TOIF XMI Conformation Testing Report.

Summary

In order to guide the development and validation of the TOIF output, a schema has been created which describes the elements required in the TOIF XMI. XSD (XML Schema Definition) is an newer and more powerful successor of DTD (Document Type Definition), allowing more constraints on a XML file's structure and a richer datatyping system. A copy of the schema can be found with the adaptors (toifXmiSchema.xsd).

Modification of the Schema

During the development of the TOIF Adaptors, it was found that additions would need to be made to the original conceptual model, and hence the schema:

Finding *has* SFP identifier

> Possibility: The finding should, if involved with security concerns, be mapped to a suitable SFP.
>
> Note: Even if there is no CWE for this finding there may still be a suitable SFP.

Finding *has* Cluster identifier

> Possibility: The finding should always have a Cluster identifier.
>
> Note: The finding should always have a Cluster identifier, even if it is just "Observation", indicating that there may not be any real security issues.

TOIF Segment *references* File

> Possibility: The segment should be related to one file.
>
> Note: Because the adaptors are run on only one file at a time, each segment is ultimately related to only one file.

Slight changes to the physical organization were also made, to reduce redundancy in each TOIF output file and save space. The housekeeping facts which facilitate correct management during the multiple phases of the TOIF tool chain have now been moved to their own file within the adaptors output directory. It still conforms to the current TOIF Schema and therefore should validate in the same way as the other segments. This new file is always labeled GENERAL_INFORMATION.toif.xml.

Validation Method

XML Validation is performed using a tool such as xmllint. Xmllint parses one or more XML files making sure that the file follows the structure and contains the correct elements as provided by the schema.

Xmllint can take a number of options, but the most useful for validating the TOIF ouput are:

`--noout`

> Suppress output. By default, xmllint outputs the result tree. We don't need to have each file printed in order to validate.

`--schema <schema>`

> Use a W3C XML Schema file named <schema> for validation. The schema in this case is the toifXmiSchema.xsd provided with the adaptors.

Note: The xmllint tool can run on many xml files if the globbing '*' is used as the input file.

The full xmllint command on Linux:

```
xmllint --noout --schema toifXmiSchema.xsd *
```

Successful output will look like:

```
AbstractDecoder.class.toif.xml validates
AbstractFeatureExtractor.class.toif.xml validates
AbstractSausageMaker.class.toif.xml validates
AbstractSausageMaker$Cluster.class.toif.xml validates
AbstractSausageMaker$ClusterComparator.class.toif.xml validates
AbstractVoiceActivityDetector.class.toif.xml validates
...
```

A failure on a file would look like:

```
NamesConversion.class.toif.xml:15: element nme: Schemas validity error :
Element 'nme': This element is not expected. Expected is ( name ).
NamesConversion.class.toif.xml fails to validate
```