

HRA – Visual Resources | Tutorial

User Manual for VRA Core 4 Transform Tool

(Version 1.0.0)



Contents

1.	Introduction and project history	3
2.	Data preparation	4
2.1.	Predefined headers	4
2.2.	Repeated values	4
2.3.	Display values	5
2.4.	Controlled data.....	6
3.	Introducing the template	7
3.1.	Downloading the template.....	7
3.2.	The template explained.....	7
3.3.	Creating a .csv file.....	9
4.	Accessing the transform tool online.....	10
4.1.	The basic user interface.....	10
5.	Uploading a file.....	11
6.	Generating VRA Core 4 XML.....	13
6.1.	Selecting a mapping	13
6.2.	Selecting a transformation	14
6.3.	Defining the records to be processed	14
6.4.	Generating XML.....	15
6.5.	Automatic validation	16
6.6.	Allowing pop-ups.....	17
7.	Advanced Mode	19
7.1.	Applying XSLs.....	19
7.2.	Validating the XML	21
8.	If validation fails	23
8.1.	Example 1 (element type):	23
8.2.	Example 2 (element type):	24
8.3.	Example 3 (date pattern):.....	24
8.4.	Example 4 (work/image ID):.....	25
8.5.	Example 5 (empty output):.....	26
9.	Generating RDF XML	27
9.1.	Simple mode.....	27
9.2.	Advanced mode.....	27
10.	Processing works with multiple image records.....	29
11.	Contact	30
12.	Appendix: Elements, displays and repetitions	31
13.	Appendix: Columns Full Template.....	33



1. Introduction and project history

The *VRA Core 4 XML Transform Tool* converts descriptive image metadata from flat tables (.csv) to structured VRA Core 4 XML. It makes use of a template with predefined headers. Users may work with the tool in a simple or advanced mode. The transform tool performs XML validation and provides feedback in case of errors.

This manual covers the whole workflow of using the *VRA Core 4 XML Transform Tool* including descriptions of all its features:

- preparation of data
- uploading data
- adjusting usage settings
- data transformation
- data validation
- handling validation feedback
- downloading the XML files

It also covers the (still experimental) transformation to RDF XML and includes a number of appendices for further reference.

Based on an initiative by Susan Jane Williams, she and Matthias Arnold successfully applied for a [Project Grant](#) at the [Visual Resources Association Foundation](#) (VRAF) in fall 2014 to develop a *VRA Core 4 XML Transform Tool*. We received additional support by the [Heidelberg Research Architecture](#) (HRA) at the [Heidelberg Centre for Transcultural Studies](#) (formerly Cluster of Excellence "Asia and Europe in a Global Context"), Heidelberg University.

Susan Jane Williams, Greg Reser, and Matthias Arnold developed the template csv files and the mappings to XML. The Heidelberg Research Architecture coordinated implementation and development of the tool, HRA's senior developer Matthias Guth conducted the programming, Tobi Krebs of [eXist-Solutions](#) provided additional code.

We released the tool as version 1.0 in February 2016. Although we put a lot of energy in testing, there may still be some hidden bugs or problems. If you find anything strange, or wish to provide us feedback on issues, bugs or problems encountered, or if you want to share your ideas on how the tool can be further improved, please do not hesitate to write us. We appreciate any feedback very much! Please direct it to Matthias Arnold at matthias.arnold@uni-hd.de (for more information see below, section "Contact").

The *VRA Core 4 XML Transform Tool* is developed as Open Source, its source code can be found at: <https://github.com/exc-asia-and-europe/csv2xml>

Note: We will update this user manual with every new feature or functionality. The latest version can always be found on GitHub: <https://github.com/exc-asia-and-europe/csv2xml/tree/master/doc>.

We also provide free access to an online version for instantaneous use at: <http://kjc-sv036.kjc.uni-heidelberg.de:8080/exist/apps/csv2xml/index.xq>.



2. Data preparation

The *VRA Core 4 XML Transform Tool* uses **predefined headers** to identify data in the template in order to map the values to the correct elements (or attributes) in XML. We arranged these headers in template files (for example in .xlsx or .ods format tables).

You can use any descriptive image metadata for transformation, be it an export from your institution's Digital Asset Management System, metadata embedded in images, or a personal image database. You only need to do two things for preparation:

1. Use the predefined template headers
2. Export data to comma separated value (.csv) file

2.1. Predefined headers

The template offers 225 columns with fixed headers for your data. To be able to transform your .csv data into XML you have to use these headers.

These include references to your local system, like "IMAGE_Filename" or "IMAGE_LocalRepoName". In the list, a number of elements are repeated to allow the transformation of multiple values, for example it offers up to three agents and up to eight subjects for the WORK record. The template also includes many columns for references to controlled vocabularies, like the name and Ref-ID of a subject in a vocabulary, and data types controlled by the VRA Core 4 schema.

For example, we express each of the eight subjects of a WORK record in a set of four columns:

```
WORK_Subject1  
WORK_SubjectType1  
WORK_SubjectVocab1  
WORK_SubjectRefid1
```

The complete list of headers can be found in "*Appendix: Columns Full Template*" below.

IMPORTANT:

You do not need to fill each column with data. The sequence of columns (headers) may be changed and you can even delete columns you will not need.

It is, however, essential **not to change the headers** themselves, i.e. not their "text" or "values".

2.2. Repeated values

For some of the eighteen elementSet's of VRA Core 4 XML for WORK and IMAGE records elements can be repeated. These values usually are combined with other information, like references to authority files.

WORK

Agent:	3
CulturalContext:	2
Date:	2
Location:	2
Material:	4
Measurements:	8 (in 2 sets)
Relation:	2 (plus 1 work-image)
Style/Period:	4
Subject:	8
Technique:	4
Title:	2 (preferred/alternative)
Worktype:	3

IMAGE

Subject:	3
----------	---

An expanded list of repeated elements can be found below, in "*Appendix: Elements, displays and repetitions.*"

2.3. Display values

The template is designed to include data for each of the eighteen elementSet's of VRA Core 4 XML for WORK and most of the IMAGE records. For a large number of elementsSets structured (or "qualified") information can be transformed.

All elementSets may have a display element, with the exception of the following four IMAGE elementSets: CulturalContext, StateEdition, StylePeriod, and Textref. These are NOT included in the template.

For an overview of the elementSets, including information about which only contains display values and which also includes other elements and attributes please refer to "*Appendix: Elements, displays and repetitions*" below.

2.3.1. Auto-filling display values

The *VRA Core 4 XML Transform Tool* allows transforming both, structured data for sub-elements and attributes, as well as unstructured data for display values.

If users do NOT have <display> data in their .csv file, *VRA Core 4 XML Transform Tool* will automatically fill the content of the <display> element based on available data from the structured information.

For example, if the template contains

WORK_AgentDisplay	[empty]
WORK_Agent1Name	Michelangelo Buonarroti
WORK_Agent1Role	painter
WORK_Agent1Attribution	school of

The following the display element is created:

```
<display>school of Michelangelo Buonarroti (painter)</display>
```



However, if users do provide data for the display element (in the example "WORK_AgentDisplay") these values will be used and NOT overwritten or changed.

2.4. Controlled data

[VRA Core 4.0](#) in the restricted version uses controlled type lists and date formats.

2.4.1. Data type values

For the **type values** please refer to the [VRA Core 4.0 Restricted Schema Type Values](#) document.

2.4.2. Date values

For **date values** there is no individual documentation file. [Vra-strict.xsd](#) states for dateValueType:

"...Defines a date which follows the ISO 8601 date format, and allows right truncation. [...] In brief, the following formats are allowed:

present	
2006	(2006)
2006-12	(December, 2006)
2006-12-31	(31 December 2006)
-44	(44 BCE)
-44-03	(March, 44 BCE)
-44-03-15	(15 March, 44 BCE)
-100000000	(10 Million Years Ago)
-1000000000000	(100 Billion Years Ago)"

Source: <http://loc.gov/standards/vracore/vra-strict.xsd>, last accessed December 12, 2015

2.4.3. Controlled data in the Transform tool

IMPORTANT:

To be able to convert your data into valid VRA Core 4 XML these type values must be used and date format rules must be followed.

The transform tool will assist you as far as possible.

For example, in <date> the sub-element <latestDate> must not be empty. But if you only entered <earliestDate> the tool will automatically fill <latestDate> with the <earliestDate> value.

If you accidentally mistyped a controlled type value, the validation will point you to this error in the validation results (see chapter "Validation errors" below).



3. Introducing the template

3.1. Downloading the template

The **full template** for *VRA Core 4 XML Transform Tool* is available for download at

<https://github.com/exc-asia-and-europe/csv2xml/tree/master/doc>

in these formats:

- Comma-separated values text file (.csv)
- Excel workbook (.xlsx)
- OpenDocument spreadsheet (.ods)

3.2. The template explained

The template consists of three worksheets.

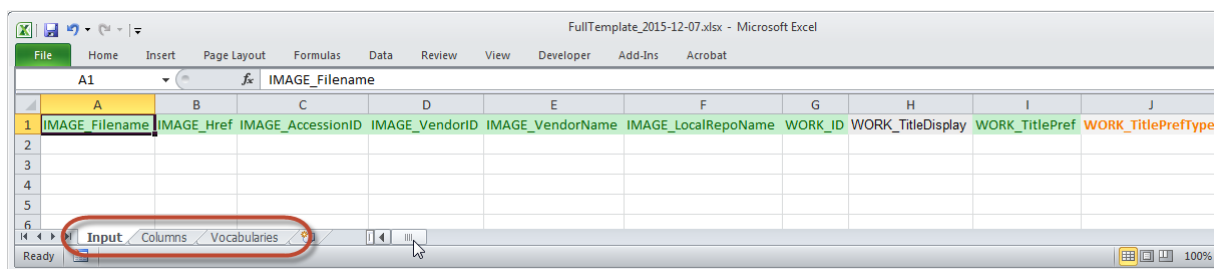


Figure 1: Full template - Input worksheet (in Excel)

The **"Input"** worksheet contains all the columns with their respective header in horizontal layout (colour coded), so that it can be used for editing data.

The **"Columns"** worksheet contains two columns. In the first, all headers are arranged vertically and colour coded, while in the second some hints to the content of the data are given.

The colour codes are included to provide visual feedback in case different types of data are expected for a "field".

Display values, for which the tool will auto-fill content if no data was provided for display are shown in black on grey background.

WORK_TitleDisplay

Type values are shown in orange on light grey. Allowed values are provided in the second column.

IMAGE_RightsType copyrighted, publicDomain, undetermined, other

References to external vocabularies are shown in dark blue on orange background.

WORK_Agent1NameVocab



Special content relevant for validation is shown in black on red background, together with the validation rules. This is used for dates.

WORK_EarliestDate1	VALIDATION <xsd:pattern value="present (-)*[0-9]{1,12}(-[0-9]{2}(-[0-9]{2})*)**"/>
WORK_LatestDate1	VALIDATION <xsd:pattern value="present (-)*[0-9]{1,12}(-[0-9]{2}(-[0-9]{2})*)**"/>

The “normal” columns are shown in dark green on lighter green.

A	B	C	D	E
IMAGE_VendorName	for IMAGE_source			
IMAGE_LocalRepoName	for IMAGE_location			
WORK_ID	for WORK_work			
WORK_TitleDisplay				
WORK_TitlePref				
WORK_TitlePrefType	brandName, cited, creator, descriptive, former, inscribed, owner, popular, repository, translated, other			
WORK_TitleAlt				
WORK_TitleAltType	brandName, cited, creator, descriptive, former, inscribed, owner, popular, repository, translated, other			
WORK_AgentDisplay				
WORK_Agent1NameType	personal, corporate, family, other			
WORK_Agent1Name				
WORK_Agent1NameVocab	Vocabularies!A1			
WORK_Agent1NameRefid				
WORK_Agent1Role				
WORK_Agent1Attribution				
WORK_Agent2NameType	personal, corporate, family, other			
WORK_Agent2Name				
WORK_Agent2NameVocab	Vocabularies!A1			
WORK_Agent2NameRefid				
WORK_Agent2Role				
WORK_Agent2Attribution				
WORK_Agent3NameType	personal, corporate, family, other			
WORK_Agent3Name				
WORK_Agent3NameVocab	Vocabularies!A1			
WORK_Agent3NameRefid				
WORK_Agent3Role				
WORK_Agent3Attribution				
WORK_CulturalContextDisplay				
WORK_CulturalContext1				
WORK_CulturalContext1Vocab	Vocabularies!A1			
WORK_CulturalContext1Refid				
WORK_CulturalContext2				
WORK_CulturalContext2Vocab	Vocabularies!A1			
WORK_CulturalContext2Refid				
WORK_DateDisplay				
WORK_Date1Type	alteration, broadcast, bulk, commission, creation, design, destruction, discovery, exhibition, inclusive, performance, publicat			
WORK_EarliestDate1	VALIDATION <xsd:pattern value="present (-)*[0-9]{1,12}(-[0-9]{2}(-[0-9]{2})*)**"/>			
WORK_LatestDate1	VALIDATION <xsd:pattern value="present (-)*[0-9]{1,12}(-[0-9]{2}(-[0-9]{2})*)**"/>			
WORK_Date2Type	alteration, broadcast, bulk, commission, creation, design, destruction, discovery, exhibition, inclusive, performance, publicat			
WORK_EarliestDate2	VALIDATION <xsd:pattern value="present (-)*[0-9]{1,12}(-[0-9]{2}(-[0-9]{2})*)**"/>			
WORK_LatestDate2	VALIDATION <xsd:pattern value="present (-)*[0-9]{1,12}(-[0-9]{2}(-[0-9]{2})*)**"/>			
WORK_DescriptionDisplay				
WORK_DescriptionSource				

Figure 2: Full template - Columns workbook (in Excel)

The “**Vocabularies**” worksheet contains a reference list of external authorities or controlled vocabularies. It is by no means comprehensive. The values of the “vocab” attribute are not controlled by the VRA Core 4 schema, but it is strongly recommended to make sure the content is consistent within your data set. This is especially of interest if you intend to transform your data to RDF and use it as Linked (Open) Data.



	A	B	C	D	E	F
1	AAT	http://vocab.getty.edu/aat/				
2	TGN	http://vocab.getty.edu/tgn/				
3	ULAN	http://vocab.getty.edu/ulan/				
4	CONA	http://vocab.getty.edu/cona/				
5	LCNAF	http://id.loc.gov/authorities/names/				
6	LCSH	http://id.loc.gov/authorities/subjects/				
7	LCGFT	http://id.loc.gov/authorities/genreForms/				
8	TGM	http://id.loc.gov/vocabulary/graphicMaterials/				
9	VIAF	http://viaf.org/viaf/				
10	ICONCLASS	http://iconclass.org/				
11						
12						

Figure 3: Full template - Vocabularies workbook (in Excel)

3.3. Creating a .csv file

Export records from your local system directly into .csv files if you can include the template headers.

Alternatively, use your preferred spreadsheet software (e.g. Microsoft Excel, OpenOffice Calc, or LibreOffice Calc) as intermediary to adjust the headers and export from there into .csv files. In the [doc folder on GitHub](#) an Excel (.xlsx) version of the template is available.

IMPORTANT:

Only .csv files can be transformed so make sure you convert spreadsheets (e.g. from Excel or Calc) to .csv text files before transformation.

Use UTF-8 as encoding to conserve possible special characters in your data.

For a guide on how to convert Microsoft Excel files into comma separated value (.csv) files see the file "*Converting .xlsx file to .csv using OpenOffice Calc*" available in the documentation folder on GitHub.

[To be covered:

Other data sources: EMWG export-import tool]

4. Accessing the transform tool online

For the time being, the *VRA Core 4 XML Transform Tool* can be accessed online for testing at <http://kjc-sv036.kjc.uni-heidelberg.de:8080/exist/apps/csv2xml/index.xq>.

Please note, that this is a server dedicated to testing software and developments.

4.1. The basic user interface

The *VRA Core 4 XML Transform Tool* interface offers a Simple and an Advanced Mode.

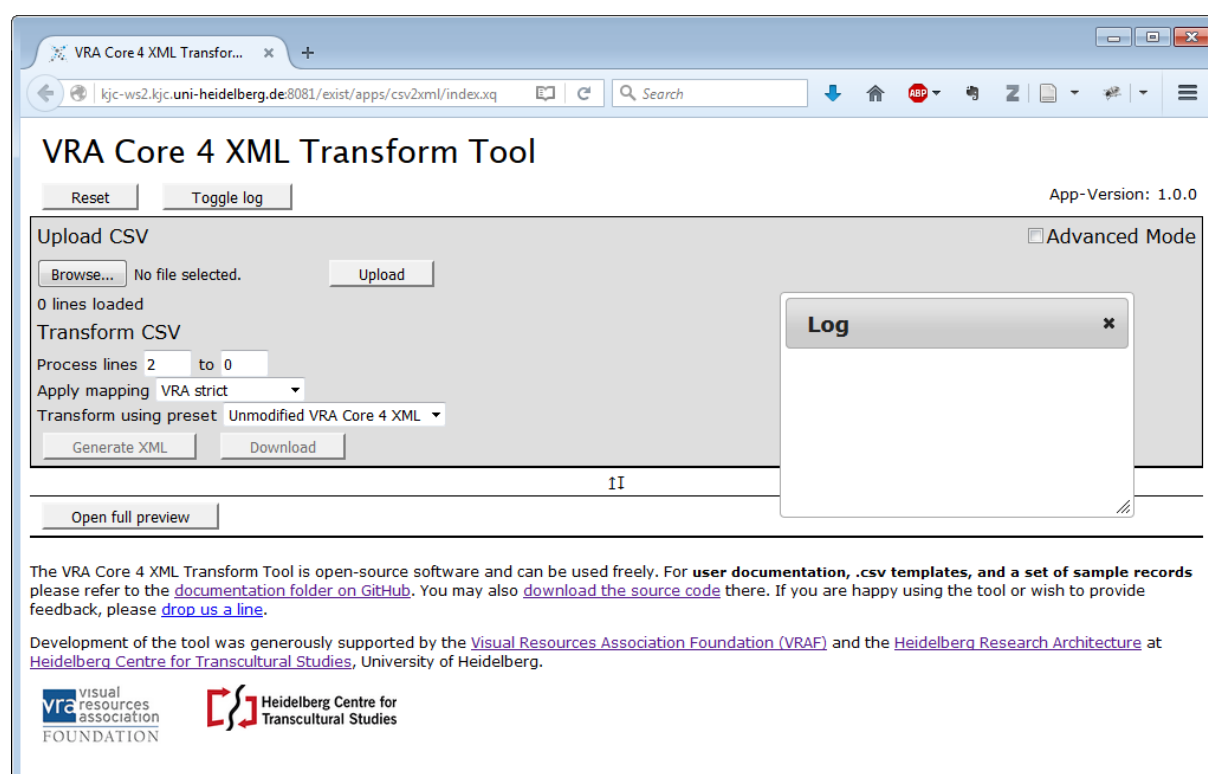


Figure 4: User interface - Simple mode.

In simple mode the XML generation and validation are processed in one step finishing with the optional download of the XML files.

The advanced interface provides additional functionalities for the advanced user including the possibility to apply additional transformations.

Both interfaces offer a "Log" window which is visible by default. The tool displays status messages here, for example which record is currently processed. It can be switched off/on using the "toggle lock" button.



5. Uploading a file

To transform data you need to upload it first. Click the "Browse" button:

Upload CSV

Browse... No file selected. Upload

0 lines loaded

Transform CSV

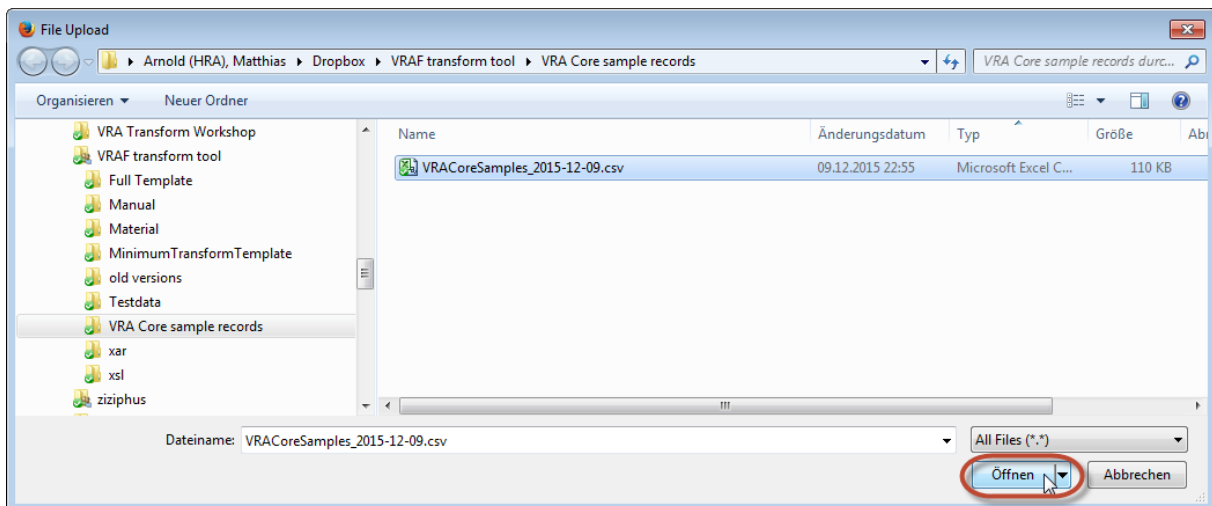
Process lines 2 to 0

Apply mapping VRA strict

Transform using preset Unmodified VRA Core 4 XML

Generate XML Download

Select the .csv file on your computer and click "Open".



The name of the .csv file will be displayed in the interface. In this example we use "VRACoreSamples_2015-12-09.csv" which contains data from the cataloguing examples on the VRA Core support pages¹.

Now click on "Upload".

Upload CSV

Browse... VRACoreSamples.csv Upload

0 lines loaded

Transform CSV

Process lines 2 to 0

Apply mapping VRA strict

Transform using preset Unmodified VRA Core 4 XML

Generate XML Download

¹ http://core.vraweb.org/vracore_examples.html



Once the upload is finished you will see an message "CSV upload successful" in the log window and the tool shows the number of lines in the .csv file that were loaded.

VRA Core 4 XML Transform Tool

ResetToggle logApp-Version: 1.0.0

Upload CSV

Browse... VRACoreSamples.csv Upload

44 lines loaded

Transform CSV

Process lines 2 to 44

Apply mapping VRA strict

Transform using preset Unmodified VRA Core 4 XML

Generate XML Download

Log

10:08:40: CSV upload successful

10:08:40: Upload successful.

10:08:32: Session resettled.

Open full preview

The VRA Core 4 XML Transform Tool is open-source software and can be used freely. For **user docu** please refer to the [documentation folder on GitHub](#). You may also [download the source code](#) there. feedback, please [drop us a line](#).

Development of the tool was generously supported by the [Visual Resources Association Foundation](#) [Heidelberg Centre for Transcultural Studies](#), University of Heidelberg.

visual resources association FOUNDATION

Heidelberg Centre for Transcultural Studies

6. Generating VRA Core 4 XML

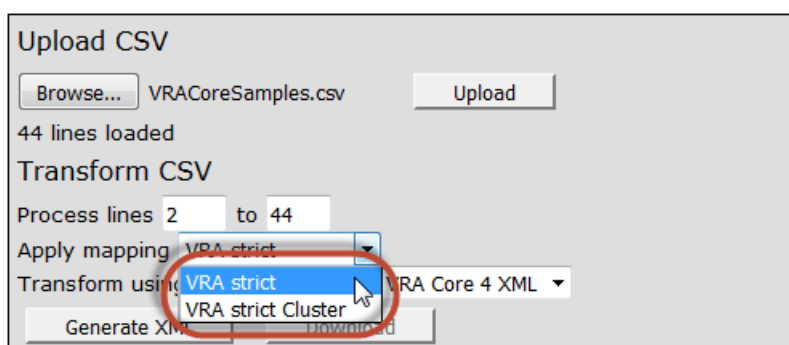
After uploading a .csv file to the *VRA Core 4 XML Transform Tool* you can just hit the "generate XML" button. This will process all records into VRA Core 4 XML using the default settings (transform all records to VRA strict without any modifications).

However, even in Simple Mode you can manipulate how the XML is generated. You have three options:

- Select another mapping
- Select another transform
- Define the range of records to be processed

6.1. Selecting a mapping

The *VRA Core 4 XML Transform Tool* supports generating XML based on different mappings. For VRA Core, two variants are available: "VRA strict" and "VRA strict Cluster".



6.1.1. VRA strict

VRA strict is the **default mapping**. It is based on the VRA Core 4 restricted version (<http://www.loc.gov/standards/vracore/vra-strict.xsd>). It uses controlled values for type attributes and specifies date formats.

For more information, please refer to <http://www.loc.gov/standards/vracore/schemas.html>.

6.1.2. VRA strict Cluster

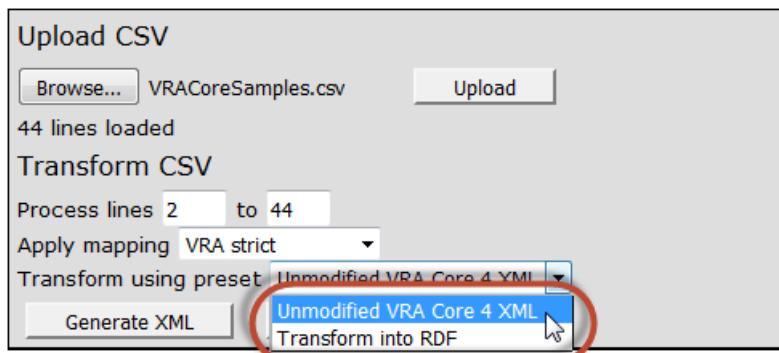
The VRA strict Cluster mapping is an expansion of the restricted VRA Core 4 version developed at the Cluster of Excellence "Asia and Europe in a Global Context", University of Heidelberg. It contains additional elements and attributes, for example, attributes for multilingual data, role attributes for agents, and an element for geo-coordinates (<http://cluster-schemas.uni-hd.de/vra-strictCluster.xsd>).

For more information, please refer to the document "[VRA Core – Extensions](#)".

6.2. Selecting a transformation

The tool supports different transformations.

The current version of *VRA Core 4 XML Transform Tool* offers two built-in transformations. Additional .xsl files may be added in the source code. In a future version, this may be changed (e.g. upload transformation stylesheets in the user interface).



Currently, two transformations are implemented:

- Unmodified VRA Core 4 XML
- Transform into RDF (explanations see below)

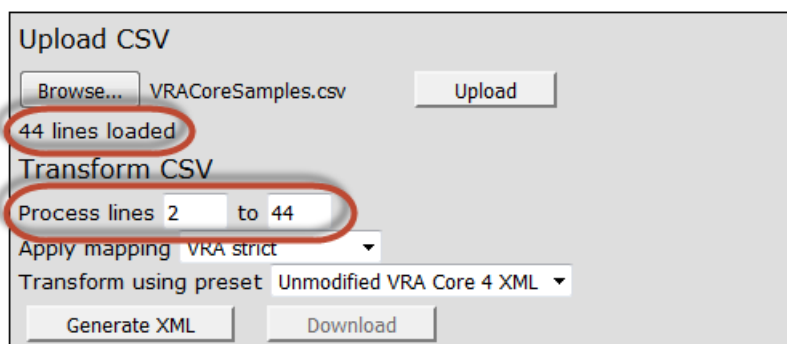
To transform the .csv data into VRA Core 4 XML choose "Unmodified VRA Core 4 XML".

6.3. Defining the records to be processed

By default, all records listed in the uploaded .csv file will be processed.

6.3.1. Calculating records for processing

Within the .csv file a record equals a line. In the template, the first record (first line) contains the column headers. Therefore, the *VRA Core 4 XML Transform Tool* will by default process data beginning with line #2 (i.e. record #1).





6.3.2. Defining a range of records

The *VRA Core 4 XML Transform Tool* also allows you to define the range of records to be processed. This is very helpful if you want to check a smaller range of records or even individual ones, look at their xml and test if they validate.

Upload CSV

Browse... VRACoreSamples.csv Upload

44 lines loaded

Transform CSV

Process lines 5 to 5

Apply mapping VRA strict

Transform using preset Unmodified VRA Core 4 XML

Generate XML Download

In this example, only line 5 (i.e. record #4) will be processed.

6.4. Generating XML

If all settings are made, click the “generate XML” button.

The tool will start generating the XML. You can monitor the progress in the “Log” window.

Upload CSV

Browse... VRACoreSamples.csv Upload

44 lines loaded

Transform CSV

Process lines 2 to 44

Apply mapping VRA strict

Transform using preset Unmodified VRA Core 4 XML

Generate XML Download

Log

11:02:49: line 8 processed.
11:02:48: line 7 processed.
11:02:48: line 6 processed.
11:02:47: line 5 processed.
11:02:46: line 4 processed.
11:02:46: line 3 processed.
11:02:45: line 2 processed.
11:02:44: Start processing 43 lines
11:02:44: -> 23d13f75-9dae-45f2-b136-872884ba6590.xml
11:02:44: Generating parent document...
11:02:44: template loaded: vra-image-template.xml
11:02:44: template loaded: vra-work-template.xml
11:02:44: loadingTemplates
11:02:44: generating...
11:01:50: CSV upload successful
11:01:50: Upload successful.
11:01:43: Session resettled.



6.5. Automatic validation

In Simple mode the tool will generate XML and automatically run a validation. When all is finished the "Generate XML" button changes its colour:



If your XML did not validate you cannot process it further. The validation will generate an error report. For instructions on how to interpret validation results, please refer to chapter "If validation fails", further down in this manual.

The screenshot shows the VRA Core 4 XML Transform Tool interface. The 'Generate XML' button is highlighted in green. A 'Log' window is open on the right, showing a list of processing steps. The main area displays XML code for a 'work' element, including agent, cultural context, date, and location information.

Log

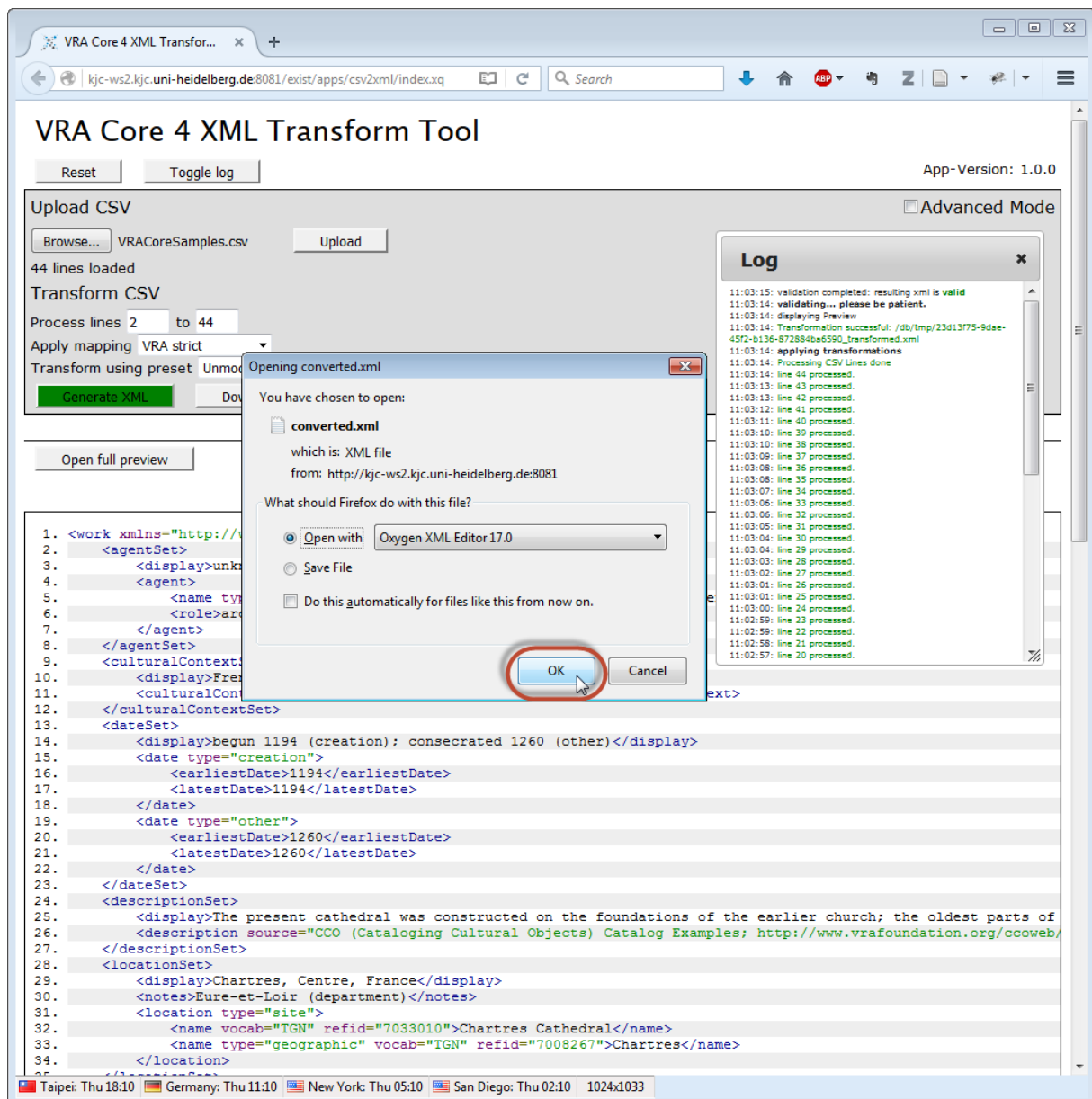
- 11:03:15: validation completed: resulting xml is valid
- 11:03:14: validating... please be patient.
- 11:03:14: displaying Preview
- 11:03:14: Transformation successful: /db/tmp/23d13775-9dae-45f2-b136-872884be6590_transformed.xml
- 11:03:14: applying transformations
- 11:03:14: Processing CSV Lines done
- 11:03:14: line 44 processed.
- 11:03:13: line 43 processed.
- 11:03:13: line 42 processed.
- 11:03:12: line 41 processed.
- 11:03:11: line 40 processed.
- 11:03:10: line 39 processed.
- 11:03:10: line 38 processed.
- 11:03:09: line 37 processed.
- 11:03:08: line 36 processed.
- 11:03:08: line 35 processed.
- 11:03:07: line 34 processed.
- 11:03:06: line 33 processed.
- 11:03:06: line 32 processed.
- 11:03:05: line 31 processed.
- 11:03:04: line 30 processed.
- 11:03:04: line 29 processed.
- 11:03:03: line 28 processed.
- 11:03:02: line 27 processed.
- 11:03:01: line 26 processed.
- 11:03:01: line 25 processed.
- 11:03:00: line 24 processed.
- 11:02:59: line 23 processed.
- 11:02:59: line 22 processed.
- 11:02:58: line 21 processed.
- 11:02:57: line 20 processed.

```
1. <work xmlns="http://www.vraweb.org/vracore4.htm" id="w_1">
2.   <agentSet>
3.     <display>unknown (French architect)</display>
4.     <agent>
5.       <name type="personal" vocab="ULAN" refid="500125235">unknown Fre
6.       <role>architect</role>
7.     </agent>
8.   </agentSet>
9.   <culturalContextSet>
10.    <display>French</display>
11.    <culturalContext vocab="AAT" refid="300111188">French</culturalContext>
12.  </culturalContextSet>
13.  <dateSet>
14.    <display>begun 1194 (creation); consecrated 1260 (other)</display>
15.    <date type="creation">
16.      <earliestDate>1194</earliestDate>
17.      <latestDate>1194</latestDate>
18.    </date>
19.    <date type="other">
20.      <earliestDate>1260</earliestDate>
21.      <latestDate>1260</latestDate>
22.    </date>
23.  </dateSet>
24.  <descriptionSet>
25.    <display>The present cathedral was constructed on the foundations of the earlier church; the oldest parts of
26.    <description source="CCO (Cataloging Cultural Objects) Catalog Examples; http://www.vrafoundation.org/ccoweb/
27.  </descriptionSet>
28.  <locationSet>
29.    <display>Chartres, Centre, France</display>
30.    <notes>Eure-et-Loir (department)</notes>
31.    <location type="site">
32.      <name vocab="IGN" refid="7039010">Chartres Cathedral</name>
33.      <name type="geographic" vocab="IGN" refid="7008267">Chartres</name>
34.    </location>
35.  </locationSet>
```

You can browse the preview by clicking the arrow button. The preview shows generated WORK and IMAGE code in XML on separate pages.



You can also directly use the "download" link which will show the download dialog "Opening converted.xml". Note that the default file name is "converted.xml".



You can save the file on a disk or open it in your preferred XML editor.

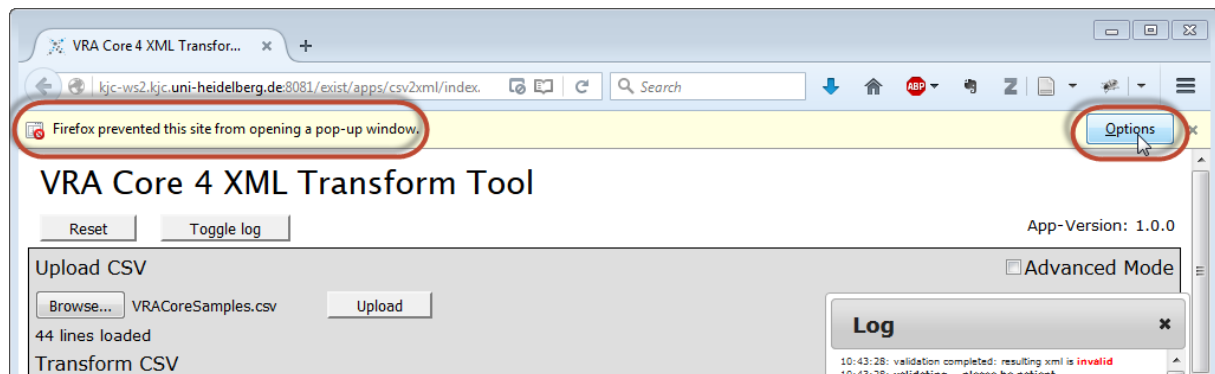
6.6. Allowing pop-ups

Instead of downloading the result immediately, you can also click on the "open full preview" button to view the whole XML in your browser. This will open a new tab.

Depending on your local browser settings, opening the new tab may trigger a pop-up blocking message.



In Firefox it may look like this:



Choose "Allow pop-ups" from the "Options" menu.



7. Advanced Mode

In the Advanced mode it is possible to

- apply XSL transforms
- choose a schema against which you wish to validate your XML file
- upload your own schema file

It also separates XML generation from validation.

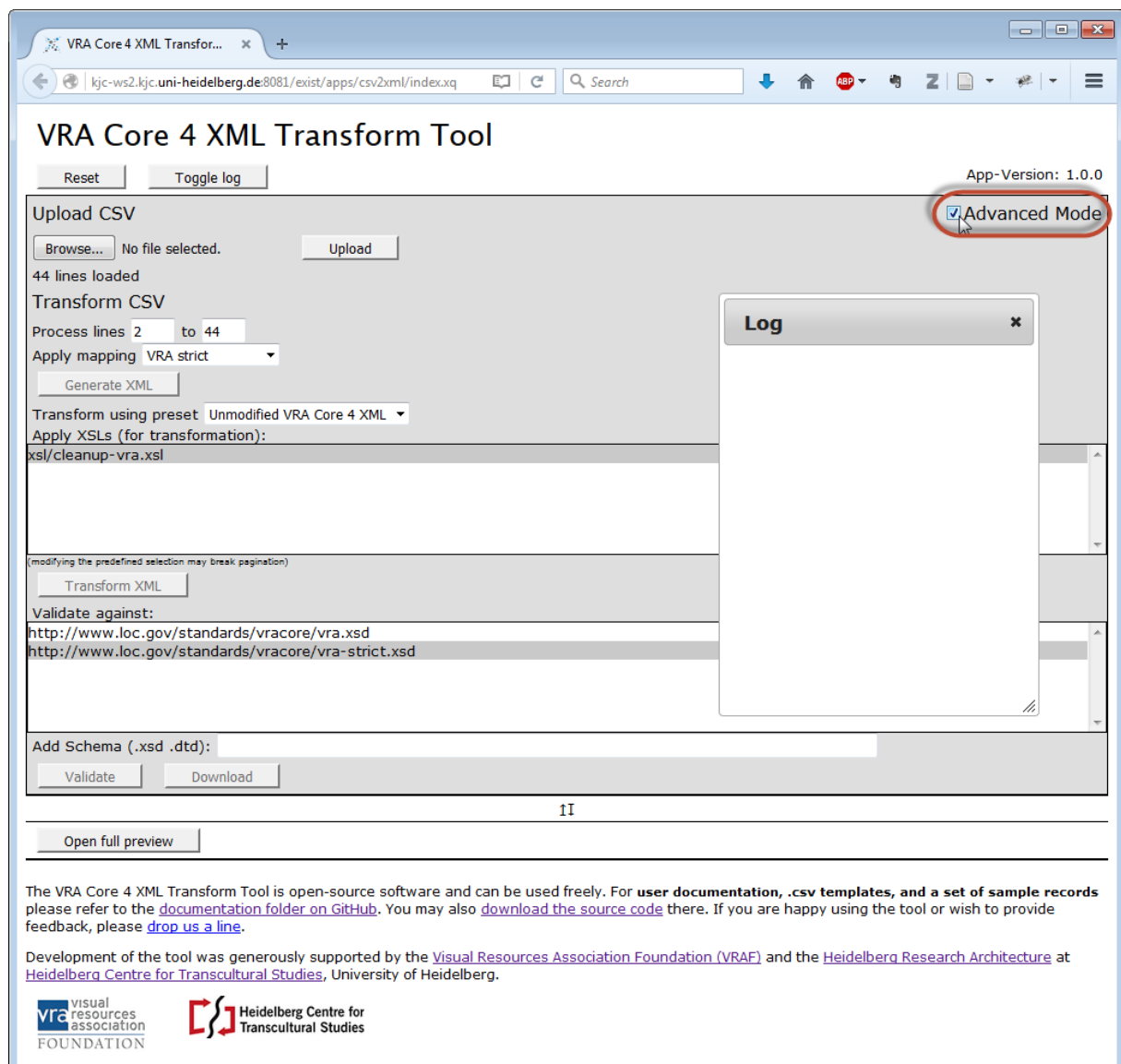


Figure 5: User interface - Advanced mode.

7.1. Applying XSLs

During the default generation of "Unmodified VRA Core 4 XML" a transformation script "cleanup-vra.xsl" will be automatically applied. It removes empty attributes and elements and keeps the resulting XML file slim and clean. It is possible to deselect it here. You should note, however, that without running "cleanup-vra.xsl" the resulting XML code may not validate.



If you choose "Transform into RDF" as transform pre-set the "VRA2RDF.xml" will be loaded and applied by default. For more information about RDF transformation, see chapter "Generating RDF" below.

It is possible to add other .xml files in the source code. In a future version of *VRA Core 4 XML Transform Tool* this may be changed.

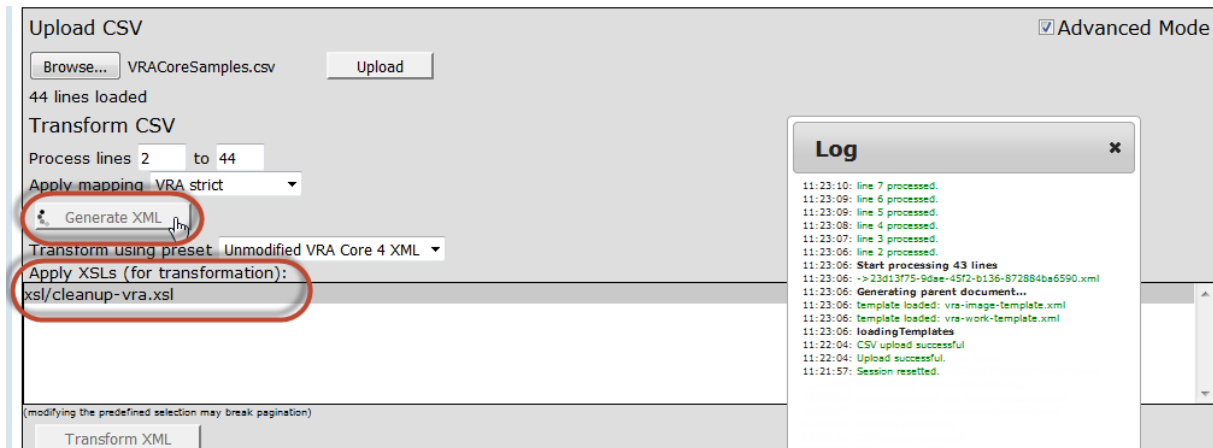


Figure 6: Advanced Mode - XML generation without additional transformation



Figure 7: Preview of XML without clean-up

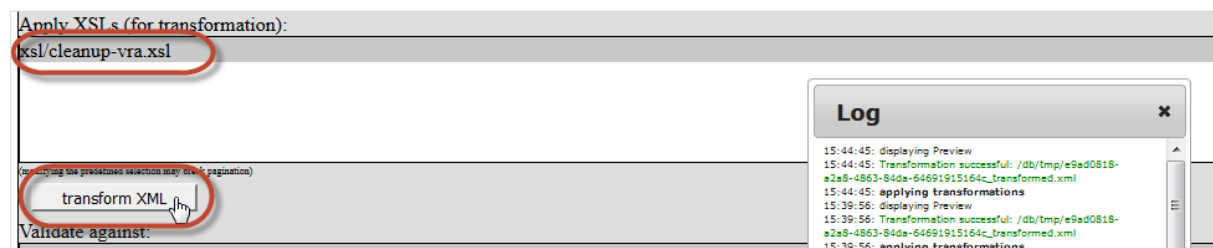


Figure 8: Starting the clean-up manually



```
1 / 80 ->
1. <work xmlns="http://www.vraweb.org/vracore4.htm" id="w_1">
2.   <agentSet>
3.     <display>unknown (French architect)</display>
4.     <agent>
5.       <name type="personal" vocab="ULAN" refid="500125235">unknown French</name>
6.       <role>architect</role>
7.     </agent>
8.   </agentSet>
9.   <culturalContextSet>
10.    <display>French</display>
11.    <culturalContext vocab="AAT" refid="300111188">French</culturalContext>
12.  </culturalContextSet>
13.  <dateSet>
14.    <display>begun 1194 (creation); consecrated 1260 (other)</display>
15.    <date type="creation">
16.      <earliestDate>1194</earliestDate>
17.      <latestDate>1194</latestDate>
18.    </date>
19.    <date type="other">
20.      <earliestDate>1260</earliestDate>
21.      <latestDate>1260</latestDate>
22.    </date>
23.  </dateSet>
24.  <descriptionSet>
25.    <display>The present cathedral was constructed on the foundations of the earlier church; the oldest parts of the cathedral s
26.    <description source="CCO (Cataloging Cultural Objects) Catalog Examples; http://www.vrafoundation.org/ccoweb/cco/examplesind
27.  </descriptionSet>
```

Figure 9: Preview of XML after clean-up

7.2. Validating the XML

Transforming data from .csv to XML does not *per se* guarantee a valid XML result. This is because the restricted VRA Core schema defines a number of fixed values. If the values in the .csv document do not match the values expected by the schema, validation will fail. We therefore strongly recommend validating the generated data in advanced mode.

7.2.1. Choosing a schema for validation

In the "Validate against" box you can choose a schema for validation. Depending on the selected mapping (see above) the respective default validation schema is automatically set.

Validate against:

<http://www.loc.gov/standards/vracore/vra-strict.xsd>

Add Schema (.xsd .dtd):

For example, if you selected "VRA strict" as mapping, the corresponding schema <http://www.loc.gov/standards/vracore/vra-strict.xsd> will be set.

It is possible to choose different validation schemas. In a future version of the transform tool users may also upload their own schema.

7.2.2. Starting validation

To start the validation process, click "validate".



Transform XML

Validate against:

<http://www.loc.gov/standards/vracore/vra.xsd>

<http://www.loc.gov/standards/vracore/vra-strict.xsd>

Add Schema (.xsd .dtd):

Validate Download

Open full preview

Log

11:23:36: displaying Preview

11:23:36: Transformation successful: /db/tmp/23d13f75-9d5e-45f2-b136-872884ba6590_transformed.xml

11:23:35: applying transformations

11:23:35: Processing CSV Lines done

11:23:35: line 44 processed.

11:23:35: line 43 processed.

11:23:34: line 42 processed.

11:23:33: line 41 processed.

11:23:33: line 40 processed.

11:23:32: line 39 processed.

11:23:31: line 38 processed.

11:23:31: line 37 processed.

11:23:30: line 36 processed.

11:23:29: line 35 processed.

11:23:29: line 34 processed.

11:23:28: line 33 processed.

11:23:27: line 32 processed.

11:23:27: line 31 processed.

7.2.3. The validation result

Once the validation is finished, a new browser tab will be opened with the validation result. In addition, the button's colour will change:



Result NOT VALID



Result VALID

Transform XML

Validate against:

<http://www.loc.gov/standards/vracore/vra.xsd>

<http://www.loc.gov/standards/vracore/vra-strict.xsd>

Add Schema (.xsd .dtd):

Validate Download

Open full preview

Log

11:32:10: validation completed: resulting xml is valid

11:32:09: validating... please be patient.

11:23:36: displaying Preview

11:23:36: Transformation successful: /db/tmp/23d13f75-9d5e-45f2-b136-872884ba6590_transformed.xml

11:23:35: applying transformations

11:23:35: Processing CSV Lines done

11:23:35: line 44 processed.

11:23:35: line 43 processed.

11:23:34: line 42 processed.

11:23:33: line 41 processed.

11:23:33: line 40 processed.

11:23:32: line 39 processed.

11:23:31: line 38 processed.

11:23:31: line 37 processed.

11:23:30: line 36 processed.

11:23:29: line 35 processed.

11:23:29: line 34 processed.

11:23:28: line 33 processed.

11:23:27: line 32 processed.

11:23:27: line 31 processed.

11:23:26: line 30 processed.

11:23:25: line 29 processed.

11:23:25: line 28 processed.

11:23:24: line 27 processed.

11:23:23: line 26 processed.

11:23:22: line 25 processed.

11:23:22: line 24 processed.

11:23:21: line 23 processed.

11:23:20: line 22 processed.

1 / 80 ->

```
1. <work xmlns="http://www.vraweb.org/vracore4.htm" id="w_1">
2.   <agentSet>
3.     <display>unknown (French architect)</display>
4.     <agent>
5.       <name type="personal" vocab="ULAN" refid="500125235">unknown
6.       <role>architect</role>
7.     </agent>
8.   </agentSet>
9.   <agentSet>
10.    <display>French</display>
11.    <culturalContext vocab="AAT" refid="300111188">French</culturalContext>
12.  </culturalContextSet>
13.  <dateSet>
14.    <display>begun 1194 (creation); consecrated 1260 (other)</display>
15.    <date type="creation">
16.      <earliestDate>1194</earliestDate>
17.      <latestDate>1194</latestDate>
18.    </date>
19.    <date type="other">
20.      <earliestDate>1260</earliestDate>
21.      <latestDate>1260</latestDate>
22.    </date>
23.  </dateSet>
24.  <descriptionSet>
25.    <display>The present cathedral was constructed on the foundations of the earlier church; the oldest parts of
26.    <description source="CCO (Cataloging Cultural Objects) Catalog Examples; http://www.vrafoundation.org/ccoweb/
```

Figure 10: The tool generated valid VRA Core 4 XML.



8. If validation fails

VRA Core 4 in the restricted version uses controlled type lists and date formats. If your data does not match the prescribed rules, your XML will not validate.

In case of problems with the validation, the error messages will be displayed in a separate tab. The messages will indicate both line and column of the respective generated XML document. This is displayed in the interface. Line numbers refer to the downloaded XML document, within the browser rendering of the XML (full preview) line numbers may differ.

Error messages usually come in pairs. The first line will name the invalid value based on an enumeration in the schema (e.g. type values defined by vra-strict). The second line will indicate the parent element.

```
<report-result>
  <xsd:http://kjc-ar016.kjc.uni-heidelberg.de:8080/exist/apps/camboti/resources/schemas/vra-strictCluster.xsd/xsd>
  <report>
    <status>invalid</status>
    <namespace>http://www.vraweb.org/vracore4.htm</namespace>
    <duration>0:00:00.000</duration>
    <message level="Error" line="30" column="36">cvc-pattern-valid: Value 'early1' is not facet-valid with respect to pattern 'present{[-]{0-9}(1,12){-}{0-9}(2){-}{0-9}(2){-}}*' for type 'dateValueType'.</message>
    <message level="Error" line="30" column="36">cvc-complex-type.2.2: Element 'date' must have no element [children], and the value must be valid.</message>
  </report>
</report-result>
```

8.1. Example 1 (element type):

Error message (screenshot):

```
<message level="Error" line="134" column="52">cvc-enumeration-valid: Value 'illustrative' is not facet-valid with respect to enumeration '[brandName, cited, creator, descriptive, former, generalView, inscribed, owner, partialView, popular, repository, translated, other]'. It must be a value from the enumeration.</message>
<message level="Error" line="134" column="52">cvc-attribute.3: The value 'illustrative' of attribute 'type' on element 'title' is not valid with respect to its type, 'titleTypeType'.</message>
<message level="Error" line="134" column="52">cvc-enumeration-valid: Value 'person' is not facet-valid with respect to enumeration '[person, corporate, family, other]'. It must be a value from the enumeration.</message>
```

Error message (transcribed):

```
<message level="Error" line="134" column="52">cvc-enumeration-valid: Value 'illustrative' is not facet-valid with respect to enumeration '[brandName, cited, creator, descriptive, former, generalView, inscribed, owner, partialView, popular, repository, translated, other]'. It must be a value from the enumeration.</message>
```

```
<message level="Error" line="134" column="52">cvc-attribute.3: The value 'illustrative' of attribute 'type' on element 'title' is not valid with respect to its type, 'titleTypeType'.</message>
```

The error message refers to `line="134" column="52"` in the XML:

```
131.         </textrefSet>
132.         <titleSet>
133.           <display>a predefined work title display</display>
134.           <title pref="true" type="illustrative">some_w_prefTitle</title>
135.           <title pref="false" type="translated">some_altTitle</title>
136.         </titleSet>
137.         <worktypeSet>
```

The first line of the message states that “illustrative” is not valid because it is not included in the list of possible values: “brandName, cited, creator, descriptive, former, generalView, inscribed, owner, partialView, popular, repository, translated, other”.

The second line adds that the error occurred within attribute “type” of element <title>, and the provided value for “type” was not valid.



Result:

Values of `<title type="">` are controlled by the schema.

You need to delete “illustrative” and use of one of the allowed values, i.e. “brandName, cited, creator, descriptive, former, generalView, inscribed, owner, partialView, popular, repository, translated, other”.

8.2. Example 2 (element type):

Error message (screenshot):



Error message (transcribed):

```
<message level="Error" line="247" column="67">cvc-enumeration-valid: Value 'person' is not facet-valid with respect to enumeration '[personal, corporate, family, other]'. It must be a value from the enumeration.</message>
```

```
<message level="Error" line="247" column="67">cvc-attribute.3: The value 'person' of attribute 'type' on element 'name' is not valid with respect to its type, 'agentNameTypeType'.</message>
```

The error message refers to `line="247" column="67"` in the XML:

```
244. <agentSet>
245.   <display>school of Michelangelo Buonarroti (painter); formerly attributed to Paul, Priya (collector);
246.   <agent>
247.     <name type="person" vocab="viaf" refid="24585191">Michelangelo Buonarroti</name>
248.     <role>painter</role>
249.     <attribution>school of</attribution>
250.   </agent>
251.   <agent>
252.     <name type="personal" vocab="viaf" refid="277848046">Paul, Priya</name>
```

The first line of the message states that “person” is not valid because it is not included in the list of possible values: “personal, corporate, family, other”.

The second line adds that the error occurred within attribute “type” of element `<name>`, and the provided value for “type” was not valid.

Result:

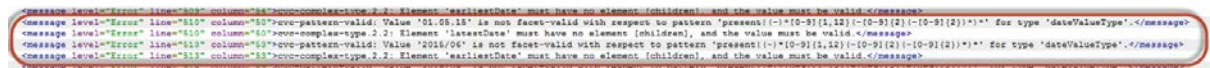
Values of `<name type="">` are controlled by the schema.

You need to delete “person” and use of one of the allowed values, i.e. “personal, corporate, family, other”.

8.3. Example 3 (date pattern):

This example consists of two similar problems:

Error message (screenshot):





Error message (transcribed):

```
<message level="Error" line="510" column="50">cvc-pattern-valid: Value '01.05.15'
is not facet-valid with respect to pattern 'present|(-)*[0-9]{1,12}(-[0-9]{2}(-[0-
9]{2}))*' for type 'dateValueType'.</message>
<message level="Error" line="510" column="50">cvc-complex-type.2.2: Element
'latestDate' must have no element [children], and the value must be
valid.</message>

<message level="Error" line="513" column="53">cvc-pattern-valid: Value '2015/06' is
not facet-valid with respect to pattern 'present|(-)*[0-9]{1,12}(-[0-9]{2}(-[0-
9]{2}))*' for type 'dateValueType'.</message>
<message level="Error" line="513" column="53">cvc-complex-type.2.2: Element
'earliestDate' must have no element [children], and the value must be
valid.</message>
```

The error message refers to `line="510" column="50"` and `line="513" column="53"` in the XML:

```
506. <dateSet>
507. <display>a predefined work date display</display>
508. <date type="creation">
509. <earliestDate>01.05.15</earliestDate>
510. <latestDate>01.05.15</latestDate>
511. </date>
512. <date type="alteration">
513. <earliestDate>2015/06</earliestDate>
514. <latestDate>2015/06</latestDate>
515. </date>
516. </dateSet>
```

The first lines of each message state that “01.05.15” and “2015/06” are not valid because they do not follow the predefined pattern for dateValues.

The second lines add that the error occurred within element `<latestDate>` (respective `<earliestDate>`), and the element must be valid.

Result:

Values of `<earliestDate>` and `<latestDate>` are pattern-controlled by the schema.

You need to write the dates in the correct format, i.e. “2015-05-01” or “2015-06”.

8.4. Example 4 (work/image ID):

Error message (transcribed):

```
<message level="Error" line="239" column="18">cvc-datatype-valid.1.2.1: '2' is not
a valid value for 'NCName'.</message>
<message level="Error" line="239" column="18">cvc-attribute.3: The value '2' of
attribute 'id' on element 'work' is not valid with respect to its type,
'ID'.</message>
```

The error message refers to `line="239" column="18"` in the XML:

```
238. </image>
239. <work id="2">
240. <agentSet>
241. <display>Carlo
```

The first line of the message states that “2” is not a valid NCName, which stands for “Non-colonized Names” (cf. <http://www.w3.org/TR/xmlschema-2/#NCName>).



The second line adds that the error occurred within attribute "id" of element <work>, and the provided value was not valid.

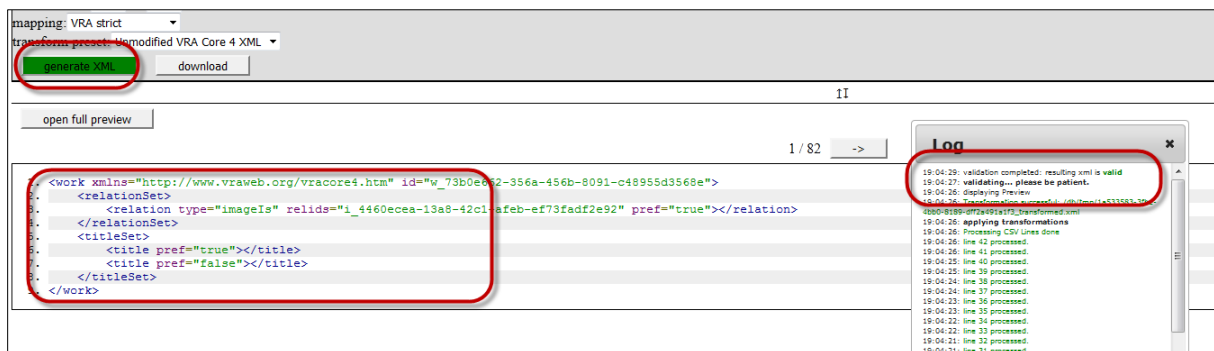
Result:

Values of <work id=""> are regulated by the schema. The "[VRA Core4 Element Description](#)" explains in a note: "The XML id attribute must begin with a character. The convention used here prefixes a numeric value with w_ for works, c_ for collections, and i_ for images."

You need to provide a work ID that begins with "w_".

8.5. Example 5 (empty output):

If your data is processed without an error and was successfully validated, but the output does not show any of your data, you might need to check the encoding of the .csv file. We had a test case where data was by mistake encoded in "Unicode" instead of "UTF-8" and the result was this "empty" output. Note that some elements are there, but no descriptive metadata.



Result:

The .csv file must be encoded in "UTF-8".

See the document "Converting xlsx to csv in Calc.pdf" also available in the doc folder on GitHub:
<https://github.com/exc-asia-and-europe/csv2xml/tree/master/doc>.



9. Generating RDF XML

In addition to generating VRA Core 4 XML it is possible to transform the data to RDF in both, simple and advanced mode.

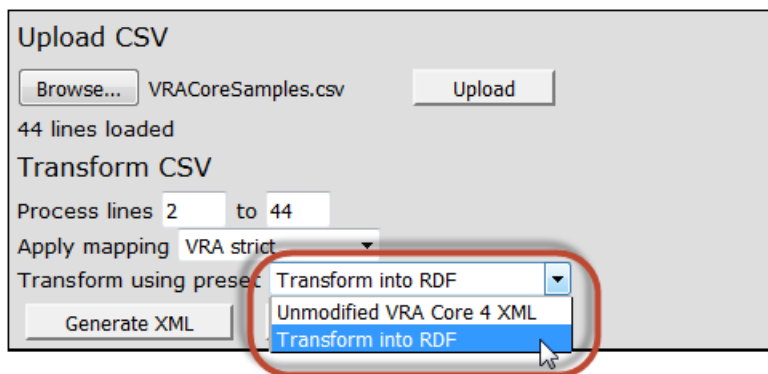
The tool makes use of the XSLT stylesheet as provided by the VRA-RDF-Project

<https://github.com/mixterj/VRA-RDF-Project>.

Transforming data to RDF uses the same workflow as transforming to XML and just performs another transformation, i.e. from the XML data into RDF/XML. Since the tool does not validate RDF output, make sure the VRA Core 4 XML data is valid.

9.1. Simple mode

In Simple mode upload your .csv data as described above and apply a mapping. Then go to "Transform using preset" and select "Transform into RDF".



Now click "generate XML".

9.2. Advanced mode

In Advanced mode you first need to generate the XML and then go to "Transform using preset" and select "Transform into RDF". Note that two new entries appear in the "Apply XSLs (for transformation):" box: "xsl/vra2xml.xsl" and "xsl/clean-up_rdf-comments.xsl".



Upload CSV

VRACoreSamples.csv

44 lines loaded

Transform CSV

Process lines to

Apply mapping

Transform using preset

Apply XSLs (for transformation):

xsl/cleanup-vra.xsl
xsl/vra2rdf.xsl
xsl/clean-up_rdf-comments.xsl

(modifying the predefined selection may break pagination)

Validate against:

Once generated, the RDF XML will be displayed in the "Preview" and can be downloaded.

1 / 80 ->

```
1. <rdf:Description xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" rdf:about="#">
2.   <rdf:type rdf:resource="http://purl.org/vra/CreativeWork"></rdf:type>
3.   <rdf:type rdf:resource="http://vocab.getty.edu/aat/300007501"></rdf:type>
4.   <rdf:type rdf:resource="http://vocab.getty.edu/aat/300170443"></rdf:type>
5.   <rdf:type rdf:resource="http://vocab.getty.edu/aat/300007466"></rdf:type>
6.   <vra:creator xmlns:vra="http://purl.org/vra/">
7.     <rdf:Description rdf:about="http://vocab.getty.edu/ulan/500125235">
8.       <rdf:type rdf:resource="http://purl.org/vra/Agent"></rdf:type>
9.       <rdf:type rdf:resource="http://purl.org/vra/Person"></rdf:type>
10.      <vra:name>unknown French</vra:name>
11.    </rdf:Description>
12.  </vra:creator>
13.  <vra:culturalContext xmlns:vra="http://purl.org/vra/">
14.    <rdf:Description rdf:about="http://vocab.getty.edu/aat/French">
15.      <vra:name>French</vra:name>
16.    </rdf:Description>
17.  </vra:culturalContext>
18.  <vra:dateCreated xmlns:vra="http://purl.org/vra/">begun 1194 (creation); consecrated 1260 (other)</vra:dateCreated>
19.  <vra:wasCreated xmlns:vra="http://purl.org/vra/">
20.    <rdf:Description>
21.      <rdf:type rdf:resource="http://purl.org/vra/Event"></rdf:type>
22.      <vra:name>Creation</vra:name>
23.      <vra:startDate>1194</vra:startDate>
24.      <vra:endDate>1194</vra:endDate>
25.    </rdf:Description>
26.  </vra:wasCreated>
27.  <vra:description xmlns:vra="http://purl.org/vra/">The present cathedral was constructed on the foundations of the
28.  <vra:placeOfSite xmlns:vra="http://purl.org/vra/">
29.    <rdf:Description>
30.      <rdf:type rdf:resource="http://purl.org/vra/Place"></rdf:type>
31.      <vra:containedIn>
32.        <rdf:Description rdf:about="http://vocab.getty.edu/tgn/7033010">
33.          <rdf:type rdf:resource="http://purl.org/vra/Place"></rdf:type>
34.          <vra:name>Chartres Cathedral</vra:name>
35.        </rdf:Description>
36.      </vra:containedIn>
37.    </vra:containedIn>
38.    <rdf:Description rdf:about="http://vocab.getty.edu/tgn/7008267">
```



10. Processing works with multiple image records

Sometimes work records can have multiple image records assigned to it. For example, in the VRA Core sample records the work in Example 23 has four images:

1	IMAGE_Href	IMAGE_ID	IMAGE_VendorID	IMAGE_LocalRepoName	WORK_ID	WORK_TitlePref	WORK_Title	WORK_TitleAlt
11	http://core.vraweb.org/examples/html/example019_full.html	108	8062207	(No VRA Data Standards Comi	10	Shower: Afternoon dresses by Doeuillet	Giboulée: Robe	
12	http://core.vraweb.org/examples/html/example023_full.html	109	1A1-PA-PR-D-A4	(No VRA Data Standards Comi	11	Le fabbriche e i disegni di Andrea Palladio	Le fabbriche e i	
13	http://core.vraweb.org/examples/html/example023_full.html	110	1A1-PA-PR-D-A3	(No VRA Data Standards Comi	11	Le fabbriche e i disegni di Andrea Palladio	Le fabbriche e i	
14	http://core.vraweb.org/examples/html/example023_full.html	111	1A1-PA-PVI-D-A1	(No VRA Data Standards Comi	11	Le fabbriche e i disegni di Andrea Palladio	Le fabbriche e i	
15	http://core.vraweb.org/examples/html/example023_full.html	118	1A1-PA-PR-D-A2	(No VRA Data Standards Comi	11	Le fabbriche e i disegni di Andrea Palladio	Le fabbriche e i	
16	http://core.vraweb.org/examples/html/example020_full.html	114	8070507	(No VRA Data Standards Comi	12	Do you know your lesson?: Tailored enser	Sais-tu ta leçon	

Therefore, the *VRA Core 4 XML Transform Tool* can only use one data set to create the work xml from. The decision will be made by looking at the contents of "WORK_ID" and using just the first set (for the work).

In the example, only the information from line 12 will be used to create the WORK record.

1	IMAGE_Href	IMAGE_ID	IMAGE_VendorID	IMAGE_LocalRepoName	WORK_ID	WORK_TitlePref	WORK_Title	WORK_TitleAlt
11	http://core.vraweb.org/examples/html/example019_full.html	108	8062207	(No VRA Data Standards Comi	10	Shower: Afternoon dresses by Doeuillet	Giboulée: Robe	
12	http://core.vraweb.org/examples/html/example023_full.html	109	1A1-PA-PR-D-A4	(No VRA Data Standards Comi	11	Le fabbriche e i disegni di Andrea Palladio	Le fabbriche e i	
13	http://core.vraweb.org/examples/html/example023_full.html	110	1A1-PA-PR-D-A3	(No VRA Data Standards Comi	11	Le fabbriche e i disegni di Andrea Palladio	Le fabbriche e i	
14	http://core.vraweb.org/examples/html/example023_full.html	111	1A1-PA-PVI-D-A1	(No VRA Data Standards Comi	11	Le fabbriche e i disegni di Andrea Palladio	Le fabbriche e i	
15	http://core.vraweb.org/examples/html/example023_full.html	118	1A1-PA-PR-D-A2	(No VRA Data Standards Comi	11	Le fabbriche e i disegni di Andrea Palladio	Le fabbriche e i	
16	http://core.vraweb.org/examples/html/example020_full.html	114	8070507	(No VRA Data Standards Comi	12	Do you know your lesson?: Tailored enser	Sais-tu ta leçon	

You need to make sure that different lines containing information about the same work are identical. In addition, you need to make sure that your WORK_ID's are unique.

In the XML output, the *VRA Core 4 XML Transform Tool* will not only relate the image records to the respective WORK, but it will also relate the work record to all IMAGE records.

1286	<work id="w_11">	
1287	<agentSet>	[15 lines]
1303	<contextSet>	[3 lines]
1307	<dateSet>	[6 lines]
1314	<descriptionSet>	[3 lines]
1318	<locationSet>	[7 lines]
1326	<materialSet>	[4 lines]
1331	<measurementsSet>	[3 lines]
1335	<relationSet>	
1336	<display>depicts Palazzo della Ragione; </display>	
1337	<relation type="depicts">Palazzo della Ragione</relation>	
1338	<relation type="imageIs" relids="i_7f164d5d-7b9d-4825-86d3-483e8b9d782d">Details, Palazzo della Ragione (plate T XXXI)</relation>	
1339	<relation type="imageIs" relids="i_c23f56e6-7b0a-40e3-b829-5ec574101dc1">Section, Palazzo della Ragione (plate T XXX)</relation>	
1340	<relation type="imageIs" relids="i_7dfcbe6b-a4bd-4ec7-8e2a-d30c0c00150f">Palazzo Vicentino; plan (plate T XXVII)</relation>	
1341	<relation type="imageIs" relids="i_27469e50-189b-4c4a-8a57-93c6ac269552">Elevation, Palazzo della Ragione (plate T XXIX)</relation>	
1342	</relationSet>	
1343	<stylePeriodSet>	
1344	<display>Eighteenth century; Palladian</display>	

Figure 11: XML output of Example 23, relationSet, showing relations to all four IMAGE records



11. Contact

If you have problems or questions please contact Matthias Arnold at matthias.arnold@uni-hd.de.

Contact details

matthias.arnold@uni-hd.de or
arnold@asia-europe.uni-heidelberg.de

Phone: +49 (0) 6221 - 54 4094

Fax: +49 (0) 6221 - 54 4012

Skype: matz-skype

Office address

Karl Jaspers Centre, Room 005b

MediaLab

Karl Jaspers Centre, Room 005c

Further links

Web-Portal of Heidelberg Research Architecture

<http://hra.uni-hd.de/>

Ziziphus VRA Core 4 XML editor:

<https://github.com/exc-asia-and-europe/ziziphus>

Ziziphus - guidelines and help files:

<http://ziziphus-help.uni-hd.de>



12. Appendix: Elements, displays and repetitions

WORK

[workID]

Agent display

3x nameType – name – nameVocab – nameRefid – role

CulturalContext display

2x culturalContext – vocab – refid

Date display

2x type – earliestDate – latestDate

Description display [display = description]

descriptionSource

Inscription display [display = text]

Location display

Location notes

2x type – Name – NameVocab – NameRefID – ObjRefID – Geo – GeoVocab – GeoRefid

Material display

4x material – vocab – refid

Measurements display

2x extent – unit – value1-4 – type1-4

Relation display

2x relatedWork – type

[transform adds work-image link(s)]

Rights display [display = text]

Source display

StateEdition display

Style/Period display

4x stylePeriod – vocab – refid

Subject display

8x subject – vocab – refid – type

Technique display

4x technique – vocab – refid

TextRef display

Title display



titlePref, titlePrefType, titleAlt, titleAltType

Worktype display

3x worktype – vocab – refid

IMAGE

[IMAGE_Href (-> href); Filename (-> refid)]

Agent display

1x name – nameVocab – nameRefid – role

Date display

Description display [display = description]

Inscription display [display = text]

Location display

[localRepoName, accessionID]

Material display

Measurements display

Relation display

[transform adds image-work link(s)]

Rights display

Rights notes

1x type – holder – text

Source display

[VendorName, VendorID]

1x sourceType, sourceValue

Subject display

3x subject – vocab – refid – type

Technique display

1x technique – vocab – refid

[if not provided: <technique vocab="AAT" refid="300237903">digital imaging</technique>]

Title display

1x type, title

WorkType display

1x technique – vocab – refid

[if not provided: <worktype vocab="AAT" refid="300215302">digital images</worktype>]



13. Appendix: Columns Full Template

IMAGE_Filename	WORK_LocationDisplay
IMAGE_Href	WORK_LocationNotes
IMAGE_AccessionID	WORK_Location1Type
IMAGE_VendorID	WORK_Location1Name
IMAGE_VendorName	WORK_Location1NameVocab
IMAGE_LocalRepoName	WORK_Location1NameRefID
	WORK_Location1ObjRefID
WORK_ID	WORK_Location1Geo
WORK_TitleDisplay	WORK_Location1GeoVocab
WORK_TitlePref	WORK_Location1GeoRefid
WORK_TitlePrefType	WORK_Location2Type
WORK_TitleAlt	WORK_Location2Name
WORK_TitleAltType	WORK_Location2NameVocab
WORK_AgentDisplay	WORK_Location2NameRefID
WORK_Agent1NameType	WORK_Location2ObjRefID
WORK_Agent1Name	WORK_Location2Geo
WORK_Agent1NameVocab	WORK_Location2GeoVocab
WORK_Agent1NameRefid	WORK_Location2GeoRefid
WORK_Agent1Role	WORK_MaterialDisplay
WORK_Agent1Attribution	WORK_Material1
WORK_Agent2NameType	WORK_MaterialVocab1
WORK_Agent2Name	WORK_MaterialRefid1
WORK_Agent2NameVocab	WORK_Material2
WORK_Agent2NameRefid	WORK_MaterialVocab2
WORK_Agent2Role	WORK_MaterialRefid2
WORK_Agent2Attribution	WORK_Material3
WORK_Agent3NameType	WORK_MaterialVocab3
WORK_Agent3Name	WORK_MaterialRefid3
WORK_Agent3NameVocab	WORK_Material4
WORK_Agent3NameRefid	WORK_MaterialVocab4
WORK_Agent3Role	WORK_MaterialRefid4
WORK_Agent3Attribution	WORK_MeasurementsDisplay
WORK_CulturalContextDisplay	WORK_Measurements1Extent
WORK_CulturalContext1	WORK_Measurements1Unit
WORK_CulturalContext1Vocab	WORK_Measurements1Value1
WORK_CulturalContext1Refid	WORK_Measurements1Type1
WORK_CulturalContext2	WORK_Measurements1Value2
WORK_CulturalContext2Vocab	WORK_Measurements1Type2
WORK_CulturalContext2Refid	WORK_Measurements1Value3
WORK_DateDisplay	WORK_Measurements1Type3
WORK_Date1Type	WORK_Measurements1Value4
WORK_EarliestDate1	WORK_Measurements1Type4
WORK_LatestDate1	WORK_Measurements2Extent
WORK_Date2Type	WORK_Measurements2Unit
WORK_EarliestDate2	WORK_Measurements2Value1
WORK_LatestDate2	WORK_Measurements2Type1
WORK_DescriptionDisplay	WORK_Measurements2Value2
WORK_DescriptionSource	WORK_Measurements2Type2
WORK_InscriptionDisplay	WORK_Measurements2Value3



WORK_Measurements2Type3	WORK_SubjectVocab7
WORK_Measurements2Value4	WORK_SubjectRefid7
WORK_Measurements2Type4	WORK_Subject8
WORK_RelationDisplay	WORK_SubjectType8
WORK_RelatedWork1	WORK_SubjectVocab8
WORK_RelationType1	WORK_SubjectRefid8
WORK_RelatedWork2	WORK_TechniqueDisplay
WORK_RelationType2	WORK_Technique1
WORK_RightsDisplay	WORK_TechniqueVocab1
WORK_SourceDisplay	WORK_TechniqueRefid1
WORK_StateEditionDisplay	WORK_Technique2
WORK_StylePeriodDisplay	WORK_TechniqueVocab2
WORK_StylePeriod1	WORK_TechniqueRefid2
WORK_StylePeriodVocab1	WORK_Technique3
WORK_StylePeriodRefid1	WORK_TechniqueVocab3
WORK_StylePeriod2	WORK_TechniqueRefid3
WORK_StylePeriodVocab2	WORK_Technique4
WORK_StylePeriodRefid2	WORK_TechniqueVocab4
WORK_StylePeriod3	WORK_TechniqueRefid4
WORK_StylePeriodVocab3	WORK_TextrefDisplay
WORK_StylePeriodRefid3	WORK_TextrefName
WORK_StylePeriod4	WORK_TextrefNameType
WORK_StylePeriodVocab4	WORK_TextrefRefid
WORK_StylePeriodRefid4	WORK_TextrefRefidType
WORK_SubjectDisplay	WORK_WorktypeDisplay
WORK_Subject1	WORK_Worktype1
WORK_SubjectType1	WORK_WorktypeVocab1
WORK_SubjectVocab1	WORK_WorktypeRefid1
WORK_SubjectRefid1	WORK_Worktype2
WORK_Subject2	WORK_WorktypeVocab2
WORK_SubjectType2	WORK_WorktypeRefid2
WORK_SubjectVocab2	WORK_Worktype3
WORK_SubjectRefid2	WORK_WorktypeVocab3
WORK_Subject3	WORK_WorktypeRefid3
WORK_SubjectType3	
WORK_SubjectVocab3	IMAGE_AgentDisplay
WORK_SubjectRefid3	IMAGE_Agent1Name
WORK_Subject4	IMAGE_Agent1NameVocab
WORK_SubjectType4	IMAGE_Agent1NameRefid
WORK_SubjectVocab4	IMAGE_Agent1Role
WORK_SubjectRefid4	IMAGE_DateDisplay
WORK_Subject5	IMAGE_DescriptionDisplay
WORK_SubjectType5	IMAGE_InscriptionDisplay
WORK_SubjectVocab5	IMAGE_LocationDisplay
WORK_SubjectRefid5	IMAGE_MaterialDisplay
WORK_Subject6	IMAGE_MeasurementsDisplay
WORK_SubjectType6	IMAGE_RelationDisplay
WORK_SubjectVocab6	IMAGE_RightsDisplay
WORK_SubjectRefid6	IMAGE_RightsNotes
WORK_Subject7	IMAGE_RightsType
WORK_SubjectType7	IMAGE_RightsHolder



IMAGE_RightsText
IMAGE_SourceDisplay
IMAGE_SourceValue
IMAGE_SourceType
IMAGE_SubjectDisplay
IMAGE_Subject1
IMAGE_SubjectType1
IMAGE_SubjectVocab1
IMAGE_SubjectRefid1
IMAGE_Subject2
IMAGE_SubjectType2
IMAGE_SubjectVocab2
IMAGE_SubjectRefid2
IMAGE_Subject3

IMAGE_SubjectType3
IMAGE_SubjectVocab3
IMAGE_SubjectRefid3
IMAGE_TechniqueDisplay
IMAGE_Technique
IMAGE_TechniqueVocab
IMAGE_TechniqueRefid
IMAGE_TitleDisplay
IMAGE_Title
IMAGE_TitleType
IMAGE_WorktypeDisplay
IMAGE_Worktype
IMAGE_WorktypeVocab
IMAGE_WorktypeRefid