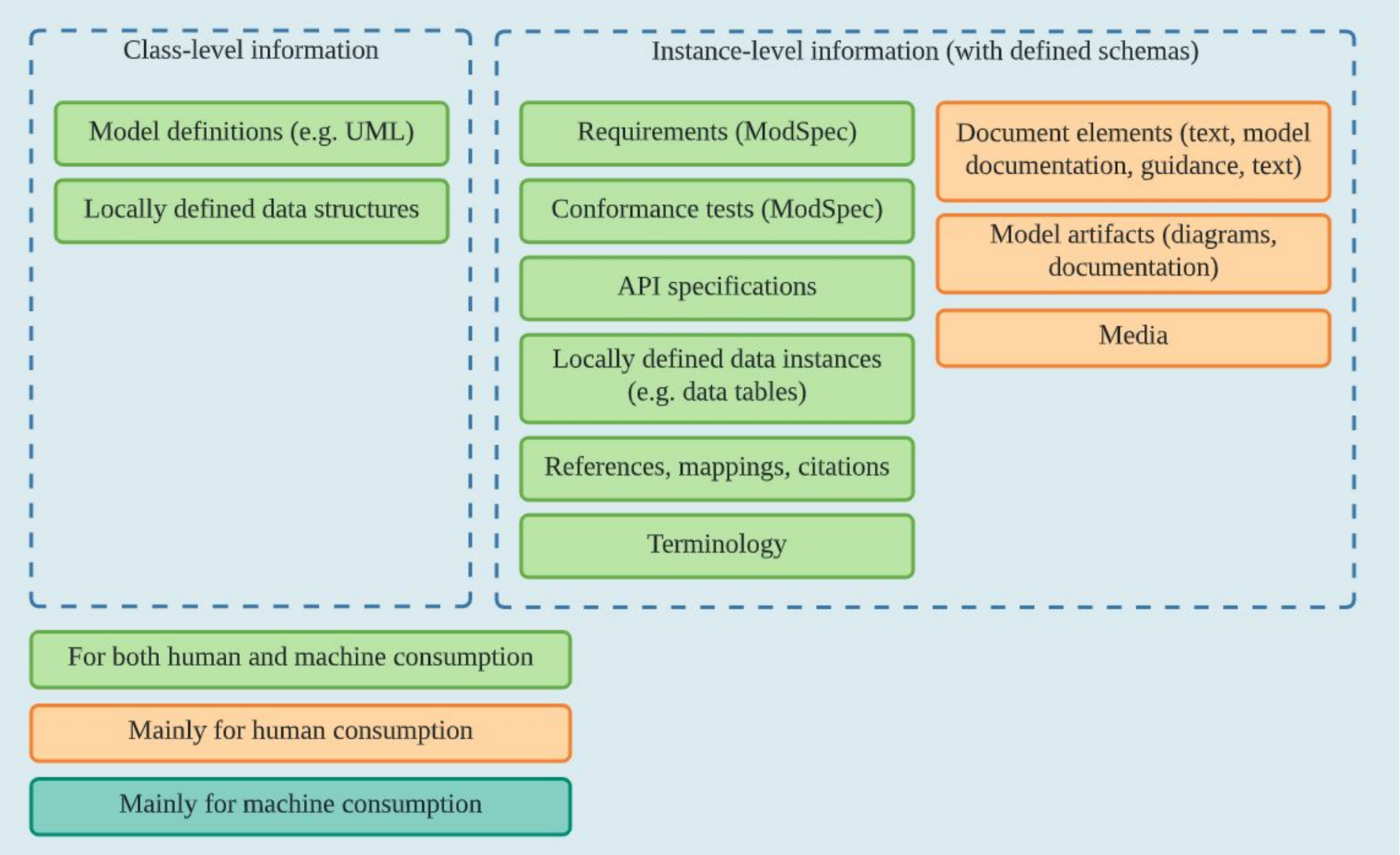


Illustration from Figure 5 of OGC 21-035r1

Models used in Metanorma

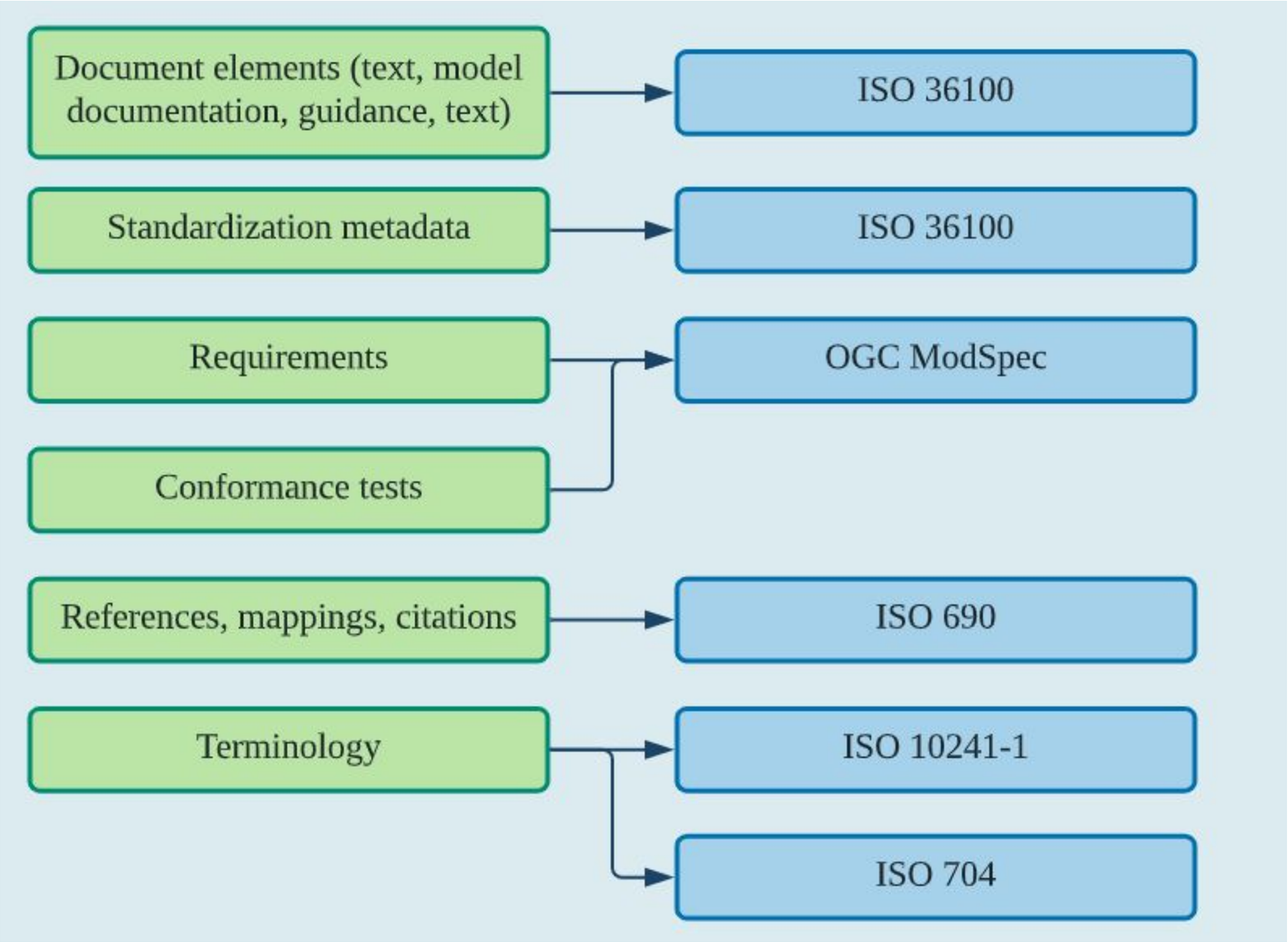
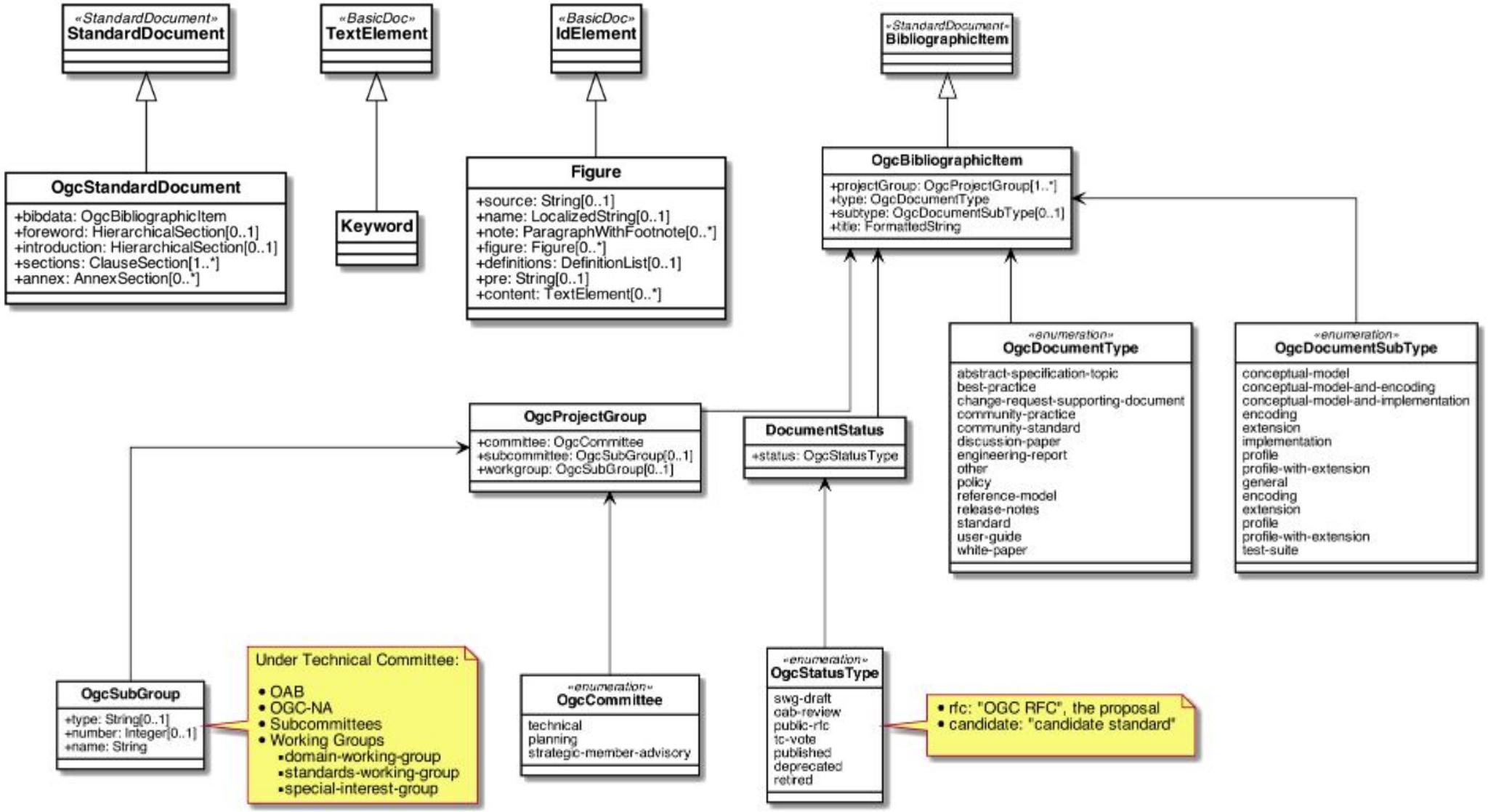


Illustration from Figure 16 of OGC 21-035r1

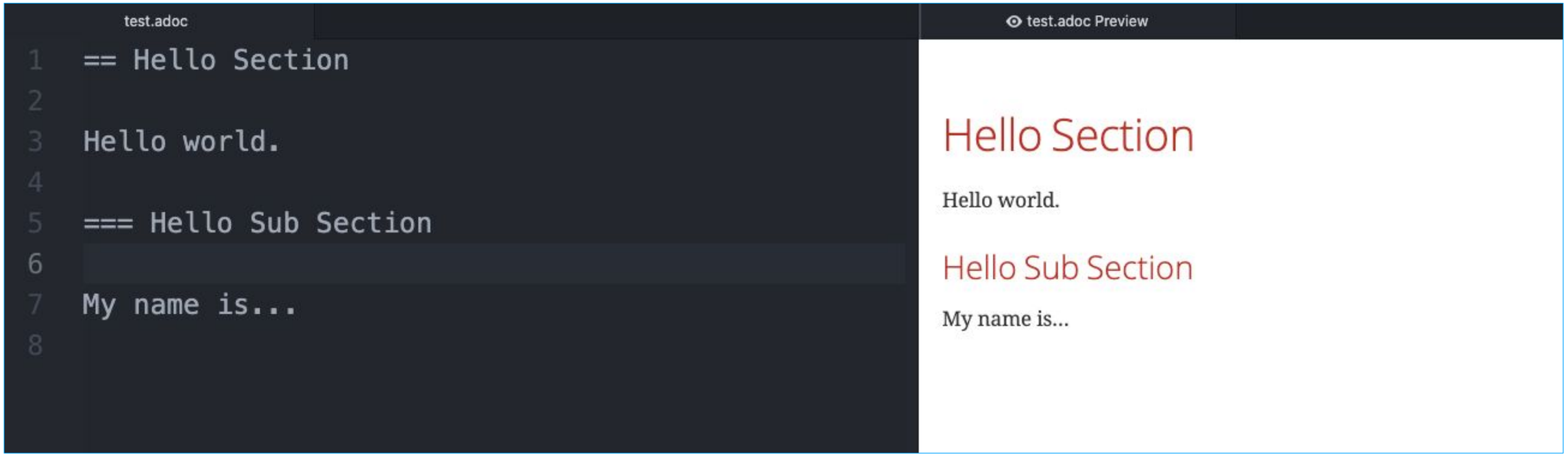
Data Model for OGC documents in Metanorma



Understanding AsciiDoc

- AsciiDoc is a plain text markup language for writing technical content.

- Documents written in Asciidoc can be converted to HTML, PDF, and other formats



Sample of tools supporting the AsciiDoc language

- Asciidoctor
- Metanorma
- AsciidocFX
- AsciidoctorJ
- Asciidoctor.js

NOTE

OGC templates require Metanorma for compilation. Do not use any other tool for compiling OGC documents.

Metanorma AsciiDoc – Basic Syntax

- Metanorma AsciiDoc is based on the AsciiDoc syntax used by Asciidoctor, and inherits most of its syntax.

- Asciidoctor is a reference implementation of AsciiDoc

Metanorma Enhancements to AsciiDoc

- Metanorma auto-numbering extends beyond typical AsciiDoc, applying to document elements including: tables, figures, formulae, and notes
- Controlled rendering of ModSpec elements (requirements, conf classes etc)
- Access to centralized bibliography of OGC, ISO, IETF, and other standards.
- Support for annotations (e.g. Editor, Reviewer, and To-do notes)

AsciiDoc features unsupported by Metanorma

- Sidebars (asides) are not supported. Instead, see annotations.

- Page breaks ("thematic break") are not supported, as Metanorma documents are focuses on semantic encoding.

- ASCII art, or preformatted text ("literal content"), are only supported in selected Metanorma flavors, as they are disallowed in most standardization organizations.

Content from this page is sourced from
https://www.metanorma.org/author/topics/document-format/

Hands-on Exercise

Pre-requisite

- Install docker, if you have not already done so. Instructions are at https://docs.docker.com/engine/install/

- Docker is a containerization software product that reserves and isolates resources on a host machine for use by an application.

- Several applications, including metanorma, can be pulled in from Docker Hub – an online registry of docker images

- Note that the following slides use the color `blue` to identify parts of a docker command that carries the actual metanorma commands

Installing Metanorma through docker

1. Open a command line terminal
2. Pull the latest docker image of metanorma by running this command from the terminal

```
docker pull metanorma/metanorma:latest
```

```
% docker pull metanorma/metanorma:latest
latest: Pulling from metanorma/metanorma
c229119241af: Pull complete
c14a196c62ee: Pull complete
365bbf24384c: Pull complete
dc13cfc653f2: Pull complete
0c1861070bf3: Pull complete
4f4fb700ef54: Pull complete
a4f7448c4c8a: Pull complete
52817fab15ee: Pull complete
d3efc0b95905: Pull complete
25c8d0468f24: Pull complete
8a904758280f: Pull complete
bf79b9964e2e: Pull complete
Digest: sha256:d0ac1b888d3fed9cded677de329fb73bce306999c97190b6d9f35fd4a1404f6a
Status: Downloaded newer image for metanorma/metanorma:latest
docker.io/metanorma/metanorma:latest
```

Creating a copy of the OGC Standards template

1. To create a copy of the template, run the following command from a terminal (i.e. from the command prompt).

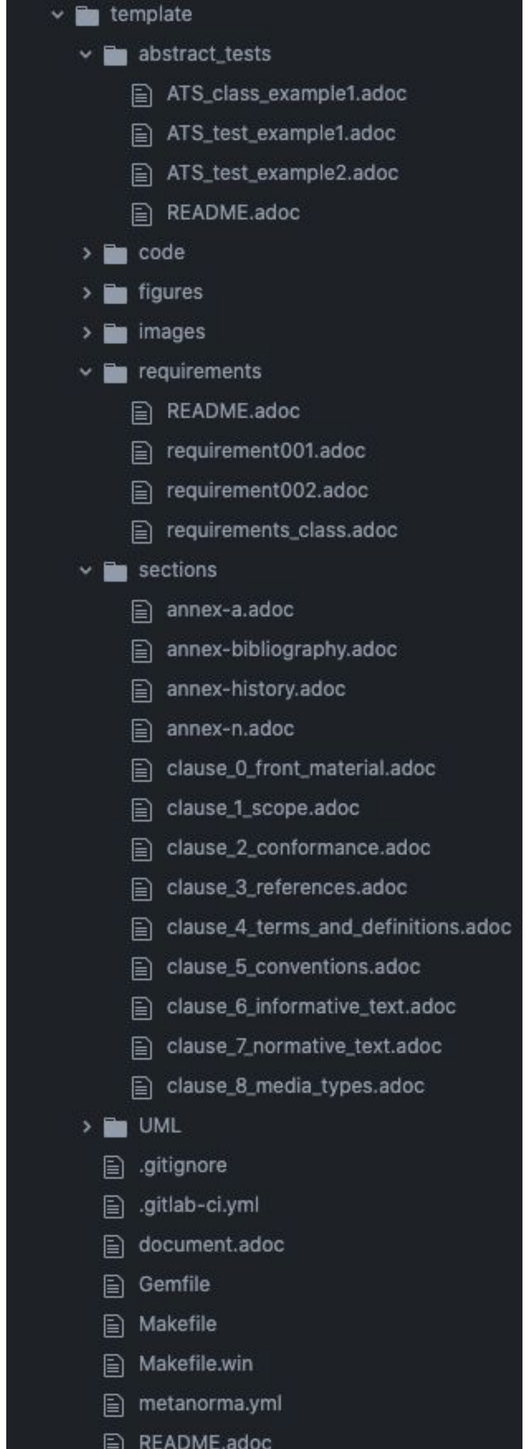
```
docker run -v "$(pwd)":/metanorma metanorma/metanorma metanorma
new -d standard -t ogc -l
https://github.com/metanorma/mn-templates-ogc
folder_for_standard
```

- NOTE: The `-d standard -t ogc` flags instruct metanorma that the template is for OGC Standards.
- NOTE: Users of Microsoft Windows should replace $(pwd) with %cd%
- NOTE: The `folder_for_standard` value can be replaced with whatever you would like to be the name of the folder that contains the copy of the template.

Understanding the template

- The template for Standards documents is organized as a folder of asciidoc files, with nested folders for sections, abstract tests, requirements and other resources.

- The main parts are:
  - document.adoc (it can be renamed)
  - images folder
  - sections folder
  - requirements folder
  - abstract_tests folder



Compiling a draft OGC Standard with a docker-containerized Metanorma instance

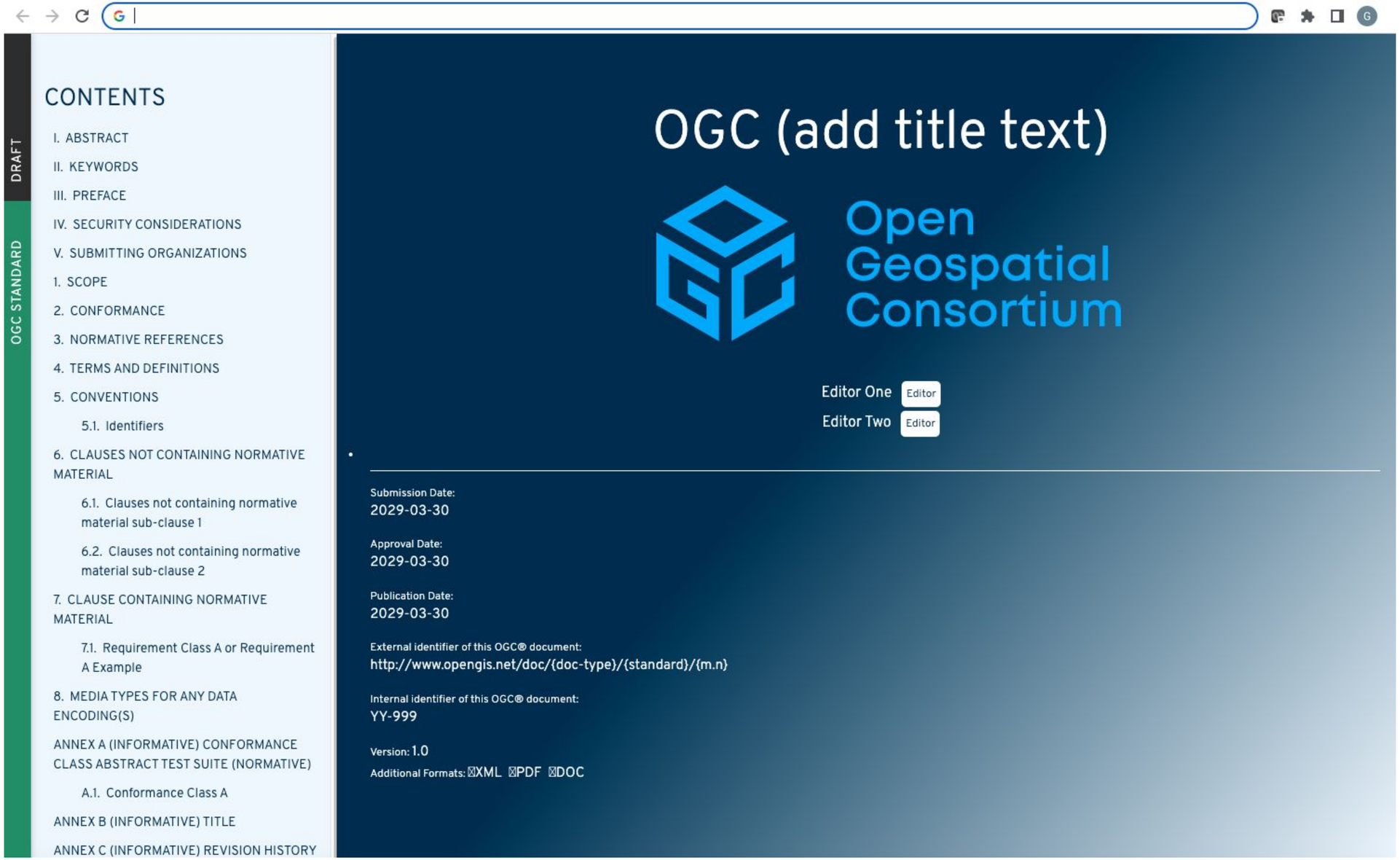
- To convert the draft standard from AsciiDoc format to HTML and PDF formats, we use the metanorma software to compile the document.

1. From the folder containing the `document.adoc` file, run the following command.

```
docker run -v "$(pwd)":/metanorma -v
${HOME}/.fontist/fonts/:/config/fonts metanorma/metanorma
metanorma compile --agree-to-terms -t ogc -x
xml,html,doc,pdf document.adoc
```

- NOTE: Users of Microsoft Windows should replace $(pwd) with %cd%

Screenshot of generated HTML document



Editing the document metadata

- Title (=)
- Metanorma document class (mn-document-class)
- Document number (docnumber)
- Document type (doctype), and optionally subtype (docsubtype)
- Document stage (status)
- Committee (committee)
- Author or editor (fullname)
- Version number (draft)
- Submitted date (received-date)
- Approval date (issued-date)
- Publication date (published-date)
- Keywords (keywords)
- Submitting organizations (submitting-organizations)

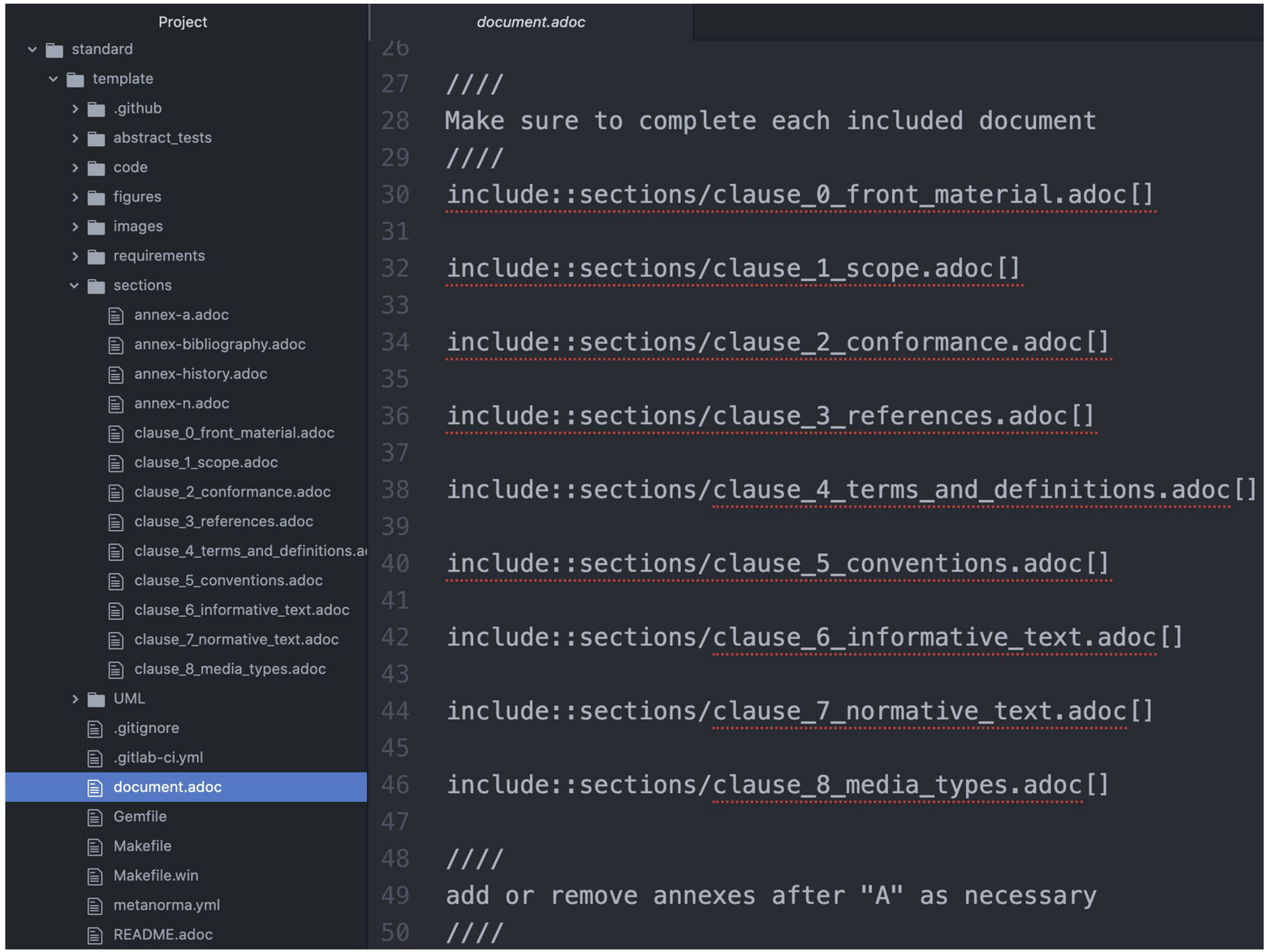
document.adoc

```
= OGC (add title text)
:doctype: standard
:encoding: utf-8
:lang: en
:status: draft
:committee: technical
:draft: 3.0
:external-id: http://www.opengis.net/doc
:docnumber: YY-999
:received-date: 2029-03-30
:issued-date: 2029-03-30
:published-date: 2029-03-30
:fullname: Editor One
:fullname_2: Editor Two
:docsubtype: Interface
:keywords: ogcdoc, OGC document, API, op
:submitting-organizations: Organization
:mn-document-class: ogc
:mn-output-extensions: xml,html,doc,pdf
:local-cache-only:
:data-uri-image:
:pdf-uri: ./document.pdf
:xml-uri: ./document.xml
:doc-uri: ./document.doc
```

Organizing the document to your liking

Use the include directive to import source files into the index/main asciidoc file

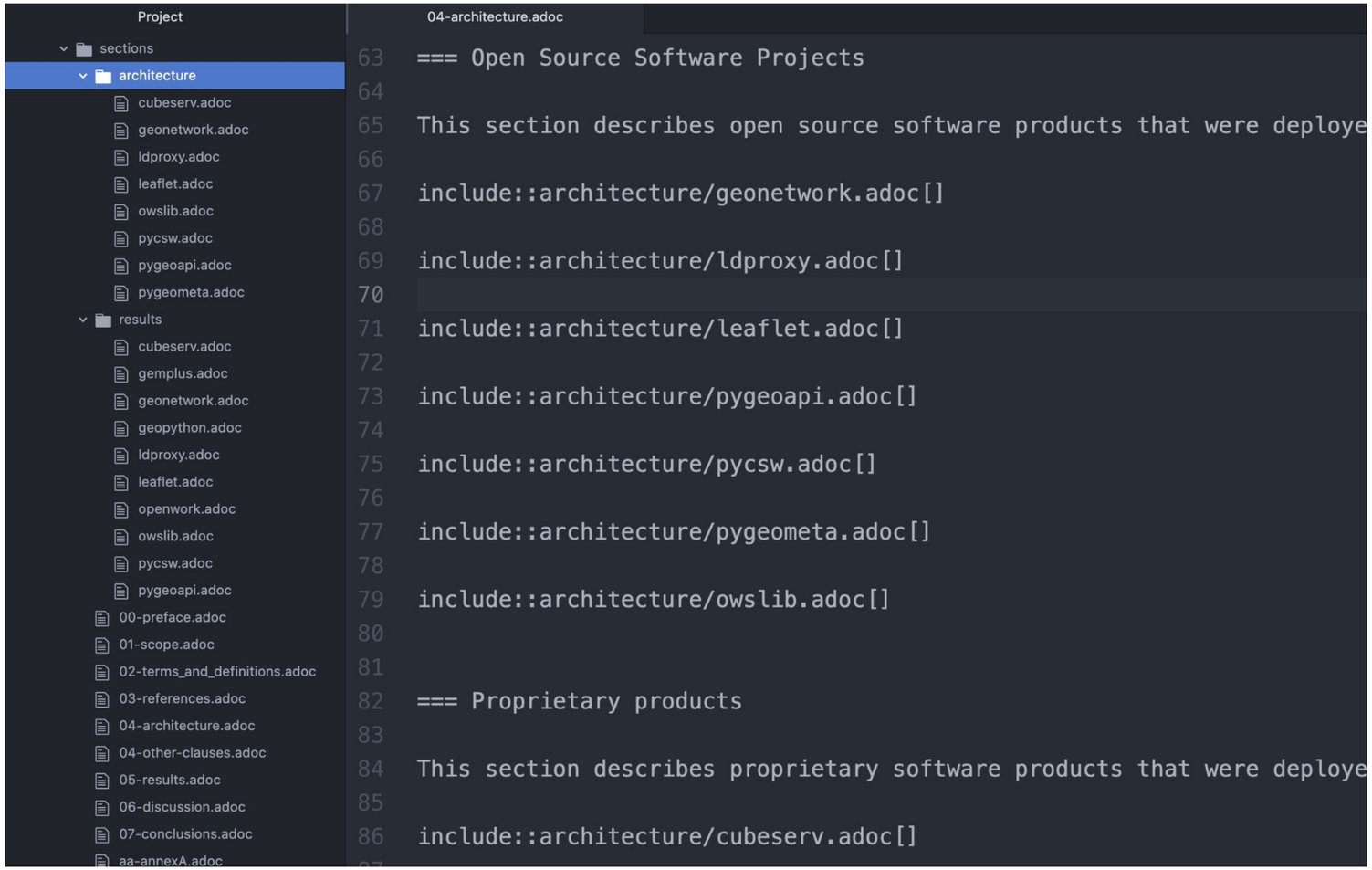
NOTE: "include:" lines must be separated with blank lines unless the sections can be stuck together. This is a requirement of AsciiDoc (inherited by Metanorma).



TIP: Organizing the document for editing as a team

Split the informative/normative section files into multiple files (e.g. one for each subsection)

Makes it possible for multiple editors to add content simultaneously, in parallel



Thank You

Community

500+ International Members
110+ Member Meetings
60+ Alliance and Liaison partners
50+ Standards Working Groups
45+ Domain Working Groups
25+ Years of Not for Profit Work
10+ Regional and Country Forums

Innovation

120+ Innovation Initiatives
380+ Technical reports
Quarterly Tech Trends monitoring

Standards

65+ Adopted Standards
300+ products with 1000+ certified implementations
1,700,000+ Operational Data Sets
Using OGC Standards