The XML Language

A MARKUP LANGUAGE

XML in theory

XML, what is it?

XML (W3C, 1998) : *eXtensible Markup Language*.

Simplification of a former language : **SGML** (*Standard Generalized Markup Language*, ISO standard, 1986).

Open (non proprietary) language, **independent** from the tools that use it. Can be read by many software ➔ **Stability** and **interoperability**.

XML is not a language, but a **metalanguage**: It is the base of other standards (XML-TEI, XML-RDF…).

Separation of the **content** (structure, meaning) from the **form** (design).

A polymorphic language

- With this flexibility, XML is everywhere**:**
  - in your **everyday life**: your laptop, your smartphone, your GPS, even the gas stations work with XML;
  - in the **editing world**: XML is a source file that can be transformed into other formats ➔ Printed (books, indices) or Digital (HTML, PDF, ePub, DocX…).

- **Use cases** of XML in editing:
  - *Aphrodisias in Late Antiquity*, Charlotte Roueché (King's College London)
    - http://insaph.kcl.ac.uk/ala2004/index.html
  - METOPES project (MRSH de Caen)
    - http://www.unicaen.fr/recherche/mrsh/document_numerique/projets/metopes

What do I need to make XML ?

- You need :
  - To know and respect the XML syntax (The next slide ;]) ;
  - An XML editor.

Nothing more, nothing less ! (for now…)

NB: Which editor do I have to use?

In fact, you can make XML with any editor (Notepad, Notepad++, SublimeText, Jedit, Oxygen XML Editor etc.). However, some editors have been developed specifically for XML: they are called "**XML editors**". They offer functionalities that help you to easily encode a text and to do it right. They are "**XML aware**" !

What do I need to make XML ?

Why use markup?

Markup is used in many different fields, for many different purposes: storing data, relating information, encoding understanding, preserving metadata

- Markup is a way of making our knowledge or understanding about a text explicit
- Markup makes strives to make explicit (to a machine) what is implicit (to a person)
- Markup assists us in facilitating re-use of the same material:
  - in different formats
  - in different contexts
  - by different sorts of users

More About XML

- An XML document is encoded as a linear string of characters
- It begins with a special processing instruction
- Element occurrences are marked by start and end-tags
- The characters **<** and **&** are Magic and must always be "escaped" using **<** or **&** if you want to use them as themselves
- Comments are delimited by **<!--** and **-->**
- Attribute name/value pairs are supplied on the start-tag and may be given in any order
- There are special attributes in the XML namespace like **xml:id** and **xml:lang**
- Attribute values are always quoted
- Everything is case-sensitive

Being Well-Formed

- There is a single root node containing the whole of an XML document
- Each subtree is properly nested within the root node
- Element/attribute names and values are always case sensitive
- Start-tags and end-tags are always mandatory (except there are combined start-and-end tags called 'empty elements' like <pb/> <gap/>)
- Attribute values are always quoted

✓ <seg> <w>some</w> <hi>text</hi> </seg>

✗ <seg> <w>some <hi></w> text</hi> </seg>

✓ <seg type="text">some text</seg>

✗ <seg type=text>some text</seg>

✗ <seg type="text"> some text <seg/>

✓ <seg type="text"> some text<gap/> </seg>

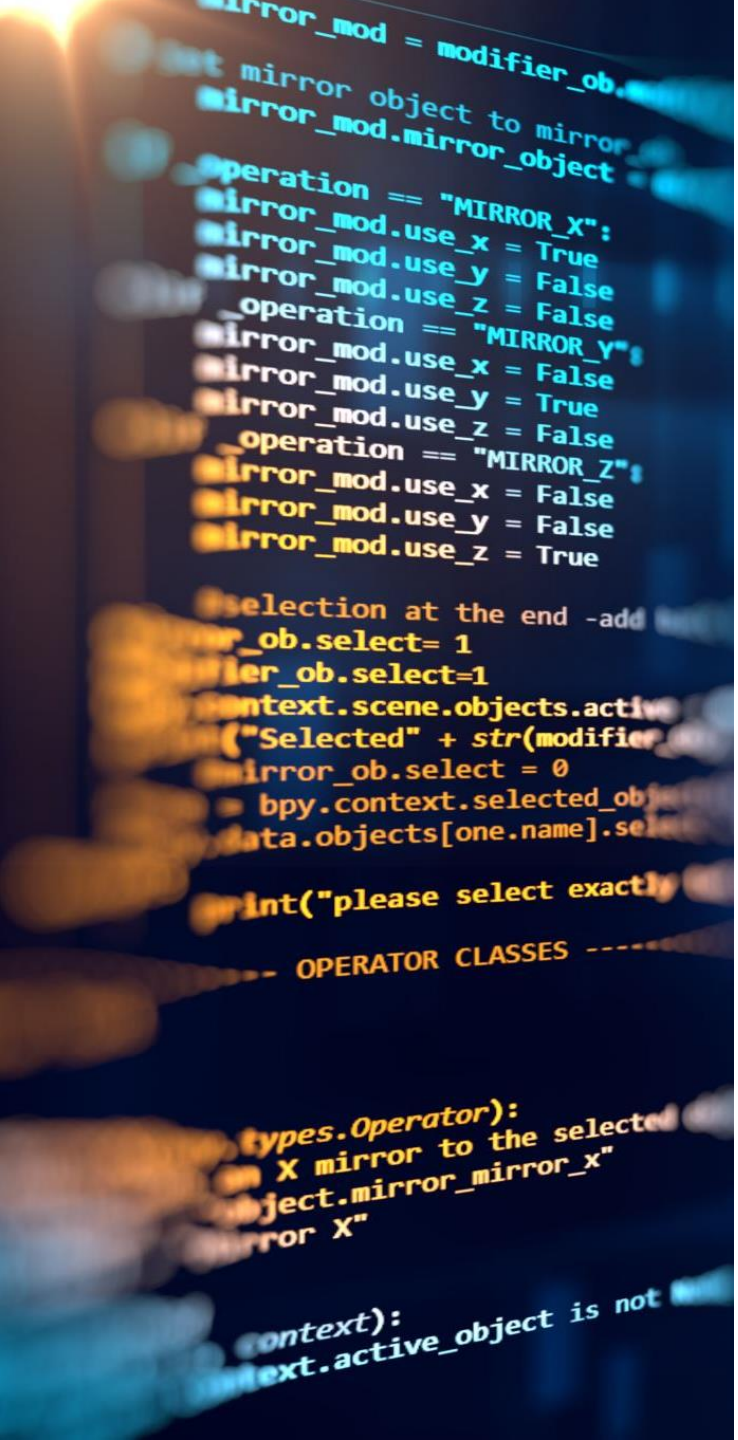
✗ <seg type="text">some text</Seg>

XML in practice

The XML syntax

- XML is based on a **tree structure** : one root, several nodes.

```
<?xml version="1.0" encoding="utf-8"?>
<library>
  <book>
    <author> J. R. R. Tolkien </author>
    <title> The Lord of the rings : The fellowship of the ring </title>
  </book>
  <book>
    <author> Lewis Carroll </author>
    <title> Alice’s Adventures in Wonderland </title>
  </book>
</library>
```



Elements

- Basic unit of the XML syntax
- An element is composed of **2 tags** :
  - The start tag begins with **<** and ends with **>** ;
  - The end tag begins with **</** and ends with **>**.

Ex. : <sentence>Chocolate is my life.</sentence>

- Careful ! Some elements are made of one tag and are called **empty elements**. They are a contraction between a start tag and a end tag. They don't contain data.

Ex. : <emptyTag />

Elements

The **Matriochka rule** : The tags are always nested inside each other. They <u>never</u> overlap !

<sentence><italic>XML is  best friend.</sentence></italic>

<sentence><italic>XML is my new best friend.</italic></sentence>

Attributes & values

<sentence type="citation">May the force be with you !
</sentence>

- An attribute specifies an element. It gives additional information about its language, its width, its height…
- Attributes work in couple : name="value".
- Attributes are only inserted in the start tag.
- An element can have any number of attributes.

Additional information about Oxygen

A TINY REMINDER OF THE MAIN FUNCTIONALITIES

Oxygen’s interface

recipes.xml [C:\Users\Elina\Documents\Thèse\Colloques et Séminaires\Fonte Gaia Summer School 2017\XML\exercises\recipes.xml] - <oXygen/> XML Editor (Usage académique seulement)

Fichier Éditer Rechercher Projet Options Outils Document Fenêtre Aide **1**

XPath 2.0 Exécuter XPath sur 'Fichier courant' **2**

Sommaire

Filtre de nom d'élément

recipe-collection

recipe "num1" Chocolate Co

3

recipes.xml

recipe-collection recipe

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <!DOCTYPE recipe-collection SYSTEM "rec.dtd">
3  <?xml-stylesheet type="text/css" href="rec.css"?>
4  <recipe-collection>
5      <recipe author="Dora" id="num1">
6          <title>Chocolate Cookies</title>
7          <ingreds>
8              <ing>1 cup butter</ing>
9              <ing>1 cup white sugar</ing>
10             <ing>1 cup packed brown sugar</ing>
11             <ing>2 eggs</ing>
12             <ing>2 teaspoons vanilla extract</ing>
13             <ing>3 cups all-purpose flour</ing>
14             <ing>1 teaspoon baking soda</ing>
15             <ing>2 teaspoon hot water</ing>
16             <ing>1/2 teaspoon salt</ing>
17             <ing>2 cups semisweet chocolate chips</ing>
18             <ing>1 cup chopped walnuts</ing>
19         </ingreds>
20         <steps>
21             <p>Preheat oven to 350 degrees F (175 degrees C).</p>
22             <p>Cream together the butter, white sugar, and brown sugar until smooth. Beat in the
23                 eggs one at a time, then stir in the vanilla. Dissolve baking soda in hot water. Add
24                 to batter along with salt. Stir in flour, chocolate chips, and nuts. Drop by large
25                 spoonfuls onto ungreased pans.</p>
26             <p>Bake for about 10 minutes in the preheated oven, or until edges are nicely browned.</p>
27         </steps>
28     </recipe>
29 </recipe-collection>
30
```

Texte Grille Auteur **6**

Attributs **4**

recipe

Attribut	Valeur
author	Dora
id	num1
rating	ok

Attributs Modèle

Scénarios de t...

Saisir le texte du filtre

Association Scénario

Projet (1)

Encodage

5

Sc.. Enti.. Él..

C:\...\Fonte Gaia Summer School 2017\XML\exercises\recipes.xml Associer une feuille de style XSLT/CSS - réussie U+000A 28 : 14 Jours restants pour évaluation: 30 10 nouveau(x) me...

Oxygen’s interface

1. Main menu:

- File: new, open, save as…
- Edit: copy, paste, check spelling, character map…
- Find: find/replace
- Project: Create/open a project (= set of files)
- Options: preferences, shortcuts, transformation scenarios…
- Tools: conversion, compilation, documentation…
- Document: management of the documents
- Windows: Display

2. Quick access toolbar:

- Traditional functionalities: New, open, save, find/replace…
- XML functionalities: transformation scenario, XSLT/CSS stylesheet…

Oxygen’s interface

3. Outline: Overview of the XML tree.

4. Attributes: List of the attributes available for one element.

5. Transformation scenario: mapping of the XML file in other formats.

6. Displays:

- Text mode: XML tree with tags;
- Grid mode: Table of tags;
- Author mode: Document layout (WYSIWYG editor).

Tips for Oxygen

- Suggestions of elements, when you start to write a tag.
- Tag autocompletion: Oxygen automatically closes the tags.
  - If you don’t like it : Option > Preferences > Editor > Content Completion > Deselect “Close the inserted element”.
- The Green/Red square indicates if your document has errors or not.
- **CTRL + E**: Allow you to tag a portion of text.

Introduction to the Text Encoding Initiative

TEI in Theory

TEI, what?

- TEI: an **XML language** for the digital representation of texts.
- Maintained by a **consortium**, which provides guidelines and tools.
  - TEI Consortium website: http://www.tei-c.org/index.xml
- More than 550 **markups**, grouped in 21 modules.
- **Modules**: Specifications of the TEI that provide a set of markups for the encoding of specific aspects of texts.
  Ex: Prose, verse, drama, dictionaries, figures…

TEI, what?

- An international consortium of institutions, projects and individual members
- A community of users and volunteers
- A freely available manual of set of regularly maintained and updated recommendations: 'The Guidelines' with definitions, examples, and discussion of over 560 markup distinctions
- A mechanism for producing customized schemas for validating your project's digital texts
- A set of free and openly licensed, customizable tools and stylesheets for transformations to many formats (e.g. HTML, Word, PDF, Databases, RDF/LinkedData, Slides, ePub, etc.)
- A simple consensus-based way of organizing and structuring textual (and other) resources
- An archival, well-understood, format for long-term preservation of digital data and metadata
- Whatever you make it! It is a community-driven standard

TEI, why?

- Objectives: **Structuring** and **description** of texts for their analysis, sharing, dissemination and publication.
- Compromise between traditional editorial requirements (**form**, structure) and researchers' needs (**content**, meaning).
- This implies different levels of encoding:
  - **Logical** encoding: representation the structure of the text;
  - **Semantics** encoding (names, places, dates…).

➔ The TEI is currently the only standard that allows you to encode the **meaning of texts** !

A Mental Exercise

Thinking about this material, and indeed your own, what do you think are the things you would like to mark up?

- Make a list of textual phenomena and metadata that are important to capture
- How likely is it that you can mark these up reliably and consistently?
- Could any of these potentially be marked up automatically by a cleverly crafted script?

TEI and schema

- To have a **valid TEI file**, you have to:
  - Respect the XML syntax;
  - Follow a schema;
  - Respect the semantic of the TEI elements.
- **Schema** = grammar and vocabulary for a specific XML language:
  - Elements, attributes and values that **can be used** in an XML file;
  - The **way you can use** these elements and attributes (mandatory, optional, repeatable…).

TEI in Practice: Text Body

MOVE YOUR BODY !

Basic structure of a TEI file

```
<TEI xmlns="http://www.tei-c.org/ns/1.0">
  <teiHeader>…</teiHeader>
  <text>
      <front>…</front>
      <body>
        <div>…</div>
      </body>
      <back>…</back>
  </text>
</TEI>
```

→ Metadata

→ Prefatory matter
(title page, preface, dedication, etc.)

→ Body of the text

→ Appendices
(table of content, index, etc.)

One text, several units

- TEI considers that a text is made of several units called **divisions**: books, parts, chapters, volumes, acts, scenes, poems…
- These divisions structure the text in logical units with ** **.
- Main attributes:
  - **@n**: number of a division;
  - **@type**: nature of a division.
- A division can be followed by a title, represented by **<head>**.
- Before encoding, it is important **to think about the structure** of your text in details : don't hesitate to represent it with a drawing, it will avoid you many inconveniences!

```
<div type="tragedy">
  <head>BERENICE</head>
  <div type="act" n="1">
      <head>Act I</head>
      <div type="scene" n="1">
          <head>Scene 1</head>
          <p>Text of the first scene.</p>
      </div>
      <div type="scene" n="2">
          <head>Scene 2</head>
          <p>Text of the second scene.</p>
      </div>
      …
  </div>
  <div type="act" n="2">
     <head>Act II</head>
     ....
  </div>
  …
</div>
```

Let’s try: Les Misérables

- Open **miserables.xml**.
- The text is green (= commented) : Remove the markups <!-- and --> at the beginning and at the end of the text.
  - Oxygen is red: don’t panic! It will become green tag after tag ;]
- From your observations of the book (**miserables.pdf**), structure the text with the tags :
  -
  - <head>
- Then, add the attributes : @n and @type.

Prose

- Page structure:
  - <pb/>: page break;
  - <fw>: running title, page number…
    - @type: header, pageNum, sig, catch…
    - @place: top-left, top-center, top-right, bottom-left, bottom-center, bottom-right.
- Text blocks:
  - <p>: paragraph;
  - <lb/>: line break;

Prose

- Italic and quotation:
  - <hi>: Any element different from the rest of the text (Generic)
    - @rend: italic, bold, center, uppercase, lowercase…
  - <said>: speech or thought by a real person (aloud or not, direct or indirect).
    - Ex: He said that <said aloud="true" direct="false">he loved apples.</said>
  - <quote>: passage made by an external agency, according to the narrator.
  - <emph>: passage emphasized for its linguistic or rhetorical aspects.
    - Ex: This is <emph>the</emph> book of the year!
  - <foreign>: passage in another language.
  - <title>
    - Ex: Racine is the author of <title>Berenice</title> and <title>Phèdre</title>.
  - <mentioned>: autonym.
    - Ex: <mentioned>Always</mentioned> always has an “s”.

Your turn !

- Structure the **miserable.xml** file with the previous elements and attributes.
- To know what tag to add, don’t forget to look at the PDF !
- If you want a preview of your encoding, remember the author mode.

To each text its markups

- The previous markups are just a little portion of the tags you can use to encode prose. According to your needs, you will use more markups.
- TEI enables **to represent the specificity of texts**. However, all text are not in prose! This is why there are markups that have been especially created for verse, drama, correspondence, dictionaries…
- Don't worry, the **TEI guidelines** are here for you.
- You can also find useful **tutorials** here.

TEI in practice: The TEI Header

"NO MATTER WHAT THE QUESTION IS, THE ANSWER IS METADATA" (KARA VAN MALSSEN)

The TEI Header

- Contains the metadata of your TEI file.
- **Metadata**: Digital data that describes and represents an other data (physical or digital).
- Metadata are **everywhere** and enable to:
  - **Describe** digital resources;
  - **Facilitate** the search of data;
  - **Manage** digital collections;
  - **Preserve** data.

The TEI Header

- The TEI Header is made of 4 main parts:
  - **<fileDesc>**: Description of the TEI file ➔ Mandatory.
  - **<encodingDesc>:** Description of the project and of editorial choices (corrections, specificities of the encoding) ➔ Optional.
  - **<profileDesc>**: Description of the production, the language, the subjects of the encoding ➔ Optional.
  - **<revisionDesc>**: History of the revisions ➔ Optional.

The TEI Header - FileDesc

- 3 main elements:
  - **<titleStmt>**:
    - **<title>**: Title of your XML file ➔ Mandatory.
    - **<respStmt>**: Names and roles of the persons who have participated in the encoding ➔ Optional.
      - <resp>: Role (OCR correction, transcription, encoding, revision…).
      - <name>
  - **<publicationStmt>**:
    - <authority>: The organisation(s) in charge of the project.
    - <address>: Organisation's address.

    OR
    - <p>: Unstructured description.

The TEI Header - FileDesc

```
<teiHeader>
    <fileDesc>
       <titleStmt>
          <title>Notre-Dame de Paris de Victor Hugo</title>
          <respStmt>
             <name>Elina Leblanc</name>
             <resp>Correction OCR</resp>
          </respStmt>
          <respStmt>
             <name>Elina Leblanc</name>
             <resp>Encodage</resp>
          </respStmt>
       </titleStmt>
       <publicationStmt>
          <authority>Université Grenoble-Alpes</authority>
          <address>
             <addrLine>621 Avenue Centrale</addrLine>
             <addrLine>38400 Saint-Martin-d'Hères</addrLine>
          </address>
          <availability>
             <p>Usage académique uniquement.</p>
          </availability>
       </publicationStmt>
```

The TEI Header - FileDesc

- **<sourceDesc>**: Description of the « physical » resource.
  - <p>: Unstructured.

  OR
  - <bibl>: Semi-structured.
    - <title>
    - <author>
    - <pubPlace>
    - <publisher>
    - <date>

  OR
  - <biblFull>: Detailed description.
    - <titleStmt>: Title and author of a physical resource.
      - <title>
      - <author>
    - <editionStmt>: Edition description
      - <edition>: Main features (First edition, new edition, special edition…)

The TEI Header - FileDesc

- <publicationStmt>: Information about the publication.
  - <pubPlace>
  - <publisher>
  - <date>
  - <idno>
- <noteStmt>: Other information.
  - <note>
  - <relatedItem>

- NB: There are specific markups for the description of manuscripts ➔ <msDesc> (also in the fileDesc).

- Try to fill the TEI header of **miserables.xml**.
  - You will find the metadata in **miserablesMetadata.txt**.

The TEI Header – FileDesc Example with <bibl>

```
<sourceDesc>
   <bibl>
      <title>Notre-Dame de Paris</title>
      <author>Victor Hugo</author>
      <publisher>Librarie de Louis Hachette et cie</publisher>
      <pubPlace>Paris</pubPlace>
      <date>1858</date>
   </bibl>
</sourceDesc>
```

The TEI Header – FileDesc Example with <biblFull>

```
<sourceDesc>
   <biblFull>
      <titleStmt>
         <title>Notre-Dame de Paris</title>
         <author>Victor Hugo</author>
      </titleStmt>
      <publicationStmt>
         <publisher>Librairie de Louis Hachette et Cie</publisher>
         <date>1858</date>
         <pubPlace>Paris</pubPlace>
         <idno>Bibliothèque nationale de France, Collection Hetzel 07 </idno>
      </publicationStmt>
      <notesStmt>
         <note>
            <p>Fac-similé numérisé par le BnF et disponile en ligne sur Gallica.</p>
            <p>Identifiant: ark:/12148/bpt6k406195r</p>
            <p>URL: http://gallica.bnf.fr/ark:/12148/bpt6k406195r/f3.item.r=Notre%20Dame%20de%20Paris%20Victor%20Hugo</p>
         </note>
      </notesStmt>
   </biblFull>
</sourceDesc>
```

Paragraphs

<p>

Fundamental unit for prose texts

<p> can contain all the phrase-level elements in the core

<p> can appear directly inside <body> or inside (divisions)

Quotation

Quotation marks can be used to set off text for many reasons, so the TEI has the following elements:

- <q> (separated from the surrounding text with quotation marks)
- <said> (speech or thought)
- <quote> (passage attributed to an external source)
- <cit> (groups a quotation and citation)

Highlighting

By highlighting we mean the use of any combination of typographic features (font, size, hue, etc.) in a printed or written text in order to distinguish some passage of a text from its surroundings. For words and phrases which are:

- distinct in some way (e.g. foreign, archaic, technical) emphatic or stressed when spoken
- not really part of the text (e.g. cross references, titles, headings)
- a distinct narrative stream (e.g. an internal monologue, commentary)
- attributed to some other agency inside or outside the text (e.g. direct speech, quotation)
- set apart in another way (e.g. proverbial phrases, words mentioned but not used)

Highlighting, examples

<hi> (general purpose highlighting); <distinct> (linguistically distinct)

Other similar elements include: <emph>, <mentioned>, <soCalled>, <term> and <gloss>

Simple linking examples

See <**ref** target="#Section12">section 12 on page 34</**ref**>.

See <**ptr** target="#Section12"/>.

Global Attributes

@rend: to describe a particular graphic feature of the source

@xml:id: to provide an element with a unique identifier. Starts with a letter

@facs: to connect a portion of text with an image

Encoding poems

Two key elements:

<lg> i.e. line groups to encode stanzas and the like

<l> i.e. line, for a verse

```
<lg>
        <l>One need not to be a chamber to be haunted</>
        <l rend="indent”>One need not to be a house</l>
</lg>
```

Drama

<sp> for a speech

<speaker> for the name of the speaker

<p> | <l> : if prose or verses

<stage> : stage direction

Drama

```
<div>
  <head>The Merchant of Venice</head>
  <byline>By William Shakespeare</byline>
  <div>
    <head> ACT I</head>
    <div>
      <head>SCENE I.</head>
      <stage type="location">Venice. A street.</stage>
      <stage type="entrance">Enter ANTONIO, SALARINO, and
SALANIO</stage>
      <sp  who="#ant">
        <speaker>ANTONIO</speaker>
        <l>In sooth, I know not why I am so sad: </l>
        <l>It wearies me; you say it wearies you; </l>
        <l>But how I caught it, found it, or came by it, </l>
        <l>What stuff 'tis made of, whereof it is born, </l>
                                <l>I am to learn; </l>
        <l>And such a want-wit sadness makes of me,</l>
        <l>That I have much ado to know myself.</l>
      </sp>
      <sp  who="#sal">
        <speaker>SALARINO</speaker>
        <l>Your mind is tossing on the ocean;</l>
      </sp>
    </div>
  </div>
</div>
```

Suggested values for @type in <stage>

setting: describes a setting.

entrance: describes an entrance.

exit: describes an exit.

business: describes stage business.

novelistic: is a narrative, motivating stage direction.

delivery: describes how a character speaks.

modifier: gives some detail about a character.

location: describes a location.

mixed: more than one of the above

Encoding letters

Several specialised elements

```
<div type="letter">
  <opener>
    <dateline>Grenoble, <date>23 November</date></dateline>
    <salute>Dear Friend</salute>
  </opener>
  <p>I hope you are doing all right. I am!</p>
  <closer>
    <salute>Have fun</salute>
    <signed>Your dull friend</signed>
  </closer>
</div>
```

Addresses

```
<address>
    <persName>Tom Willock</persName>
    <street>14 Frederick Street</street>
    <postCode>EH2 2HB</postCode>
    <settlement type="city">Edinburgh</settlement>
    <country>United Kingdom</country>

</address>
<email>tom.willock@email.com</email>
```

Notes

<note> (contains a note or annotation)
Notes can be those existing in the text, or provided by the editor of the electronic text

A @place attribute can be used to indicate the physical location of the note

Notes should usually be encoded where its identifier/mark first appears; notes can also be kept separately and point back to their location with a @target attribute

Figures

<graphic> (indicates the location of an inline graphic, illustration, or figure)

<binaryObject> (encoded binary data embedding a graphic or other object)

The figure module provides <figure> and <figDesc> for more complex graphics

```
<figure>
    <head>The View from the Bridge</head>
    <figDesc>A Whistleresque view showing four or five
sailing boats in the foreground, and a series of buoys
strung out between them.</figDesc>
    <graphic url="http://www.example.org/fig1.png"
     scale="0.5” height="234px" width="187px"/>

</figure>
```

Exercise

Create a new TEI file and encode the poem:

Machado.xml

Save your newly created TEI file within the same folder

Marking up people

The easy and the complex

- <name type="person">John Smith</name>

- <persName>
  <surname>Smith</surname>
  <forename>John</forename>
  </persName>

Referencing strings

Marking up allusions and implicit references:

« Mon Vieux »→ Louis Bouilhet

```
<rs type="person"
ref="https://en.wikipedia.org/wiki/Louis_Bouilhet">Louis Bouilhet</rs>
```

Referring names

In many case, the best way to disambiguate is to make a reference to external resources

- To internally defined lists
  - <persName key="jsmith">John</persName>
- To standard authority files
  - <persName ref="#n78085430">John</persName>

Describing people

<listPerson> → In the teiHeader (<particDesc>) or in a <p> in the main text

```
<person xml:id="CS">
        <persName xml:lang="de">
          <forename type="first">Clara</forename>
          <forename type="middle">Josephine</forename>
          <surname type="maiden">Wieck</surname>
          <surname type="married">Schumann</surname>
        </persName>
 </person>

</listPerson>
```

How many details?

http://www.tei-c.org/release/doc/tei-p5-doc/en/html/ND.html#NDPERSEpc

```
<person xml:id="cs">
        <persName xml:lang="de">
          <forename type="first">Clara</forename>
          <forename type="middle">Josephine</forename>
          <surname type="maiden">Wieck</surname>
          <surname type="married">Schumann</surname>
        </persName>
 <birth when="1876-09-12"><placeName>Tunbridge Wells</placeName></birth>
        <sex>Female</sex>
        <education>PhD <date when="1902"/><orgName>University of
Oxford</orgName></education>
        <langKnowledge>
          <langKnown tag="de">German</langKnown>
          <langKnown tag="en">English</langKnown>
        </langKnowledge>
        <event type="marriage" when="1895-11-24">
          <desc>On 24 November 1836, Clara married
            artist <persName>Ludwig Schumann</persName>. </desc>
        </event> </person>
```

Places

Many level of details, really

Simple:

- <name type="place">London</name>

Complex:

- <placeName>
- <geogName>

<placeName>

<district>

<settlement>

<region>

<country>

<bloc>

```
<placeName>
    <district type="arondissement">6ème</district>
    <settlement type="city">Paris, </settlement>
    <country>France</country>
</placeName>
```

Places

```
<place xml:id="BGbldg" type="building">
        <placeName>Brasserie Georges</placeName>
        <location>
          <country key="FR"/>
          <settlement type="city">Lyon</settlement>
          <district type="arrondissement">IIème</district>
          <district type="quartier">Perrache</district>
          <placeName type="street">Cours de Verdun</placeName>
        </location>
      </place>
```

```xml
<place xml:id="IS">
    <placeName xml:lang="en">Iceland</placeName>
    <placeName xml:lang="is">Ísland</placeName>
    <location>
      <geo>65.00 -18.00</geo>
    </location>
    <terrain>
      <desc>Area: 103,000 sq km</desc>
    </terrain>
    <state type="governance" notBefore="1944">
      <p>Constitutional republic</p>
    </state>

    <state type="governance" notAfter="1944">
          <p>Part of the kingdom of <placeName key="DK">Denmark</placeName></p>
    </state>
    <event type="governance" when="1944-06-17">
      <desc>Iceland became independent on 17 June 1944.</desc>
    </event>
    <state type="governance" from="1944-06-17">
      <p>An independent republic since June 1944</p>
    </state>
  </place>
```

Dates

```
<date>Last year</date>

<date when="2014">Last year</date>

<date notAfter="2014" notBefore="2014">last year</date>

<date from="2014-01" to="2014-12">last year</date>
```

Format: yyyy-mm-dd

Complex dates: non-Gregorian calendar

```
<calendarDesc>
      <calendar xml:id="Julian_England">
        <p>The Julian calendar, as used in late 16th-century England.</p>
       </calendar>
</calendarDesc>


<!-- in the text -->

<date calendar="#Julian_England" when-custom="1320-09-24">24th of Septemebr</date>


<date when-custom="1620-10-30"
            when="1620-11-09" calendar="#Julian_England"> 30. of October, 1620.
 </date>
```

Modelling with the TEI

Core elements

paragraphs

highlighting, emphasis and quotation

simple links and cross-references

lists, notes, annotation, indexing

More specific encoding

Genres: verse and drama

Adding meaning: names, events, people numbers, dates, addresses

Adding layout and decorations: graphics

Adding editorial care: bibliographic citations, corrections, critical apparatus

Specialised sets: dictionaries, manuscript description, linguistic annotations, editing the TEI

How to choose

500+ elements

Many modules and possibilities

A tool: Roma (don’t get too attached to it!)

The Guidelines: the BFG of the DH!