Modelling humanities data with TEI-XML

SCHOLARLY EDITING AND MANUSCRIPT CATALOGUING IN THE DIGITAL AGE

Dr Katarzyna Anna Kapitan
9 October 2024

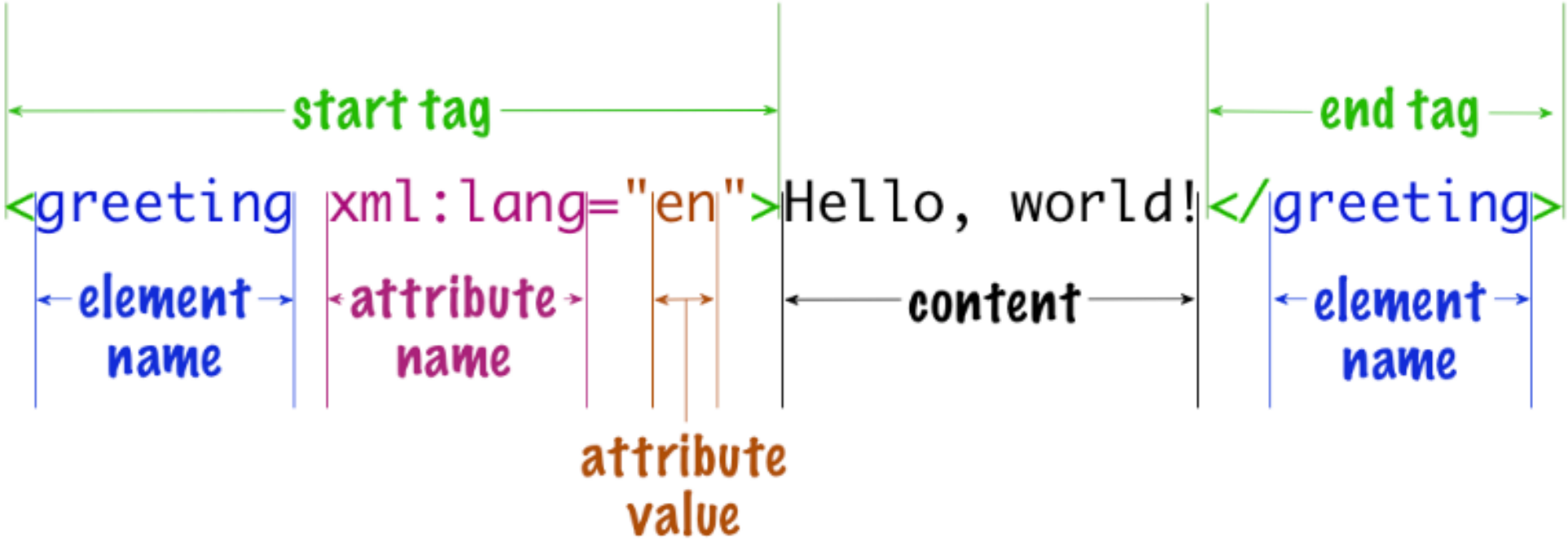
Course Materials

- **GitHub Repo:** https://github.com/KAKDH/ENC_TNAH_2024/

XML Recap

- XML is an international non-proprietary standard, which is widely used to export, share, and store structured data.
- XML is expressed in plain text, so it's hardware and software independent.

XML Recap



Elements Recap

Model:

<Element>

Content

</Element>

Example:

<person>

Paul Simon

</person>

Attributes Recap

Model:

```
<Element Attribute1="Value1">
        Content
</Element>
```

Example:

```
<person job="musician">
        Paul Simon
</person>
```

Encoding choices: Attribute or Element

Example 1:

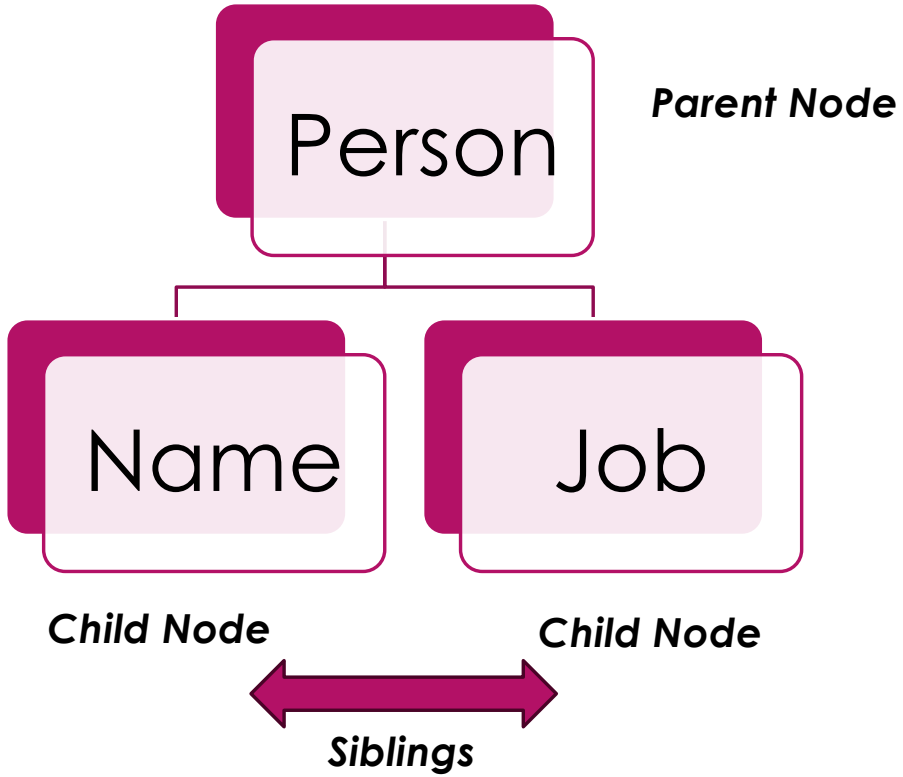
```
<person job="musician">
        Paul Simon
</person>
```

Example 2:

```
<person>
    <name>
            Paul Simon
    </name>
    <job>
            musician
    </job>
</person>
```

Element Nesting

```
<person>
    <name>
        Paul Simon
    </name>
    <job>
        musician
    </job>
</person>
```



```
<?xml version="1.0" encoding="UTF-8"?>
    <workshop name="XML_workshop”>
        <instructors>
            <name>
                <firstName>Katarzyna</firstName>
                <lastName>Kapitan</lastName>
            </name>
        </instructors>
        <participants>
            <name>
                <firstName>John</firstName>
                <lastName>Doe</lastName>
            </name>
            <name>
                <firstName>Anna</firstName>
                <lastName>Smith</lastName>
            </name>
            <name>
                <firstName>Jan</firstName>
                <lastName>Kowalski</lastName>
            </name>
        </participants>
    </workshop>
```

Exercise 1: AM 30 fol.

https://github.com/KAKDH/ENC_TNAH_2024/tree/main/Classes/Week2/Week2_Exercises/Week2_Ex1

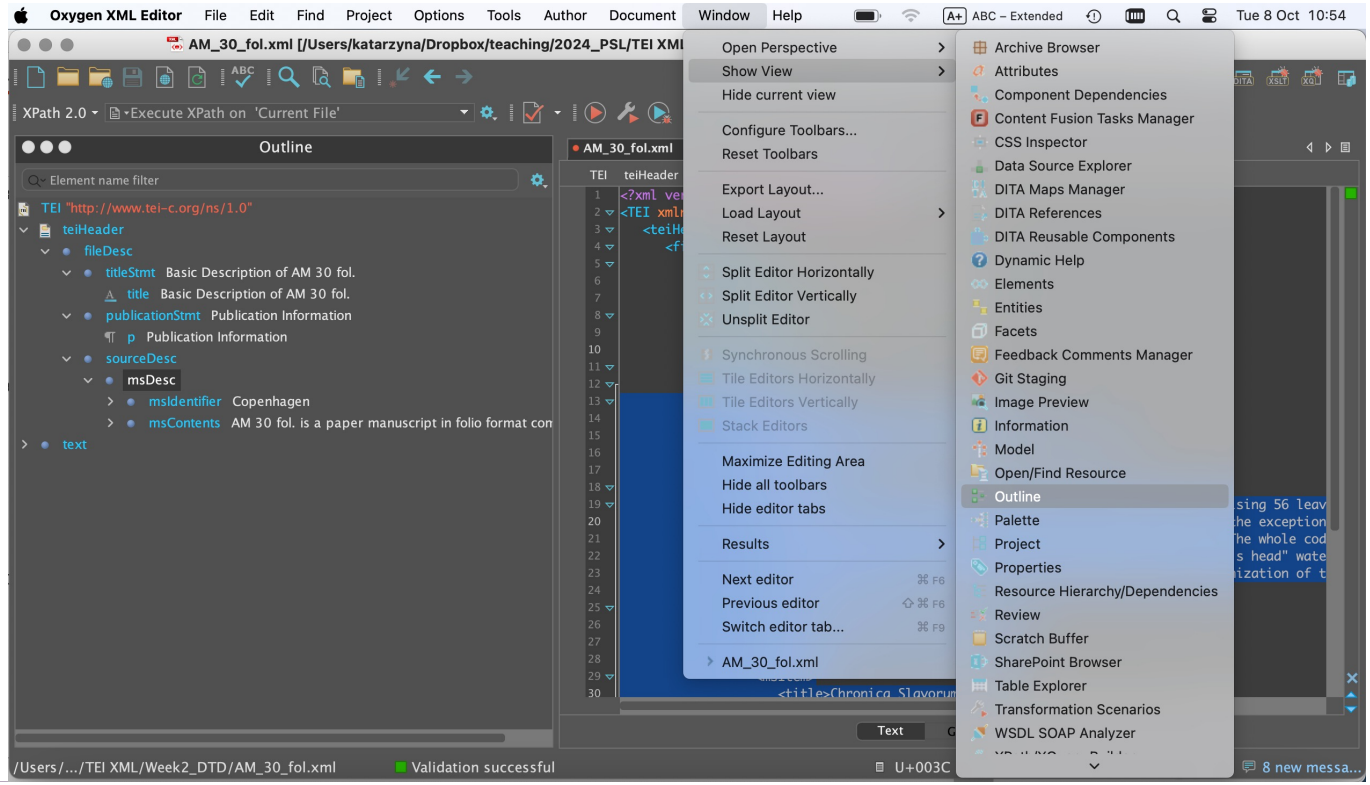
```
<msDesc>
        <msIdentifier>
          <settlement>Copenhagen</settlement>
[…]
        </msIdentifier>
        <msContents>
[…]
          <msItem>
            <title>Chronica Slavorum</title>
            <author>Helmold of Bosau</author>
          </msItem>
        </msContents>
</msDesc>
```

- **Task:** Analyse the structure of an XML document.
  - In PowerPoint, using "Smart Art" create a tree illustrating the structure of msDesc, including all descendants (children of children).
  - Which element is the parent element of msDesc?
  - Which elements are siblings of the parent element of msDesc?
  - Which elements are children elements of msDesc?

Exercise 1: AM 30 fol.xml

- Check your answers in Outline

- Oxygen XML Editor -> Window -> Show View -> Outline



DTD (Document Type Definition)
&
DOCTYPE (Document Type Declaration)

DTD (Document Type Definition)

- A **document type definition** (DTD) is a specification that defines the valid building blocks of an XML document.
- A DTD defines the document structure with a list of validated elements and attributes.
- A DTD can be declared inline inside an XML document, or as an external reference
- The DTD specification file can be used to validate XML documents.

Source: https://en.wikipedia.org/wiki/Document_type_definition

DTD (Document Type Definition)

- DTDs describe the structure of a class of documents via
  - **Element declarations** (describing elements and their relationship)
  - **Attribute-list declarations** (describing attributes and their values)

Source: https://en.wikipedia.org/wiki/Document_type_definition

DTD (Document Type Definition)

- **Element Declarations** list the elements which are allowed within the document.
- **Element Declarations** specify whether and how declared elements may be nested (contained within each element).

Source: https://en.wikipedia.org/wiki/Document_type_definition

Element Declarations

- <!ELEMENT **ElementName ElementSpec**>
- **Specification** of the **Element** can have different values, for example
  - **EMPTY**: for specifying that the defined element allows no content.
  - **ANY**: for specifying that the defined element allows any content.
  - an **expression** in brackets (), specifying the only elements allowed as direct children in the content of the defined element, including:
    - **#PCDATA**: parsed character data for specifying that the defined element allows textual content.

Element Declarations

- **DTD**: <!ELEMENT lb EMPTY>
  - Element Name: lb
  - Line beginning (lb) 'marks the beginning of a new (typographic) line in some edition or version of a text' (*TEI Guidelines*).
  - Element Specification: EMPTY
- **XML:** <lb/>

Element Declarations

- **DTD:** <!ELEMENT title (#PCDATA)>
  - Element Name: title
  - Element Specification: Contains #PCDATA (i.e. textual content)
- **XML:** <title> My title </title>

Element Declarations

- **DTD:** <!ELEMENT publication (title, author, date)>
- **XML:**

```
<publication>
        <title></title>
        <author></author>
        <date></date>
</publication>
```

The elements included in the declaration of another element need their own declaration.

Element Declarations

- **Sequence list** – a list of one or more content particles. It is specified within parentheses and separated by a comma. **All the content particles must** appear successively as direct children in the content of the defined element.
  - **DTD:** <!ELEMENT publication **(title, author, date)**>
- **Choice list** – a mutually exclusive list of two or more content particles. It is list specified within parentheses and separated by a pipe. **Only one of these content particles may** appear in the content of the defined element at the same position.
  - **DTD:** <!ELEMENT publication **(title | author |date)**>

Element Declarations

- **Quantifiers:**
  - + for specifying that there must be one or more occurrences of the item – one or more
  - * for specifying that any number of occurrences is allowed (the item is optional) – zero or more
  - ? for specifying that there must **not** be more than one occurrence (the item is optional) – zero or one
- **DTD:** <!ELEMENT publication (title, **author+**, date, **publicationPlace?**)>

Exercise 2:1

- Open the XML file bibliography_dtd_internal.xml in Oxygen
- Add a new element publisherName as a child of the publication element.
- Make sure it required, i.e. there must be only one publisherName element per publication.

- Adjust your encoding of your bibliography accordingly.

Attribute-list declaration

- **Attribute-list Declarations** list the attributes which are allowed for each declared element.
- **Attribute-list Declarations** specify the type of each attribute value, and/or an explicit set of valid values.

Attribute-list declaration

- **<!ATTLIST ElementName AttributeName DataType Value>**
- An attribute list specifies the list of all possible attribute associated with the element type.
- For each possible attribute, it contains:
  - the declared name of the attribute,
  - its data type (or a list of its possible values),
  - its default value (or usage)
- **Example: <!ATTLIST date when CDATA #REQUIRED>**

Attribute-list declaration

- **Model: <!ATTLIST ElementName AttributeName DataType Value>**
- The most common values for **DataType** are:
  - **CDATA (characters data)** – value of the attribute can be any textual value.
  - **ID (identifier)** – value of the attribute must be a valid identifier. It is used to define the current element.
  - **IDREF** (reference to an identifier) – value of the attribute must be a valid identifier and must be referencing the unique element with an ID.
  - a defined list of values within parenthesis.
- **Example: <!ATTLIST date when CDATA #REQUIRED>**

Attribute-list declaration

- **Model: <!ATTLIST ElementName AttributeName DataType Value>**
- The most common values for **Value** are:
  - *value* – the default value of the attribute
  - #REQUIRED – the attribute is required
  - #IMPLIED the attribute is optional;
  - #FIXED the attribute has a fixed value
- **Example: <!ATTLIST date when CDATA #REQUIRED>**

Exercise 2:2

- Open the XML file bibliography_dtd_internal.xml in Oxygen
- Create a closed list of attributes for the types of publications, the values of the attribute should be *book, book chapter, journal article*, make the attribute required.
- Adjust your encoding of your bibliography accordingly.

DTD & DOCTYPE

- A DTD is associated with an XML document by means of a **document type declaration (DOCTYPE)**.
- The DOCTYPE appears in near the start of an XML document.
- The declaration establishes that the document is an instance of the type defined by the referenced DTD.

DOCTYPE

- DOCTYPEs make two sorts of declarations:
  - an optional internal subset
    - **<!DOCTYPE RootElement [** *<!-- internal subset declarations -->* **]>**
  - an optional external subset:
    - **<!DOCTYPE RootElement SYSTEM** "myDtdFile.dtd"**>**
    - **<!DOCTYPE RootElement PUBLIC** "/quotedFPI/" "/quotedURI/" **>**

Exercise 3

- Open the XML file bibliography_dtd_external.xml in Oxygen
- Associate the DTD file bibliography_dtd_external.dtd with your XML file to validate, follow the model:

  <!DOCTYPE RootElementOfYourDTDFile SYSTEM "NameOfYourDTDFile.dtd">
- Add a new element publisherName as a child the publication, make it optional, but restrict its use to max one occurrence.
- Encode one more publication to your XML file, the details of the publication are in the comment at the bottom of the file.
- Make all the changes in your DTD that are necessary for you to be able to encode the second example.

Exercise 4

- Create a stand-alone DTD for the file AM_30_fol.xml (From Ex1).
- Associate it with AM_30_fol.xml, make sure that your XML file validates correctly.
- Send both files to Katarzyna by email (before 23:59 Tuesday 15/10):
  - katarzyna.kapitan [at] chartes.psl.eu

Useful links to explore (in addition to the reading list)

- XML DTD, *w3schools*:
  - https://www.w3schools.com/xml/xml_dtd_intro.asp
- Document type definition, *Wikipedia*:
  - https://en.wikipedia.org/wiki/Document_type_definition
- Document type declaration, *Wikipedia*:
  - https://en.wikipedia.org/wiki/Document_type_declaration