CS598 – Data Curation Final Project

By Ved Chugh(vedpc2)

As part of Data Curation Coursework, we learnt the concept of Canonicalization. Which as per the course work – is a technique for determining representational identity and is a reasonable proxy for propositional identity. (CS598 – Week 10). **Canonical XML** is a normal form of XML, intended to allow relatively simple comparison of pairs of XML documents for equivalence; for this purpose, the Canonical XML transformation removes non-meaningful differences between the documents. Any XML document can be converted to Canonical XML. (https://en.wikipedia.org/wiki/Canonical_XML)

Thus, the task here is to have the two source files Consumer_Complaints_FileA.xml and Consumer_Complaints_FileB.xml which are consumer complaints file from old system and new system respectively.

SECTION 1 – The Files

The source files are in folder in submission - **Source Files**

File A: Consumer_Complaints_FileA.xml:

a. At first glance there is no DTD defined, neither internal nor external for the xml file.
b. Even though the file seems to have a formatting in place, and follows a strict availability of elements and attributes but without the proper DTD in place it would be difficult to verify and validate the xml file
c. Further analysis of the file shows that towards the end the last complaint with id – 14038 – doesn't strictly adhere to sequence of elements as compared to other complaint ids.
d. But apart from the above observations the file is cleaner and more readable version as compared to what I will point in later points when comparing to Consumer_Complaints_FileB.
e. Apart from the above observations the file has elements and attributes which I will highlight in DTD creation approach, and in xml normal form of the file.

There are eight complaints in the file and each complaint is identified by unique Complaint Id.

The file further has below elements and attributes:

⇒ consumerComplaints – The complaints filed by consumer.
⇒ complaint – This element occurs one or more time
⇒ event. – This element occurs one or more time
⇒ product - This element helps identify the product and sub products for which the complaint was logged. And has following Attribute list.

    i) productType – Type of product – which as per the file occurs once per complaint.
    ii) subProduct – Subcategory of Product. This as per the file seems to occur 0 or 1 time per complaint.

⇒ issue – This is the categorization of Issue and has details about the issueType and subissue as attributes.

    i) issueType – Category of Issue - which as per the file occurs once per complaint.
    ii) subissue – Subcategory of Issue. This as per the file seems to occur 0 or 1 time per complaint.

⇒ consumerNarrative – This is the actual text of complaint shared by consumer. This as per the file is optional thus can occur 0 or 1 time per complaint.

⇒ company – Here we have the details of the Respondent, and their demographic details. And all attributes appear to have at least/mandatory one occurrence per complaint.

    i) companyName
    ii) companyState
    iii) companyZip

⇒ submitted – This is the mode of submission like web, referral, phone.

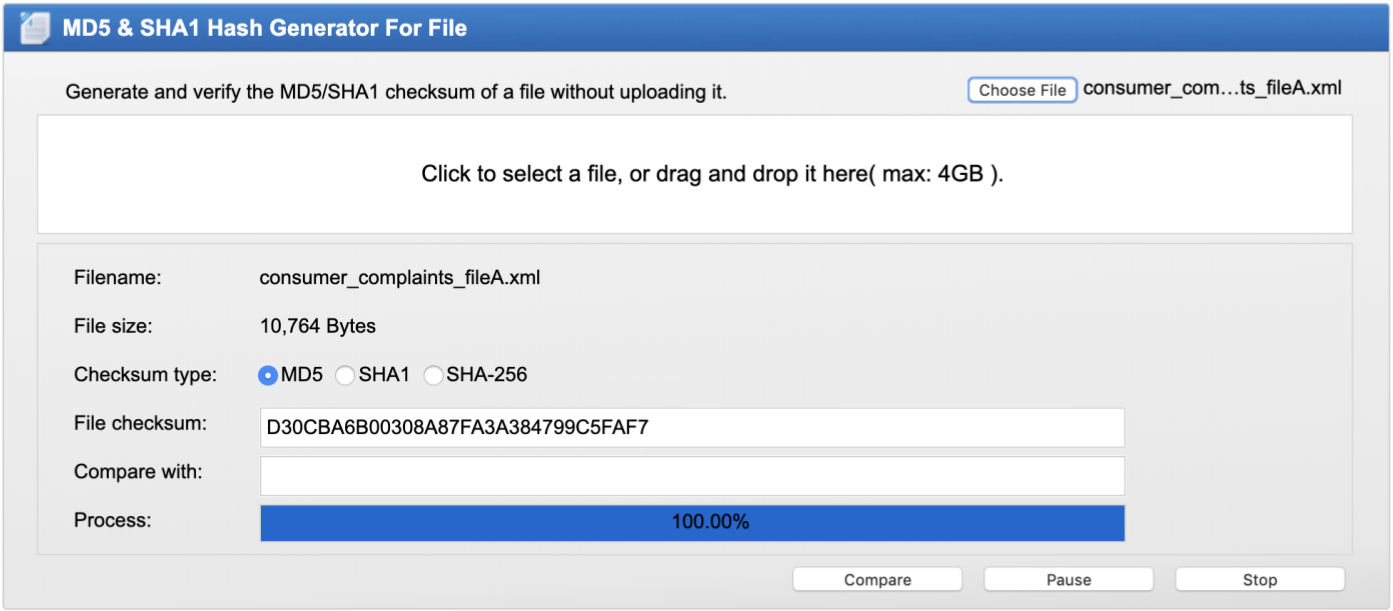
⇒ response – This is the response by respondent and further details if the response was whether it is provided on timely manner or not.
It also captures the category of response.
Further it also has details if it is a consumer disputed or not.
And has publicResponse information which gives if the reponse was chosen not to make public.

    i) publicResponse – Optional can occur 0 or 1 time per complaint/response.
    ii) responseType – Category of response and seems to have one occurrence.

XML validation yields to invalid file, as missing DTD, and even schema definition is Invalid.

Checksum details – MD5:

consumer_complaints_fileA.xml - D30CBA6B00308A87FA3A384799C5FAF7



MD5 & SHA1 Hash Generator For File

Generate and verify the MD5/SHA1 checksum of a file without uploading it. consumer_com…ts_fileA.xml

Click to select a file, or drag and drop it here(max: 4GB).

Filename: consumer_complaints_fileA.xml

File size: 10,764 Bytes

Checksum type: ● MD5 ○ SHA1 ○ SHA-256

File checksum: D30CBA6B00308A87FA3A384799C5FAF7

Compare with:

Process: 100.00%

File 2 - Consumer_Complaints_FileB.xml:

a. At first glance there is DTD defined but it is incomplete and doesn't help in documentation.
b. The file seems to have NO formatting in place and follows a NO strict availability of elements and attributes.
c. Elements and Attributes are not consistent and are missing in file as compared to counterpart old system file.
d. But apart from the above observations the file is NOT ALL cleaner and NEITHER readable version as compared to old system Consumer_Complaints_FileA
e. Apart from the above observations the file has elements and attributes which I will highlight in DTD creation approach, and in xml normal form of the file.

There are eight complaints in the file and each complaint is identified by unique Complaint Id.

The file further has below elements and attributes:

⇒ redaction – is an entity for consumerNarrative referenced redaction value. This is better approach though as rather than changing in multiple places, a future change can be addressed by change of entity value here itself.

⇒ consumerNarrative – Same as highlighted above the only difference we notice here is it has a refernce entity value &redaction.

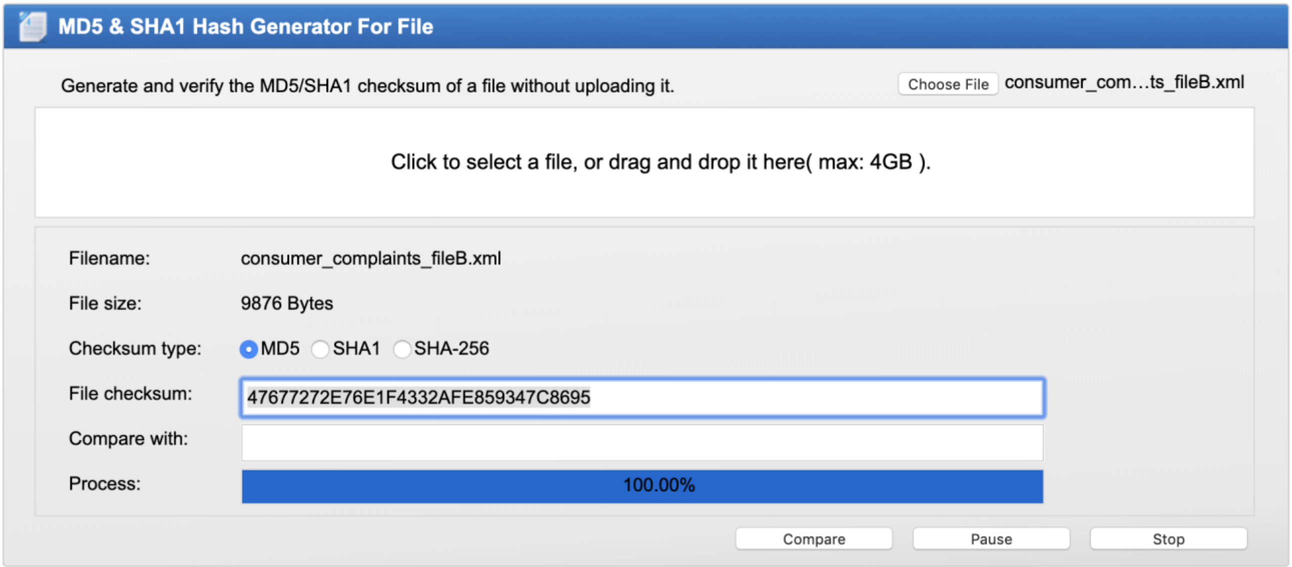
⇒ submitted – Here we see that submissionType which is "via" in file A has been moved as an attribute of parent entity Complaint.

⇒ response – Here the attribute timely has text value of "yes" or "no" rather than a standard flag value of "Y" and "N".

XML validation yields to invalid file, as missing DTD.

Checksum details – MD5:

consumer_complaints_fileB.xml - 47677272E76E1F4332AFE859347C8695



Generate and verify the MD5/SHA1 checksum of a file without uploading it.

consumer_com…ts_fileB.xml

Click to select a file, or drag and drop it here(max: 4GB).

Filename: consumer_complaints_fileB.xml

File size: 9876 Bytes

Checksum type: ● MD5 ○ SHA1 ○ SHA-256

File checksum: 47677272E76E1F4332AFE859347C8695

Compare with:

Process: 100.00%

SECTION 2 - DTD

DTD for file A and file B with formatting to have correct aligned DTD:

The DTD for both the files is in the Submission folder - **DTD_Source_files_with_Formating** containing:

1. consumer_complaints_fileA.xml
2. consumer_complaints_fileB.xml

SECTION 3 – Canonicalization

Canonicalization for file A and file B:

Both the files are in the Submission folder - **Canonicalized Files** containing:

1. Canc_consumer_complaints_fileA.xml
2. Canc_consumer_complaints_fileB.xml
3. Final_Merged_Canc_consumer_complaints_file.xml

Below are the MD5 checksum for all three files.

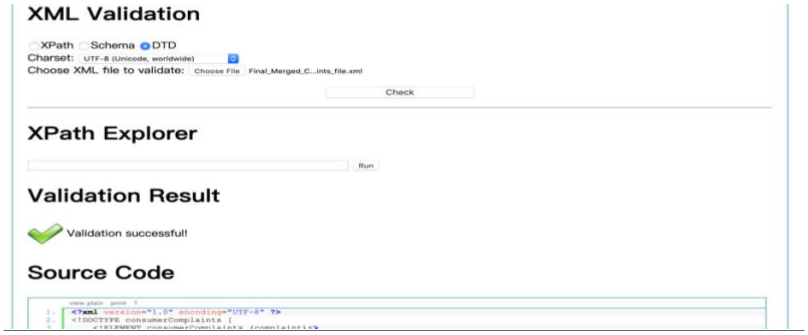
1) Canc_consumer_complaints_fileA.xml - C0CAB6445665D2E8FB3905C7A32B80DE
2) Canc_consumer_complaints_fileB.xml - 6532E5E29F2A879A0141451524BD8E17
3) Final_Merged_Canc_consumer_complaints_file.xml - C0CAB6445665D2E8FB3905C7A32B80DE

SECTION 4 – DTD for Canonicalized files

The DTD for all the files are included internally in respective files.

SECTION 5 – XML validation of Canonicalized files using DTD

All the individual files have been validated using DTD and they pass validation



XML Validation

○ XPath ○ Schema ● DTD
Charset: UTF-8 (Unicode, worldwide)
Choose XML file to validate: Choose File Final_Merged_C...ints_file.xml

Check

XPath Explorer

Run

Validation Result

✓ Validation successful!

Source Code

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE consumerComplaints [
    <!ELEMENT consumerComplaints (complaint)+>
```

SECTION 6 – Canonicalization steps and reflections

Canonicalization steps and relfections is included in file - **CS598-Final-Project-SECTION-6–Canonicalization-steps-reflections.docx**

SECTION 7 – Part 2 - Memo

Memo is included in Submission file - **CS598-Final-Project-Memo-Vedpc2.docx**