

# Open Contracting Data Standard Governance & Revision Control Process

This is a draft for discussion. Updated December 2nd 2015. Please use the comments feature to raise any discussion points.

This draft is open for comment until 5th January 2016.

Following feedback on this draft, and based on that feedback, we anticipate that:

- We will create a final revision control document, to include as part of future documentation. This will include checklists for each stage of the revision process.
- We will start the 1.1 revision process.
- Appropriate groups (peer-reviewers, standard governance working group) will be established.

## Contents

[Introduction](#)

[Stewardship and governance](#)

[Intellectual property](#)

[Governance principles](#)

[Versioning and upgrade process](#)

[Versions](#)

[Revision process](#)

[Proposals](#)

[Prioritisation](#)

[Prioritisation review](#)

[Development & docs](#)

[Peer-review](#)

[Revisions](#)

[Release](#)

[Deprecation policy](#)

[Support policy](#)

[Annex: Worked Example of the Version 1.1 Process](#)

[Proposals](#)

[Prioritisation](#)

[Review](#)

[Development & docs](#)

[Peer review stage \(2 months - begins 2nd May 2016\)](#)

[Release of 1.1 \(1st August 2016\)](#)

[Subsequently](#)

## Introduction

The Open Contracting Data Standard (OCDS) has many stakeholders: governments (procuring entities, monitoring & oversight authorities, project managers and policy makers), the private sector, and civil society organizations. The needs and interests of these stakeholders, as publishers and users of the data, are varied. As OCDS develops over time, with updated versions and new features, it is important that a diverse group of stakeholders are engaged in the process.

This document consults on revision processes for OCDS

## Version 1.1 and beyond

OCDS was initially developed through a year long iterative process, resulting in version 1.0 in November 2014. Over 2015 many organisations have been piloting use of the standard.

In Q1 & Q2 2016, we are working towards a first revised version of the standard. This is intended to fix a number of bugs identified through wider adoption of the standard over recent months, and will add a number of features which did not make it into 1.0.

This document outlines a process for management of changes to OCDS during the move from version 1.0 to 1.1.

Following this revision process, the Open Contracting Partnership will evaluate either:

- Continuing to operate a self-managed revision process;
- Submitting OCDS into a formal standards process, such as W3C community group, OASIS or other body;

We welcome views on this approach of short-term self-managed revisions, followed by exploration of formal standards processes.

## Stewardship and governance

The Open Contracting Partnership (OCP) was established as an independent non-profit in early 2015, and acts as the lead steward of the Open Contracting Data Standard.

OCP is led by an executive director, and is supported by a multi-stakeholder advisory board with representation from governments, multilateral organisations, academia, NGOs and business.

The OCDS **technical team** work under contract to the Open Contracting Partnership, under the direction of the OCP Data & Engagement manager, providing a help desk service, and responsible for the day-to-day management of the standard documentation and validation tools. The technical team can be contacted via [data@open-contracting.org](mailto:data@open-contracting.org)

In the pursuit of a consensus and **community-driven process**, subscribers to the [standard-discuss@open-contracting.org](mailto:standard-discuss@open-contracting.org)<sup>1</sup> discussion list and those watching and engaging with the [standard GitHub repository](#) should be kept informed at all stages about planned revisions to OCDS, and should be offered clear and timely opportunities to input and comment.

To ensure the relevance, quality and effective implementation of proposed updates to the standard, new version releases will be subjected to a process of **peer review** with invited reviewers from publisher and user communities, and an open review process.

A lightweight **standard governance working group**, made up of representatives from OCP staff, the multi-stakeholder advisory board, and the technical team will be responsible for giving final approval to formal upgrades of the standard and ensuring the processes in this document have been properly carried out.

## Intellectual property

The Open Contracting Data Standard is the intellectual property of the Open Contracting Partnership. The schema is provided under the **Creative Commons Attribution-NoDerivatives 4.0 International** license<sup>2</sup>, with accompanying documentation under a **Creative Commons Attribution 4.0 International license**.

Contributors to the standard agree to transfer any copyright in their contributions to the Open Contracting Partnership, in order that it is held in trust as part of the standard.

No content infringing upon third-party Intellectual Property Rights will be included in the standard.

---

<sup>1</sup> Join by sending an email to [standard-discuss+subscribe@open-contracting.org](mailto:standard-discuss+subscribe@open-contracting.org)

<sup>2</sup> <https://creativecommons.org/licenses/by-nd/4.0/> - The No Derivatives terms is in line with the [W3C Document Notice](#) which does not permit derivatives in order to prevent interoperability problems.

## Governance principles

We are committed to the [Open Stand principles](#) for standards development<sup>3</sup>. The Open Contracting Data Standard will be developed with:

- **Due process.** Decisions made with equity and fairness among participants. Through an open process for submitting issues, extensions and requests for updates, no one party will dominate or guide standard development. All processes will be transparent and opportunities will exist to appeal decisions. Processes for periodic standards review and updating are well defined in this document.
- **Broad consensus.** The process will allow for all views to be considered and addressed, such that agreement can be found across a range of interests.
- **Transparency.** We will provide advance public notice of proposed standards development activities, the scope of work to be undertaken, and conditions for participation. Easily accessible records of decisions and the materials used in reaching those decisions will be provided. Public comment periods will be provided before final standards approval and adoption.
- **Balance.** Standards activities will not exclusively dominated by any particular person, company or interest group.
- **Openness.** The Open Contracting Data Standard processes are open to all interested and informed parties.

In the future, the Open Contracting Data Standard may be submitted to a formal standardisation body, such as the World Wide Web Consortium, or OASIS. Before such a decision is made, a model of community-driven governance shall be established based on an **open and consensus-based processes** for updating the standard.

---

<sup>3</sup> <https://open-stand.org/about-us/principles/>

## Versioning and upgrade process

Over time, changes will be needed to the standard, including addition of new codes and fields, and occasionally involving changes to existing fields and structures.

The revision process should ensure:

- The consequences of any change for different stakeholders are identified and considered;
- It is clear why changes are needed, and that there is broad support for any proposed changes;
- Changes are easy to identify and are transparent, and publishers, users and intermediaries have clear documentation to allow them to update their data and tools;

Changes to the OCDS schema should be made periodically, with the version number of the standard incremented to indicate that changes have been made, and a change-log maintained.

## Versions

Distinct **branches** of the standard will be maintained within Github for each version.

Branches can be in one of two states:

- **Development** - indicated by a -dev suffix (e.g. 1.1-dev)  
Both schema and documentation on a development branch can be updated and should only be implemented on an experimental basis.
- **Live** - with no suffix ( e.g. 1.0)  
Only documentation updates are permitted on a live branch. All documentation changes must be reviewed to ensure they do not make any changes to the meaning of the standard.

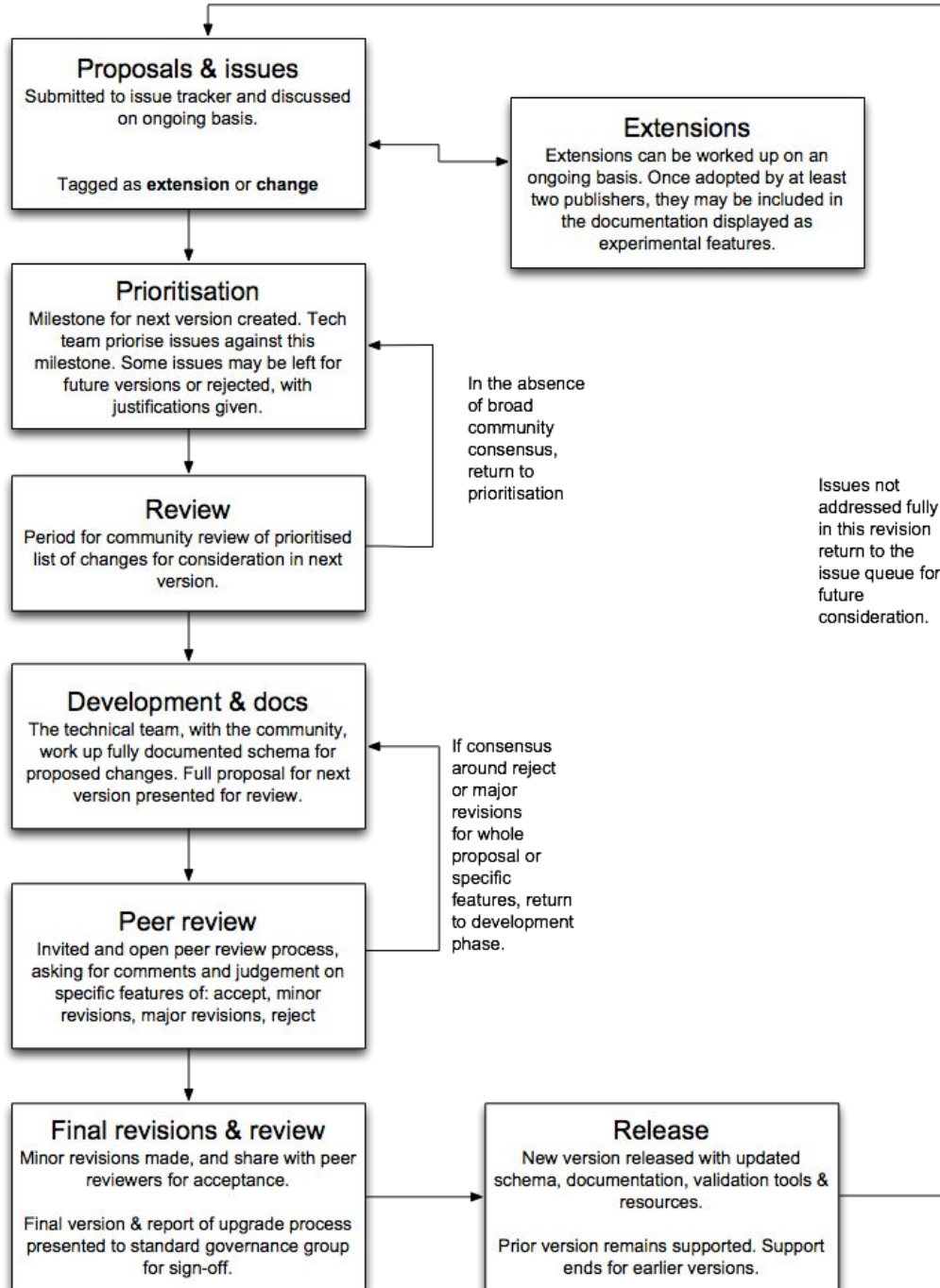
Semantic Versioning practices will be used to distinguish between:

- **Major versions** which make backwards-incompatible API changes
- **Minor versions** which add functionality in a backwards-compatible manner

These are captured by a version number in the format MAJOR.MINOR

## Revision process

To release a new minor or major version upgrade will involve a number of stages outlined in the flowchart below, and described in more depth in the following sections.



## General principles:

- **Publicity:** All stages of the revision process will be announced via the standard-discuss mailing list, and through GitHub issues. These are the formal channels for notification during the process.
- **Consensus:** All processes should aim towards gaining community consensus for changes.

The technical team are responsible for generating key documentation during the process, but should always be guided by community consensus, submitting all decisions for discussion.

- **Appeal:** Any party may appeal against decisions made during the process by writing to the standard governance working group (address tbc.). The working group has the authority to reject proposed revisions on the standard in response to appeals.

## Proposals

Changes to OCDS can be proposed at any point via the [standard's issue tracker](#) either as issues for discussion, or pull requests with a clear description of the proposed change.

Contributors are encouraged to raise discussions prior to pull requests to seek consensus on proposed changes.

Changes may be proposed as:

- **Extensions** - which add new features to the standard.
- **Changes** - such as updated field definitions or codelist entries.

If there are at least two parties interested in using an extension, and following discussion of the extension draft, then it may be displayed in the current version of the documentation as an 'experimental feature'.

## Prioritisation

The technical team, with reference to community views, identify change proposals and extensions which should be considered for adoption in the next version of the standard, assigning these to milestones in the issue tracker where they are open for discussion.

Periodically, at the start of a revision process a cut-off date for proposals will be announced with at least two weeks notice. After that date a prioritised list of updates is produced. Any new proposed extensions or changes received after this period may not be considered until the next prioritisation phase.

## Prioritisation review

The list is shared for feedback, with at least a two-week window for discussion.

Based on discussions, a final list is then proposed by the technical team with all the issues that will be taken forward into the rest of the process. A proposals that has made it this far



may or may not make it into the final upgrade. As the proposal is worked into final concrete examples and schema changes further issues may arise.

### Development & docs

The technical team, working with community members, will work on a development branch to prepare updates to the schema, documentation and codelists, according to the prioritised list.

This stage is likely to involve broad community engagement and discussion of specific decisions through GitHub issues.

At the point where all open issues are suitably addressed, the development branch can be submitted for peer review.

### Peer-review

The updated schema, documentation along with a change log and narrative description of the changes will be released for peer-review.

A group of invited reviewers, representing different stakeholders, and including data publishers and users, will be asked to complete a full review of the changes, and to submit:

- A judgement on whether the overall upgrade, and/or specific changes should be **accepted, accepted with minor changes, substantially revised, or rejected.**
- Comments on each request for revisions or rejection.

All reviews, with reviewer names, will be published. Community members may also submit their own reviews of the whole revision, or specific elements. The minimum period for peer-review is one month.

### Revisions

The technical team, with reference to the working group as appropriate, should evaluate reviews, and decide whether the whole upgrade, or specific features thereof, need to be revised, rejected or postponed to future processes.

If only minor changes are suggested, then the revised standard can be submitted back to reviewers for a brief review period of at least two weeks.

If major changes are required, then a longer follow up review process of at least one month should be allowed for.

### Release

Once all reviewer comments have been addressed to the satisfaction of the reviewer in question, then the updated version of the standard should be submitted to the **standard governance working group** for final approval, along with a short report of the process.

Following **working group** approval, the revision branch can be set to live.

## Deprecation policy<sup>4</sup>

If a term (a class or property) is scheduled to be renamed or removed from the specification as a result of the revision process, the next major release of the specification must deprecate<sup>5</sup> the term within the schema, and the following major release must rename or remove the term from the schema, making the term obsolete. Implementations may use deprecated terms, but will receive warnings from the OCDS validator described below. Implementations may not use obsolete terms, and will receive errors from the OCDS validator.

## Support policy

Support will be offered for one prior version of the standard. Support for any earlier versions than this will be ended when a new version is released.

For example, when 1.1 is the latest release, 1.0 will be supported in the validator and other tooling. When 1.2 is released, support for 1.0 will end.

Publishers are encouraged to review each new version when it released, and to consider how they might adopt new features.

Publishers should aim to move to a new **major** version within 18 months of its release.

---

<sup>4</sup> <https://github.com/open-contracting/standard/issues/189>

<sup>5</sup> <https://en.wikipedia.org/wiki/Deprecation>

## Annex: Worked Example of the Version 1.1 Process

This worked example sets out how the process might be used for Version 1.1 of the standard, along with outline timescales.

This will not form part of the final document, but is provided to give an illustration of the required stages.

### Proposals

Proposals for changes or extensions can be submitted at anytime.

**In practice:** This means all proposals should be on GitHub. The technical team may need to help people get their submissions on there.

In order to get version to 1.1 we will set a **cut-off date** for proposals for that version.

**In practice:**

Set a date (Mon 29th Feb 2016). We also need to think about advertising the upgrade process, canvas opinion, get people to contribute their ideas, opinions on other contributions etc. It can be no work, or a lot of work. Communication channels need to be in place (is this blog, newsletter, mailing list? who does it?)

Proposals received after that date will not (necessarily) be considered for inclusion in 1.1. There may however be exceptional circumstances and the **standard governance working group** would need to approve that.

### Prioritisation

Prior to the cut-off date work and discussion on existing proposals can take place.

After the cut-off date the technical team need to decide what will be worked on and what will not be, and they need to publish those decisions.

**In practice:**

Meet by 7th March, Publish by 14th March (Meet and publish could happen within a week? Especially if the meeting took place on 7th, say)

### Review

People may disagree with those decisions - so we will set a date by which we would have published those decisions (two weeks after the cut-off date: 14th March ) and then a period of time for people to raise objections (2 weeks later). This allows people to refine the list or make a case for something that is not being considered.

**In practice:**

Comments close 30th March. This is a period of consultation. See notes below.

Publish the revised list of extensions and issues and proposals that **are part of 1.1 milestone** by 7th April

What this also does is create a very clear set of issues for the development team to work on without moving the goalposts.

### Development & docs

So now we have the ideas that will be worked into 1.1, this is the period where the technical work is done. (This could be called Feature Freeze).

### From Feature Freeze to 1.1

There is a stage here where quite often an idea needs to be explored in more detail. So e.g the issue could be 'improve location data'. At this point it may be that a proposal needs to be worked up to clarify the intention of the proposal. It's not a schema/documentation/codelist change yet - but the political and technical detail needs to be worked out and approval sought.

Key to this stage of working out/translating the will of the suggestion into a technical document/proposal that could then be implemented in the schema/documents etc

These 'fully worked up proposals' become something that a technical team could implement.

The fully worked up proposals are things that could be consulted on.

At this point we will inevitably test our communication systems, and our technical systems for displaying previews to the world and gathering feedback in a useful, open manner.

### In practice:

- Iteration 1 produced by (within 1 month - 7th May)
- Iteration 21 produced by (within 1 month - 7th June)
- Issues can still be places of conversation
- Fully worked up proposals created and ..
- .. turned into 'dev' versions of documentation, codelists, schema

### Peer review stage (2 months - begins 7th June 2016)

At this stage, we're nearing the final iteration (last 2 iterations maybe?) The idea is that everything is ready. This is it. This is 1.1 - albeit subject to review.

- Schema & documentation updated on a development branch (dev)
- Should ensure all elements are tested in practice
- Invited reviewers & open review process

Outcome: accept, accept with minor revisions, reject

**1.1 Ready for sign off (7th August) - 1 week before OCP Sign off , 2 months after Peer review stage starts**

**OCP Sign off (14th August) - 2 weeks before staging sites ready, one week after 1.1 Ready for Sign Off)**

After this we might want to give our developers 2 weeks to get the staging site ready?

**1 week before release (30th August) - 2 weeks after OCP Sign Off**

- Staging sites for all websites in place (Final version in place - nothing changes, techies make sure the sites work)

**Release of 1.1 (30th August 2016)**

- Blog post & e-mails announcing new standard;
- Change log **and** user friendly upgrade documentation;
- Updated schema published - on a GitHub branch (set as default on repository);
- Updated codelists; (not in spreadsheets)
- Updated documentation website (access to 1.1 and 1.0) - defaulting to 1.1;
- Updated example data - version 1.1 data that validates;
- Discussion trail about upgrades;
- Validator updated to work with 1.1; (CoVE) CovE monitoring the versions of data sent to it.
- Helpdesk should be ready to handle 1.1 questions/support
- Supporting tools for conversion if required;
- Updated copies of any guidance tools - including mapping spreadsheet;
- Strategy(or not) about 1.1 adoption

**Subsequently**

- Webinar launching 1.1;
- Work towards 1.2 begins.