INSPIRE Schemas - Versioning and schema repository

Jordi Escriu, Fabio Vinci, Marco Minghini, Alexander Kotsev

Governance of INSPIRE artefacts

67 MIG-T Meeting – October 7th, 2021

Objectives

- Establish practical rules for managing changes to the schemas endorsed by MIG, after going through the governance process.

- Come up with an agreed functional solution for restructuring the official repository of INSPIRE XML app. schemas:

  https://inspire.ec.europa.eu/schemas

- Update the criteria used for versioning schema releases:

  https://inspire.ec.europa.eu/XML-Schemas/Data-Specifications/2892

- Keep trace of historic / old versions of schemas in the repository, if possible maintaining them operative.

- Encourage MS to update existing implementations using the latest endorsed versions of schemas, while minimising the impact on them.

Versioning of schema releases
Proposal to maintain existing criteria (*)

Follow the **X.Y.Z pattern**, where:

Breaking changes = Non-backwards compatible
Non-breaking changes = Backwards compatible

- **X** (major version number):

  To be updated (e.g. v3.X --> v4.0) when changes to the applicable legal framework (e.g. Implementing Rules) are applied. These are always **breaking changes**.

- **Y** (minor version number):

  To be updated (e.g. v3.0.X --> v3.1) when **non-breaking changes** to the schemas are applied.

- **Z** (bug-fix version number):

  To be updated (e.g. v3.0 --> v3.0.1) when errors or bugs are fixed in the XML schemas. These can be **breaking** or **non-breaking changes**.

* Already endorsed by MIG and used in the previous years.

Restructuring the schema repository
1a. Initial proposal - Structure

am (https://inspire.ec.europa.eu/schemas/am/)

   latest ← Shortcut pointing to the latest version of the schema endorsed by MIG

   deprecated

      3.0.0

         AreaManagementRestrictionRegulationZone.xsd ← version 3.0.0

      3.0.1

         AreaManagementRestrictionRegulationZone.xsd ← version 3.0.1

      4.0.0

         AreaManagementRestrictionRegulationZone.xsd ← version 4.0.0

   3.0

      AreaManagementRestrictionRegulationZone.xsd ← version 3.0.2

   4.0

      AreaManagementRestrictionRegulationZone.xsd ← version 4.0.1 (Last endorsed schema)

Restructuring the schema repository
1a. Initial proposal - Example

Release 2021.2

4.0.1 → breaking changes

- https://inspire.ec.europa.eu/schemas/**am/4.0/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.1)
- https://inspire.ec.europa.eu/schemas/**am/deprecated/4.0.0/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.0)

Release 2022.1

4.0.2 → breaking changes

- https://inspire.ec.europa.eu/schemas/**am/4.0/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.2)
- https://inspire.ec.europa.eu/schemas/**am/deprecated/4.0.1/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.1)
- https://inspire.ec.europa.eu/schemas/**am/deprecated/4.0.0/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.0)

Restructuring the schema repository
1a. Initial proposal – Pros vs. Cons

Advantages

- Structure based on OGC current practice.

- Existing implementations would not be impacted when breaking changes, non-breaking changes, or non-breaking bug-fixes are applied to the schemas.

Disadvantages

- Breaking bug-fixes may impact existing implementations – However:
  - They are corrections to the schemas, previously endorsed by MIG (i.e. accepted by MS).
  - MS may be properly warned through different mechanisms:
    - INSPIRE Reference Validator (when using deprecated versions of schemas).
    - Notifications sent time in advance to applying changes to the repository (e.g. 1 month before).

- It does not allow storing different bug-fix versions of schemas in the same folder – This forces to use the 'deprecated' folder and change the namespace for the deprecated schemas.

- Hard to maintain dependencies between deprecated schemas (schema imports), and keep them operative.

Restructuring the schema repository
1b. Variant of initial proposal - Example

Release 2021.2

4.0.1 → breaking changes

- https://inspire.ec.europa.eu/schemas/**am/4.0/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.1)

- https://inspire.ec.europa.eu/schemas/**am/deprecated/4.0/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.0)

Release 2022.1

4.0.2 → breaking changes

- https://inspire.ec.europa.eu/schemas/**am/4.0/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.2)

- https://inspire.ec.europa.eu/schemas/**am/deprecated/4.0/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.1)

- https://inspire.ec.europa.eu/schemas/**am/deprecated/4.0/**AreaManagementRestrictionRegulationZone**_4.0.0.zip** (version 4.0.0)

> - **Namespace for the deprecated schemas is maintained.**
> - **Only keeping operative the immediately previous version for bug-fixed schemas.**

Restructuring the schema repository
2. Evolved proposal - Structure

am (https://inspire.ec.europa.eu/schemas/am/) ← Last schema release: 2022.1

    latest ← Shortcut pointing to the latest version of the schema endorsed by MIG

    3.0

        AreaManagementRestrictionRegulationZone.xsd ← version 3.0.2

    4.0

        AreaManagementRestrictionRegulationZone.xsd ← version 4.0.2 (Last endorsed schema)

2021.2/am (https://inspire.ec.europa.eu/schemas/2021.2/am/) ← Release: 2021.2

    3.0

        AreaManagementRestrictionRegulationZone.xsd ← version 3.0.2

    4.0

        AreaManagementRestrictionRegulationZone.xsd ← version 4.0.1

2021.1/am (https://inspire.ec.europa.eu/schemas/2021.1/am/) ← Release: 2021.1

    3.0

        AreaManagementRestrictionRegulationZone.xsd ← version 3.0.2

    4.0

        AreaManagementRestrictionRegulationZone.xsd ← version 4.0.0

Restructuring the schema repository
2. Evolved proposal - Example

Release 2021.2

4.0.1 → breaking changes

- https://inspire.ec.europa.eu/schemas/**am/4.0/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.1)
- https://inspire.ec.europa.eu/schemas/**2021.1/am/4.0/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.0)

Release 2022.1

4.0.2 → breaking changes

- https://inspire.ec.europa.eu/schemas/**am/4.0/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.2)
- https://inspire.ec.europa.eu/schemas/**2021.2/am/4.0/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.1)
- https://inspire.ec.europa.eu/schemas/**2021.1/am/4.0/**AreaManagementRestrictionRegulationZone.xsd (version 4.0.0)

Restructuring the schema repository
2. Evolved proposal – Pros vs. Cons

Advantages

- Same advantages than the 'Initial' proposal + …

- It allows storing all bug-fix versions of schemas, one in each historic schema release folder - Namespace for deprecated schemas remain the same.

- Easier to maintain dependencies between schemas (schema imports), and keep them operative - Automatic replaces possible when managing a new release.

Disadvantages

- As in the 'Initial' proposal, breaking bug-fixes may impact existing implementations. Same 'buts' as well.

- More storage is needed at the official repository (full set of XML schemas per each release).

Technical Guidelines - Conversion from Docx to AsciiDoc

Lukasz Ziemba, Jordi Escriu, Fabio Vinci, Marco Minghini, Alexander Kotsev

Governance of INSPIRE artefacts

67 MIG-T Meeting – October 7th, 2021

TGs. Conversion Docx-AsciiDoc
Draft procedure

Objective

- Conversion: Docx to AsciiDoc

- Facilitate access and maintenance of TGs in a GitHub repository.
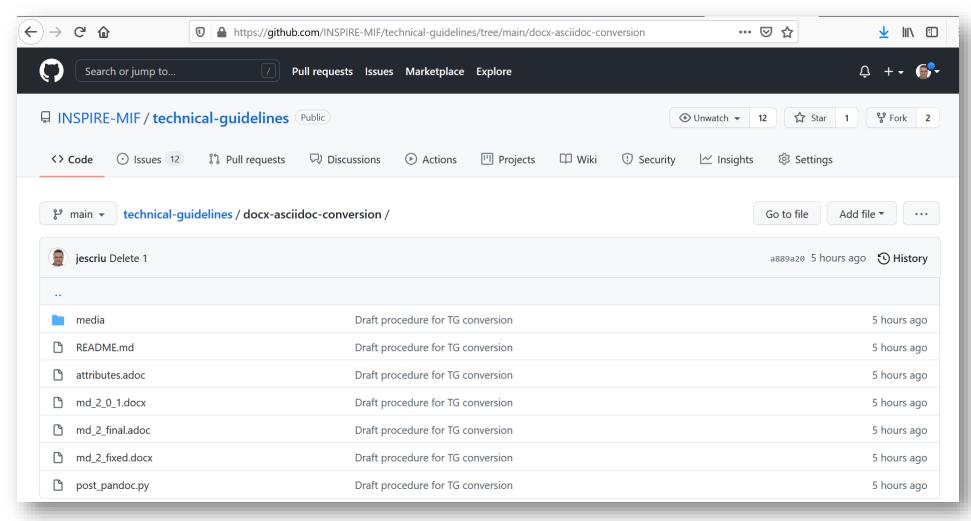
Technical details

- AsciiDoc format:
  - Supports more advanced features than Markdown; Same benefits in GitHub.
- Tools used:
  - Pandoc v2.14 + Python v3.8 script + Beautiful Soup Python library v4.9

Scope & Limitations

- **Technical Guidance for the implementation of INSPIRE dataset and service metadata based on ISO/TS 19139:2007 v2.0.1**

- Conversion procedure has not been tested with other TGs yet.

TGs. Conversion Docx-AsciiDoc
GitHub Repository

https://github.com/INSPIRE-MIF/technical-guidelines/tree/main/docx-asciidoc-conversion

TGs. Conversion Docx-AsciiDoc
Future evolution

Evolution

- Improve the process to correct small defects identified.

- Test it to make it effective for all TGs.

- Further evolve it taking into account the test results.

- Study how to proceed with complex elements (e.g. UML models).

Planning

- Start conversions in 2022.

- Priority of conversions: Based on number of change proposals in the GitHub stack and TGs importance/popularity.

Thank you!

✉ Jordi.ESCRIU@ec.europa.eu

European Commission