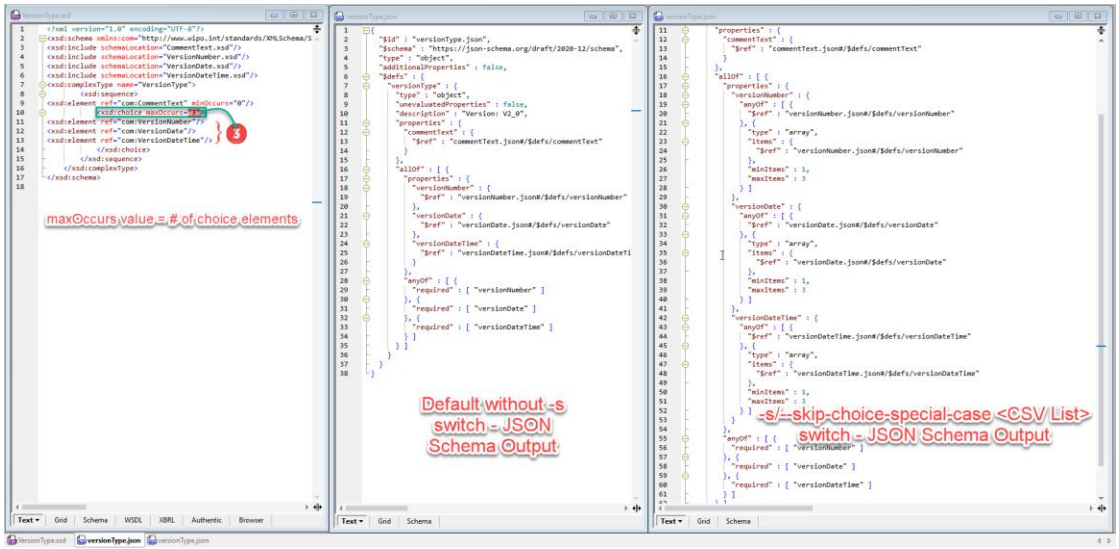
Xsd2JsonSchema Transformation Tool

Overview

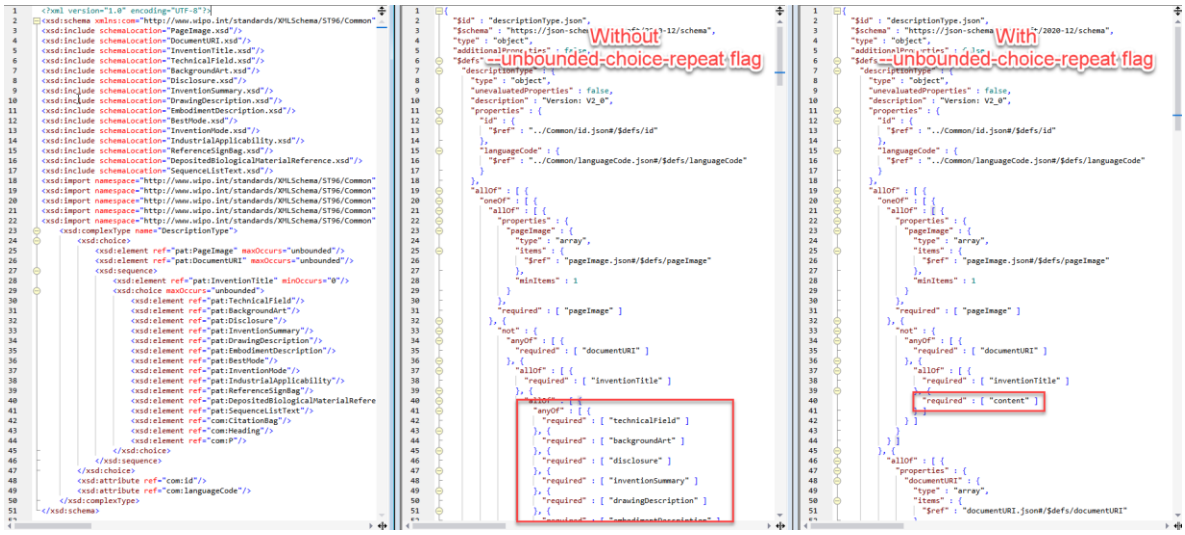
Xsd2JsonSchema converts XML Schema (XSD) files into JSON Schema (draft-2020-12) files. Each input XSD generally produces one JSON file containing a "$defs" map with all named types discovered in that XSD.

Key features

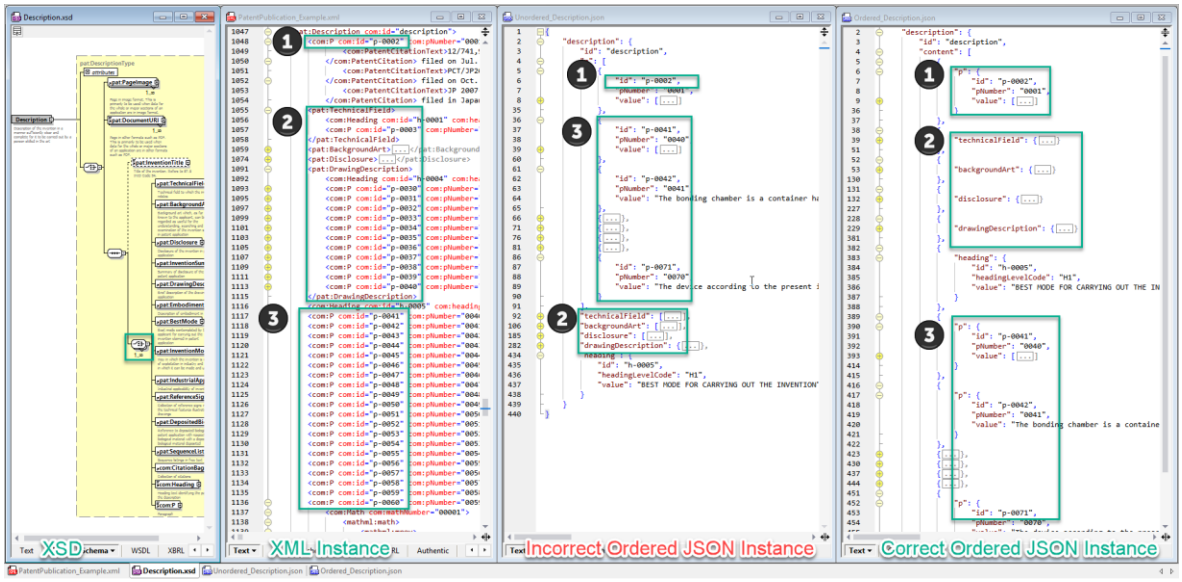
- Preserves reuse via $ref: types defined in other files are referenced rather than in-lined.
- Represents XSD complexContent extensions as allOf([baseRef, overlay]) where the overlay contains only extension-specific additions.
- Emits minimal anyOf/required alternatives for xsd:choice constructs so parent-level properties remain authoritative.
- Flattens wrapper complexTypes that contain a single repeating element (sequence with one element and maxOccurs > 1) into a direct JSON array type. Usually ST.96. Bag components.
- Maps common XSD built-in types (xsd:string, xsd:integer, xsd:date, xsd:boolean, etc.) to JSON Schema primitives and formats.
- Enhanced mixed content schema model to align with XSD model.
- Cleaner extended type definitions.
- Special case choice handling for maxOccurs = # of choice components - default is to model anyOf choice components without array model. If needed an array model use -s/--skip-choice-special-case without any list for all or for a selected components provide the CSV list.
    - See the difference in JSON Schemas below:



- Unbounded choice handling for maxOccurs = "unbounded" - default is to flatten choice components as an individual arrays. This is needed for content like models where same properties might be repeated multiple times mixed with other properties. If needed, use -u/--unbounded-choice-repeat with a list for a selected components (provide the CSV list file if the list has too many entries).
    - See the difference in JSON Schemas below:



    - See the difference in JSON Instances below:



JDK Requirements

Current compiled version tool requires a minimum version of Java 17

Building from Source

- Checkout source code from repository
- Run Maven clean & package commands
- Locate Xsd2JsonSchema-<N.n>-Tool.zip in 'target' folder
    - Semantic Versioning Scheme
        - N - Major version
        - n - Minor version
- Follow the instructions to run - from the Usage from executable

Usage from executable

1. Download Xsd2JsonSchema-<N.n>-Tool.zip
2. Unzip to your local drive. For example, C:\Xsd2JsonSchema_Tools
3. Run java -jar Xsd2JsonSchema.jar -h for usage from unzipped folder from above step

```
C:\Xsd2JsonSchema-1.0>java -jar Xsd2JsonSchema.jar -h
Usage: java -jar Xsd2JsonSchema.jar [options]
Options (any order):
 -i, --input <path> input XSD file or folder (required)
 -o, --output <folder> output folder for JSON Schema files (required)
 -v, --version <text> version text appended to descriptions (required)
 -r, --recursive emit dependencies too when input is a single XSD file (optional;
invalid with directory input)
 -a, --additional-terms <file> additional known terms file (optional)
 -p, --print-known-terms print known terms list and exit (exclusive)
 -l, --log-level <LEVEL> java.util.logging.Level (optional, default INFO)
 -s, --skip-choice-special-case <csv> comma-separated XSD basenames to skip
choice special-case (optional)
 -u, --unbounded-choice-repeat <csv> comma-separated XSD basenames to use ordered
                                     content-array projection for eligible
unbounded
                                     repeating choices (optional)
 -n, --no-schema-simplification disable conservative JSON Schema post-processing
cleanup
                                     (enabled by default; cleanup collapses
redundant branch
                                     properties/type, collapses single-entry
anyOf/allOf,
```

```
                                    and simplifies not{anyOf:[X]} to not:X)
(optional)
 -h, --help print this help
```

4. Print current list of known ST.96 ANNEX I - APPENDIX C - LIST OF ACRONYMS AND ABBREVIATIONS

```
C:\Xsd2JsonSchema_Tool_>java -jar Xsd2JsonSchema.jar -p
```

5. Transform one XSD with standard ST.96 Appedix C Terms into an output folder with JSON schema version

```
C:\Xsd2JsonSchema_Tool_>java -jar Xsd2JsonSchema.jar -i
"C:\ST96XMLSchema_V9_0\Common\AbstractNumber.xsd" -o "C:\JSON_Schemas" -v V2_0
```

6. Transform one XSD with user provided Abbreviations/Terms into an output folder with JSON schema version

```
C:\Xsd2JsonSchema_Tool_>java -jar Xsd2JsonSchema.jar -i
"C:\ST96XMLSchema_V9_0\Common\AbstractNumber.xsd" -o "C:\JSON_Schemas" -v V2_0 -a
C:\CustomAbbreviations.txt
```

7. Transform one XSD with standard ST.96 Appedix C Terms into an output folder with JSON schema version with all dependent child components

```
C:\Xsd2JsonSchema_Tool_>java -jar Xsd2JsonSchema.jar -i
"C:\ST96XMLSchema_V9_0\Common\AbstractNumber.xsd" -o "C:\JSON_Schemas" -v V2_0 -r
```

8. Transform one XSD with user provided Abbreviations/Terms into an output folder with JSON schema version with all dependent child components

```
C:\Xsd2JsonSchema_Tool_>java -jar Xsd2JsonSchema.jar -i
"C:\ST96XMLSchema_V9_0\Common\AbstractNumber.xsd" -o "C:\JSON_Schemas" -v V2_0 -r
-a C:\CustomAbbreviations.txt
```

9. Transform an entire ST.96 XSD schemas set into an output folder with JSON schema version

```
C:\Xsd2JsonSchema_Tool_>java -jar Xsd2JsonSchema.jar -i "C:\ST96XMLSchema_V9_0" -
o "C:\JSON_Schemas" -v V2_0
```

10. Transform an entire ST.96 XSD schemas set with user provided Abbreviations/Terms into an output folder with JSON schema version

```
C:\Xsd2JsonSchema_Tool_>java -jar Xsd2JsonSchema.jar -i "C:\ST96XMLSchema_V9_0" -
o "C:\JSON_Schemas" -v V2_0 -a C:\CustomAbbreviations.txt
```

11. Transform an entire ST.96 XSD schemas set with skip-choice-special-case into an output folder with JSON schema version

```
C:\Xsd2JsonSchema_Tool_>java -jar Xsd2JsonSchema.jar -i "C:\ST96XMLSchema_V9_0" -
o "C:\JSON_Schemas" -v V2_0 -s VersionType
```

12. Transform an entire ST.96 XSD schemas set with unbounded-choice-repeat into an output folder with JSON schema version

```
C:\Xsd2JsonSchema_Tool_>java -jar Xsd2JsonSchema.jar -i "C:\ST96XMLSchema_V9_0" -
o "C:\JSON_Schemas" -v V2_0 -u DescriptionType
```

Contact / support

Sunil Gera (sunildotgeraatusptodotgov)