

# Manuel d'encodage XML-TEI

## Édition électronique de correspondance

Josselin Morvan - jeudi 18 juin 2015

## Première partie

# Principes généraux du langage XML

# Chapitre 1

## eXtensible(1) Markup Language(2)

1. X : *eXtensible*.
2. ML : *markup language* (Langage à balise)

### 1.1 XML comme langage de balisage

Les balises ont un rôle syntaxique, permettant de délimiter et de localiser une séquence dans une chaîne de caractères.

Exemple : Ici j'ai du texte en **gras** et ici en *italique*

— Principe :

Ici j'ai du texte en /débutGras **gras** /finGras et ici en /débutItalique *italique* /finItalique

— HTML :

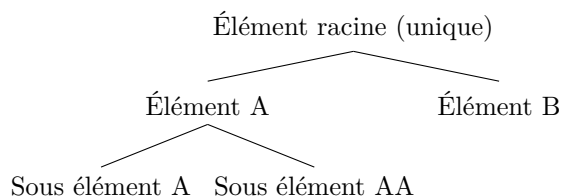
```
1 Ici aussi c'est en <b>gras</b> et là en <i>italique</i>.
```

XML permet de distinguer la définition sémantique des informations et leur mise en forme.

```
1 <!--Deux solutions pour un encodage XML:-->
2
3 <!--Prise en compte de l'apparence : a priori est en italique-->
4 On emploie <italique>a priori</italique> les italiques pour les locutions et termes é
  trangers
5
6
7 <!--Prise en compte du sémantisme de l'information : a priori est une locution
  étrangère-->
8 On emploie <locutionEtrangere>a priori</locutionEtrangere> les italiques pour les
  locutions et termes étrangers
```

## 1.2 Un langage à balise structuré

Un document XML respecte une structure hiérarchique formant une arborescence. Ce qui définit peut être le mieux cette structure ce sont les *Matriochkas*.



Cette structure implique que les éléments ne peuvent pas s'entrecroiser.

```

1 <!--Bien formé-->
2 <donateur><personne>Louise</personne> et <personne>Henri Leblanc</personne></donateur>
3 <!--Mal formé-->
4 <donateur><personne>Louise et <personne></personne>Henri Leblanc</donateur></personne>
  
```

## 1.3 Avantages et inconvénients d'un langage sémantique et structuré

### Avantages

- La structure sémantique et hiérarchique est compréhensible par les machines.
- Contenu  $\neq$  représentation. Cette dernière peut être gérée à part et est configurable à volonté.
- Utilisation d'un texte comme base de données.

### Inconvénients

- Structure parfois lourde

## 1.4 Un langage de balisage eXtensible ?

- XML est un métalangage : il permet de créer des langages avec leur propre grammaire (ex : EAD - TEI - XHTML).
- XML ne propose donc pas un jeu d'élément prédéfini, mais un ensemble de règles sur ce que doit être un document bien formé et valide.

## Chapitre 2

# Composition d'un document XML

- Les éléments : Structures de base d'un document XML. Ils sont délimités par des balises de début et des balises de fin. Tout ce qui se trouve entre ces deux balises représente le contenu

```
1 <personne>Henri IV</personne>
2 <!--La balise de début commence par un "<" et la balise de fin par "</", on
   trouve ensuite le nom de l'élément enfin, toutes deux se terminent par un
   ">"-->
3 <!--Le nom des éléments est sensible à la casse-->
4 <pb/>
5 <!--Il existe des éléments vides, c'est à dire sans contenu. Elle commence
   alors de la même manière par un "<" mais se termine par ">"-->
```

- Les attributs : Ce sont des paire "nom-valeur" associées à la balise de début d'un élément. Ils apportent généralement une information complémentaire pour l'encodeur ou sur le contenu de l'élément.

```
1 <element attribut="valeur">mon contenu</element>
2 <!--Le nom de l'attribut est séparé de la valeur par un signe "=" et la
   valeur doit être contenu dans entre des apostrophes ou des guillemets-->
3 <personne naissance="1553-12-13" mort="1610-05-14">Henri IV</personne>
4 <!--Un élément ne peut avoir plus d'un attribut avec un nom donné-->
5 <!--Il seront maintenant écrits sous la forme : attribut="valeur" ou
   @attribut-->
```

- Des entités : substituts pour des séquences d'information

```
1 <bonbon>tic &amp; tac</bonbon> <!--L'esperluette est un caractère réservé-->
```

- Des instructions de traitements : applications extérieures

```
1 <?xml-stylesheet type="text/css" href="style.css"?>
```

- Des commentaires

```
1 <!--Les commentaires sont des notes dans le code-->
```

- Des sections CDATA : Contenu textuel non parsé

```
1 <![CDATA[#Ici je peux mettre toutes les esperluettes que je veux &&&&]]>
```

- PCDATA : contenu textuel balisé.

Deuxième partie

**L'encodage en XML-TEI**

## Chapitre 3

# Introduction

L'aspect généraliste de la TEI a impliqué sa personnalisation à fin de l'adapter à notre projet d'édition de correspondance. Ce modèle d'édition, est basé sur les recommandations de la version P5 de la TEI.

Si dans l'ensemble la TEI offre les éléments nécessaires à l'encodage de correspondance, certains éléments ont tout de même nécessité une personnalisation plus poussée afin de répondre aux spécificités liées à l'encodage de correspondance, notamment dans le traitement des adresses. Plutôt que de créer un élément spécifique, dans la mesure où l'élément <address> est déjà existant, celui-ci peut maintenant être précisé avec les attributs @type et @place non prévus dans les recommandations de la *Text Encoding Initiative*.

## Chapitre 4

# Principe généraux



## Chapitre 5

# L'en-tête TEI : `<teiHeader>`

Tout texte encodé en TEI comporte un élément `<teiHeader>`<sup>1</sup>. Il s'agit de l'en-tête du document contenant les informations (métadonnées) permettant son identification. il se subdivise ainsi :

```
1 <teiHeader>
2   <fileDesc>(obligatoire) description bibliographique</fileDesc>
3   <encodingDesc> description des normes d'encodage</encodingDesc>
4   <profileDesc> description des aspects non bibliographiques</profileDesc>
5   <revisionDesc> historique des révisions du fichier numérique</revisionDesc>
6 </teiHeader>
```

Notons tout de même que tous les éléments du `<teiHeader>` n'ont pas vocation à être remplis. Nous nous bornerons donc à la description des éléments principaux présentant un intérêt général pour l'édition de correspondance. Pour plus d'information quant aux autres éléments disponibles, voir la documentation générale du projet contenant la description de l'ensemble des éléments retenue lors de la réalisation de ce modèle.

### 5.1 `<fileDesc>`

Cet élément de description bibliographique.

#### 5.1.1 `<titleStmt>`

Cet élément se réfère à la déclaration du titre, il est obligatoire. On y trouve les éléments suivants :

`<title>` : (répérable) Il désigne le/les titre(s) donnés à l'édition électronique.

`<respStmt>` : (statement of responsibility), cet élément se réfère aux mentions de responsabilité concernant l'édition électronique, les sous-éléments `<resp>` et `<name>` peuvent par exemple renvoyer vers la description du rôle d'un des responsable de l'édition et à son nom.

`<author>` : nom de l'auteur

`<editor>` : mention de responsabilité secondaire

exemple de `<titleStmt>`

---

1. Cet élément n'a pas été personnalisé dans la mesure où il s'agit essentiellement de métadonnées et que chaque projet d'édition peut avoir besoin de répondre à des besoins spécifique de ce point de vue.

```

1 <titleStmt xml:id="titleStmt">
2   <title n="1">Don de Madame <persName ref="#Comtesse_Forest">la Comtesse de la
      Forest</persName></title>
3   <title n="2">Souvenir de soldats</title>
4   <title n="3">Cartes postales illustrées</title>
5   <author>Armand Horel</author>
6   <respStmt>
7     <resp>Transcription et encodage XML/TEI : </resp>
8     <name xml:id="J_M">Josselin Morvan (étudiant à l'École des Chartes)</name>
9   </respStmt>
10  <respStmt>
11    <resp>Sous la direction de </resp>
12    <name xml:id="F_J_S">Frédérique Joannic-Seta (directrice adjointe de la
      <orgName>Bibliothèque de Documentation Internationale
      Contemporaine</orgName></name>
13  </respStmt>
14 </titleStmt>

```

### 5.1.2 <editionStmt>

la balise *edition statement* (optionnelle) renvoie aux mentions d'édition. le sous-éléments <edition> permet de décrire les spécificités de l'édition d'un texte.

### 5.1.3 <publicationStmt>

L'élément <publicationStmt> (obligatoire) regroupe les informations relatives à la publication et à la diffusion d'un texte.

<authority> : nom de la personne ou de l'organisme responsable de l'édition.

<address> : adresse du responsable de l'édition

<publisher> : responsable de publication.

<availability> : licence.

### 5.1.4 <noteStmt>

L'élément <noteStmt> (optionnel) permet d'ajouter des notes sur l'édition.

### 5.1.5 <sourceDesc>

<sourceDesc> (obligatoire), il consigne toutes les informations sur la source originale qui peuvent être détaillées à l'intérieur d'un simple balise <p>, ou bien très détaillé avec l'utilisation d'un élément <msDesc>.

<msDesc>

*manuscript description*, cette balise est à l'origine dédiée à la description des manuscrits anciens. Il est admis que son utilisation peut être étendue à d'autres types de texte. Cet élément peut contenir les éléments de description suivants :

<msIdentifier> : information d'identification de la source (lieu de conservation, identifiant...)

exemple de msIdentifier :

```

1 <msIdentifier xml:id="msIdentifier">
2   <country>France</country>
3   <settlement>Nanterre</settlement>
4   <repository>Bibliothèque de Documentation Internationale Contemporaine</repository>
5   <idno type="cote">F delta 1854/20</idno>
6 </msIdentifier>

```

<msContent> : cet élément est relatif au contenu intellectuel du texte édité, particulièrement intéressant dans le cadre d'un manuscrit, il peut aussi être décrit à l'aide d'un élément <p>

<physDesc> : Description physique du document, cette balise peut contenir deux sous-éléments principaux, <objectDesc> et <bindingDesc>, *a priori* seul le premier est intéressant dans le cas d'une édition de correspondance, car il est relatif au format et au support, le second est quant à lui attaché à la description des reliures.

<history> : Cet élément permet de retracer l'historique du document.

## 5.2 <encodingDesc>

Cet élément, relatif à l'encodage précise les principes éditoriaux qui ont permis la transcription du texte. il peut contenir :

<projectDesc> : description succincte du projet électronique.

<editorialDecl> : pratiques éditoriales de l'encodage. Cet élément peut contenir entre autres :

<correction> : Correction des erreurs détectées dans la source.

<hyphenation> : Procédure à suivre pour la coupure des mot en fin de ligne

<normalization> : Consigne de transcription, normalisation et régularisation du texte.

<ponctuation> : Règles d'encodage de la ponctuation

## 5.3 <profileDesc>

Cette partie concerne les informations non bibliographiques de description d'un document.

### 5.3.1 <langUsage>

l'élément <langUsage> associé à <language> précise la ou les langues utilisées. l'identification de la langue peut se faire à l'attribut @ident auquel est rattaché le code de la langue (format ISO 639-3)

```

1 <profileDesc>
2   <langUsage xml:id="langUsage">
3     <language ident="frm">Français</language>
4   </langUsage>
5 </profileDesc>

```

### 5.3.2 <correspDesc>

Cet élément contient la description des actions relatives à un acte de correspondance. Il s'agit d'un ajout récent de la TEI et est complété par les sous-éléments <correspAction> et <correspContext>

#### <correspAction>

Cet élément peut contenir le nom de l'expéditeur ou du destinataire (<persName> ou <orgName>), le lieu d'expédition ou de réception (<placeName>), et la mention de date (<date>) relatifs à un acte d'envoi ou de réception d'une correspondance. Il doit être précisé par l'attribut *@type* dont les valeurs sont les suivantes :

**forwarded** : informations concernant la transmission d'un message

**received** : informations concernant la réception d'un message

**redirected** : informations concernant la redirection d'un message non lu

**sent** : informations concernant l'envoi d'un message

**transmitted** : informations concernant la transmission d'un message, à savoir entre l'envoi et la réception suivante.

exemple de <correspAction>

```

1 <correspAction type="sent">
2   <persName ref="#Horel">Armand Horel</persName>
3   <placeName ref="#Port_Iero">Port Iero</placeName>
4   <date from="1915-10-09" to="1915-11-10"/>
5 </correspAction>
6 <correspAction type="received">
7   <persName ref="#Comtesse_Forest">Madame la Comtesse L. de la Forêt</persName>
8 </correspAction>

```

#### <correspContext>

Cette balise permet de replacer chronologiquement une correspondance dans un échange épistolaire et de lier les envois entre eux. La solution retenue est la suivante ; la correspondance suivante et la précédente seront identifiées par l'élément <ref> qui "définit une référence vers un autre emplacement". À la différence de l'élément <ptr> (pointeur), la référence peut être "complétée par un texte ou un commentaire". Dans le cas d'une correspondance, nous les complèterons avec les éléments <persName> et <placeName> précisés par les attributs *@type* de valeur "sentBy" ou "deliveredTo" afin d'identifier expéditeur et destinataire, et par l'élément <date>.

Exemple de <correspContext>

```

1 <correspContext>
2   <ref type="previous" target="../Moudros/F_delta_1854_20_1_Moudros_03-09-1915.xml">
3     <persName type="sentBy" ref="#Horel">Armand Horel</persName>
4     <persName type="deliveredTo" ref="#Comtesse_Forest">la Comtesse L. de la
5       Forêt</persName> :
6     <placeName type="sent" ref="#Moudros">Moudros</placeName>
7     <date type="sent" when="1915-09-03">03 septembre 1915</date>

```

```

7  </ref>
8  <ref type="next" target="../Mytilene/F_delta_1854_20_1_Mytilene_06-10-1915.xml">
9    <persName type="sentBy" ref="#Horel">Armand Horel</persName>
10   <persName type="deliveredTo" ref="#Comtesse_Forest">la Comtesse L. de la
      Forêt</persName> :
11   <placeName type="sent" ref="#Mytilene">Mythilene</placeName>
12   <date type="sent" when="1915-10-06">06 octobre 1915</date>
13 </ref>
14 </correspContext>

```

Dans le cadre de l'édition de la correspondance d'Armand Horel il nous est paru intéressant de développer cette re-contextualisation des lettres envoyées en fonction du lieu d'envoi.

Un deuxième élément <correspContext> peut donc être ajouter. il sera complété de la même manière que le précédent à la différence que les élément <ref> comporteront un attribut *@type* de valeur "context".

### <correspDesc> et <teiCorpus>

Si vous avez décidé d'unifier votre édition au sein d'un <teiCorpus>,

## Chapitre 6

# Le corps de la dépêche <text>

L'élément <text> contient l'ensemble du texte édité. Il peut être subdivisé en quatre sous-parties :

<front> : "(texte préliminaire) contient tout ce qui est au début du document, avant le corps du texte"

<group> : "(groupe) contient un ensemble de textes distincts (ou des groupes de textes de ce type), considérés comme formant une unité, par exemple pour présenter les œuvres complètes d'un auteur, une suite d'essais en prose, un groupe de correspondances liées par un lieu, une date etc."

<body> : "(corps du texte) contient la totalité du corps d'un seul texte simple, à l'exclusion de toute partie pré- ou post-liminaire."

<back> : "(texte annexe) contient tout supplément placé après la partie principale d'un texte : appendice, etc."

### 6.1 <front>

Cet élément est optionnel. Il peut contenir la page de titre, des dédicaces, une préface, l'introduction. La page de titre pourra être encodé avec l'élément <titlePage> ; pour le reste, aucune balise spécifique n'a été prévue l'introduction pourra donc être encodée de la manière suivante :

```
1 <front>
2   <div @type="introduction">
3     <p>
4       Ici le premier paragraphe de mon introduction
5     </p>
6     <p>
7       Ici le second
8     </p>
9     <p>
10      ...
11    </p>
12  </div>
13 </front>
```

## 6.2 <back>

L'élément <back> est dévolu aux textes annexes, il contient tout supplément placé après la partie principale d'un texte. Nous y placerons l'index et de glossaire.

## 6.3 Balisage du corps du document.

### 6.3.1 Introduction

L'encodage en XML-TEI ne permet pas un affichage direct en HTML, il faut pour cela passer par une feuille de style XSLT.

Le modèle d'encodage proposé par la BDIC a été développé de manière à respecter les sources qui feront l'objet d'une édition électronique. Ainsi, le texte pourra être encodé dans le respect de sa typographie puis normalisé par l'intervention de la TEI. c'est la feuille de style qui permettra ensuite de choisir une version de sortie en appliquant ou non certains choix éditoriaux.

### 6.3.2 Le Balisage physique

Cette section est relative à la mise en page du document originale. S'agissant de correspondances manuscrites, elles peuvent être rédigées sur plusieurs feuillets délimités par des sauts de page. De même au sein d'un page, les lignes de texte peuvent être identifiées par des passages à la ligne, il peut aussi exister des changement de colonne dans le cas des cartes postales par exemple, enfin le texte peut aussi être mis en valeur.

#### Le saut de page <pb/>

Le changement de feuillet ou le passage du recto au verso doit être encodé par l'utilisation de l'élément <pb/>. Il peut être complété par l'attribut *@n* afin de numéroté les pages. Dans ce cas il faut également penser à en placer un avant le premier feuillet afin que la numérotation soit complète. Dans le cas d'un corpus numérisé, les références aux fac-similés peut être faite par l'attribut *@facs*. Enfin si l'encodage porte sur des cartes postales, il est intéressant lorsque la correspondance tient sur plusieurs cartes de typer ce changement avec l'attribut *@type* en lui donnant la valeur "postcardBreak"

Exemple d'un d'utilisation d'un élément <pb/> dans le cadre d'une édition de cartes postales.

```

1 <pb n="3" facs="
2 http://argonnaute.u-paris10.fr/medias/customer_3/archives/lettres_soldats_jpg
3 /BDIC_FD_1854_20_1_179.jpg
4 http://argonnaute.u-paris10.fr/ark:/naan/86a661ceae
5 " type="postcardBreak"/>
6 <figure>
7 <graphic url="../../FACS/BDIC_FD_1854_20_1_179.jpg"/>
8 <figDesc>
9 <placeName ref="#Toulon">TOULON.</placeName> - Entrée de la nouvelle Caserne de
    Grignan
10 </figDesc>
11 </figure>
12 <pb n="4" facs="
13 http://argonnaute.u-paris10.fr/medias/customer_3/archives/lettres_soldats_jpg

```

```

14 /BDIC_FD_1854_20_1_180.jpg
15 http://argonnaute.u-paris10.fr/ark:/naan/86a661ceae"/>
16 <p>
17 <lb/><add type="numbering" n="3" rend="left"><pc>.</pc>3<pc>.</pc></add>
18 <lb/>J'ai reçu une lettre de mon <rs type="person" ref="#Horel_Louis"
    xml:id="Horel_Louis_23091915">frère</rs> datée

```

### Les sauts de ligne <lb/>

Par convention l'élément <lb/> marque le début d'une nouvelle ligne; il est donc préférable de la placer avant le texte. Les lignes peuvent être numérotées avec l'attribut @n.

### Coupure des mots

La coupure d'un mot sera aussi matérialisé avec l'élément <lb> mais cette fois-ci précisé par l'attribut @rend de valeur "hyphen." Afin de faciliter le développement d'un affichage ne tenant pas compte des changements de ligne nous les encoderons ainsi :

```

1 <lb/>je profite pour vous envoyer
2 quel<lb rend="hyphen"/>ques petites nouvelles de guerre<pc>.</pc> nous
3 <lb/>sommes appareillés pour l'Ile
4 <lb/><placeName ref="#Imbros">d'Imbros</placeName> qui se
5 trou-<lb rend="hyphen"/>ve juste à

```

### Les changements de colonnes <cb/>

L'élément <cb/> (saut de colonne) marque le début d'une nouvelle colonne de texte sur une page multi-colonne. Cette balise est particulièrement utilisée dans le cas de cartes postales. elle s'utilise de la même manière que les deux précédentes.

### La mise en valeur <hi>

L'élément <hi> (mis en évidence), "distingue un mot ou une expression comme graphiquement distincte du texte environnant, sans en donner la raison". Il doit être précisé par l'attribut @rend. Cette balise et son attribut n'interprète pas le texte contenu, ils précisent son apparence physique, dans le cadre d'une correspondance manuscrite, il peut s'agir d'une portion de texte soulignée, centrée, ou écrite en lettre capitale. Les valeurs de l'attribut @rend seront donc, "underline", "center", "sc". Ces informations sont ensuite traitées avec la feuille de style XSLT.

## 6.3.3 Segmentation ou balisage Logique

Le corps du document doit être balisé par l'élément <body>. Toutefois, chaque type de document (correspondance, poème, manuscrit ancien etc. dispose d'une structure qui lui est plus ou moins propre et qui peut être répartie dans des divisions logiques.

Une correspondance, réponds généralement d'une "structure diplomatique à la fois simple et flexible" : mention de date, apostrophe, paragraphes de texte, formule de courtoisie et signature. Afin d'uniformiser la sortie et de faciliter la lecture, il convient d'identifier ces différentes divisions.



**La division <div>**

Cette balise correspond à une subdivision du corps du texte édité. Cet élément doit être caractérisé par l'attribut *@type* afin d'identifier au mieux son contenu.

Dans le cadre de l'édition des correspondances nous avons retenu deux valeurs pour l'attribut *@type* qui sont "letter" et "enveloppe"

exemple :

```

1 <body>
2   <div type="letter">
3     Ici le corps de ma dépêche.
4   </div>
5   <div type="enveloppe">
6     Ici la transcription des mentions présentes sur l'enveloppe.
7   </div>
8 </body>

```

**Formule d'ouverture <opener>**

L'élément <opener> contient une formule d'ouverture. Pour une correspondance il s'agit généralement des mentions de date et lieu (<dateline> ainsi que de la formule de politesse ou apostrophe <salute>.

exemple de <opener>

```

1 <opener>
2   <dateline>
3     <placeName ref="#Imbros">Imbros</placeName>,
4     <date when="1915-09-23">le 23 7bre 1915</date>
5   </dateline>
6   <salute>
7     <persName ref="#Comtesse_Forest">Comtesse</persName>
8   </salute>
9 </opener>

```

**paragraphe <p>**

la balise <p> représente un paragraphe. C'est à l'intérieur de cet élément que l'on placera l'essentiel du texte de la correspondance. Chaque paragraphe dans le texte original doit être placé dans un élément <p>

Il peut être numéroté avec l'attribut *@n* et s'il présente un alinéa, il sera indiqué par l'attribut *@rend="indent"*.

Dans le cadre d'une correspondance tenant sur plusieurs cartes postales, et afin de rendre compte de cette matérialité, c'est le texte de chaque carte qui sera placé à l'intérieur d'un paragraphe.

Pour rappel, dans le cadre des cartes postale, l'élément <pb/> devra être précisé par l'attribut *@type="postcardBreak"*.

```

1 <p>
2   <lb/>Un bonjour de <placeName ref="#Port_Iero">Port
3   <lb/>Iéro</placeName> d'un petit ami qui
4   <lb/>ne vous oubliera jamais.
5 </p>

```

#### Formule de politesse/fermeture : <closer>

<closer> permet d'encoder les formules de fin d'une correspondance, de la même manière que pour l'<opener> on peut y trouver, une formule de courtoisie (<salute>), une signature (<signed>), mentions de date (<dateline>), une adresse (<address>) etc. notons que cet élément est répétable.

#### postscriptum : <postscript>

<postscript> trouve sa place soit avant soit après l'élément <closer>. Il peut reprendre une structure similaire au corps de la dépêche.

exemple de <postscript>

```

1 <postscript>
2   <opener>
3     <salute>Comtesse,</salute>
4   </opener>
5   <p>
6     je viens de recevoir votre carte
7     ...
8   </p>
9   <closer>
10    <signed rend="center">A. Horel</signed>
11  <closer>
12 </postscript>

```

### 6.3.4 Le balisage éditorial

Dans la mesure où l'édition porte sur des documents qui n'ont jamais été édités, ce balisage fait essentiellement référence aux interventions ajoutées dans le texte par les encodeurs afin d'en améliorer la compréhension et la lisibilité.

#### L'orthographe

Lors de la transcription du texte, l'encodeur dispose de plusieurs possibilités quant à la gestion d'une orthographe fautive, qui, faut-il le rappeler fait aussi partie de la condition de l'auteur. Le transcripteur peut tout d'abord décider de moderniser cette orthographe, "au fil de l'eau", lors de l'encodage du texte, perdant ainsi toute trace de la graphie originale. À l'inverse, il peut décider de ne rien corriger, ce qui peut poser problème lorsque l'orthographe est particulièrement déficiente. Enfin, la TEI permet, par un jeu de balise, de proposer une

lecture actualisée du texte tout en maintenant l'orthographe fautive ou ancienne de l'auteur <sup>1</sup>.

À cet effet, l'élément <choice> autorise de regrouper des balisages alternatifs (au moins deux) pour certains segments du texte, afin d'actualiser l'orthographe tout en gardant la trace de l'écrit.

**Les fautes d'orthographe** Il est possible d'affiner sa correction en fonction des cas de figure. Dans la mesure du possible, il est intéressant de distinguer les fautes d'orthographe récurrentes ou systématiques, de celles plus isolées.

Dans la première éventualité, il peut s'agir par exemple d'un auteur qui confond systématiquement l'infinitif des verbes du premier groupe et leur participe passé ou encore qui utilise le verbe avoir à la place la préposition "à". Il conviendra d'utiliser dans ce cas les éléments <orig>, pour la graphie originale, et <reg> pour la forme corrigée.

Concernant les fautes plus éparses, sur le même principe nous utiliserons alors les éléments <sic> pour l'orthographe fautive et <corr> pour la correction.

exemple :

```

1 <p>
2 tous ces jours ici il n'a fait que
3 <lb/><choice><orig>tombé</orig><reg>tomber</reg></choice> de la neige.
4 [...]
5 preuve que le
6 <lb/><choice><orig>barhomètre</orig><reg>baromètre</reg></choice>
7 est tombé à 3 degrés au dessous de zéro.
8 </p>
```

**Abréviations** Dans la mesure du possible, les abréviations doivent être développées ; à l'exception des plus courantes et des sigles. Il en va de même pour les noms de personnes si elles peuvent être identifiées.

Il convient d'utiliser la balise <choice> complétée par les éléments <abbr>, contenant l'abréviation, et <expan> pour la version développée. Il n'est en revanche pas nécessaire de distinguer les lettres ajoutées pour la résolution de l'abréviation.

**Les nombres** S'il est décidé de moderniser l'orthographe il convient de faire de même pour les nombres. Les chiffres doivent être écrits en toutes lettres à quelques exceptions :

**en chiffres arabes :** les dates, énumérations, degrés, numéros d'ordre etc.

**en chiffres romains :** les siècles, les numéros dynastiques ou les régimes politiques, les régions militaires etc.

Il convient d'utiliser les mêmes balises <reg> et <orig>.

---

1. Attention, ces opérations permettent d'améliorer la lisibilité, il ne s'agit en aucun cas de réécrire le texte.

**La ponctuation.** Lorsque la ponctuation gêne la compréhension parce qu'elle est fautive ou tout simplement absente il convient de la corriger. Voici comment procéder :

- Toute marque de ponctuation sera encodée avec l'élément <pc>.
- L'établissement d'une ponctuation inexistante devra être signifiée par l'attribut *@type*="supplied".
- Les éléments superflus seront aussi précisé par l'attribut *@type* mais de valeur "surplus"
- Enfin en cas de mauvaise utilisation d'une marque de ponctuation, il convient d'utiliser les éléments <reg> et <orig>.

**Rétablissement des Majuscules et minuscules** Le rétablissement de la ponctuation nécessite bien souvent de rétablir l'utilisation des majuscules et des minuscules lorsque cela est nécessaire. Il en est de même dans le texte, en cas de correction, il faut veiller à rétablir les majuscules pour les débuts de phrase et les noms propres et les minuscules pour les noms communs, à l'exception de noms abstraits ou lorsque l'auteur semble donner un sens ou une valeur particulière à un terme.

**L'État du texte** Il s'agit ici de rendre compte de l'état physique d'un texte, en signalant les parties, abîmées, incomplètes, illisibles, ajoutées etc.

**lecture incertaine :** il convient de signifier cette incertitude par l'élément <unclear> et dont la raison sera précisée dans la mesure du possible par l'attribut *@reason* (exemple de valeur, illisible (illisible), damage (dommage) etc.

**Les manques :** toute partie manquante doit être encodée par l'élément <gap> qui peut être complété par le même attribut.

**Les suppressions :** l'élément <del> identifie les parties qui ont été supprimées volontairement soit par l'auteur (une rature par exemple), ou par un tiers. Il peut être précisé avec l'attribut *@rend* et la valeur "overline" lorsqu'il est simplement barré ou "censored" lorsque le texte a été censuré. L'attribut *@hand* permet d'identifier la main responsable de cette suppression lorsqu'il ne s'agit pas de celle de l'auteur et qu'elle peut être identifiée.

**les ajouts :** Les ajouts de l'auteur ou d'une autre personne seront encodés à la place où elles se situent dans le texte avec la balise <add>. Elle sera précisée par les attributs *@place*, *@rend* et *@hand*. Lorsqu'il s'agit d'une signature ou d'une formule de politesse ajoutée en marge par manque de place, il convient de distinguer plusieurs cas de figure :

- L'ajout est par exemple en marge mais à la fin du texte, il faut alors utiliser la balise <closer> complétée de la balise <add>

```

1 <closer>
2   <salute>
3     <add place="margin-right" rend="vertical">
4       un petit ami qui vous envoie ses
5     </add>
6     <add place="margin-left" rend="vertical">
7       meilleurs souvenirs de <placeName
8         ref="#Salonique">Salonique</placeName>&vir;
9     </add>
10  </salute>
11 </closer>
```

- la signature sur fait sur un précédent feuillet, il faut alors attribuer à l'élément <add> l'attribut @type avec les valeur "closer" ou "signed"

```

1 <add type="closer">
2   <add type="signed" place="margin-right" rend="vertical">
3     <persName ref="#Horel" xml:id="Horel_27041916">Armand
        Horel</persName>
4   </add>
5   <add place="margin-left" rend="vertical">
6     Bonne santé &a; tous et bonne<lb/> chance
7   </add>
8 </add>

```

**La restitution** Lorsqu'il apparait d'un mot a été accidentellement oublié par l'auteur celui-ci peut être restitué avec l'élément <supplied> et la raison identifiée par l'attribut @reason="omitted"

**Les notes** Les notes que l'on souhaite ajouter dans le texte (ou bien pour respecter l'écrit de l'auteur) doivent être encodées ainsi :

- Au sein de l'élément <back>, il convient de créer une division <div> comportant l'attribut @type="note". Nous y placerons autant d'élément <note> que nécessaire. Ils devront comporter un @xml:id<sup>2</sup> et l'attribut de numérotation @n.
- dans le text : à la place ou doit apparaitre la note nous utiliserons un élément <ref> de @type="note" faisant et lié à son élément <note> par l'attribut @target. Cette balise ne doit pas comporter de contenu. l'attribut @n doit être de la même valeur que celui de la note.

exemple :

```

1 <body>
2   <p>
3     Comtesse<ref type="note" target="#12081915_note1" n="1"/>,
4     <lb/>je vous souhaite [...]
5   </p>
6 </body>
7 <back>
8   <div type="note">
9     <note xml:id:"12081915_note1" n="1">
10      Il s'agit de Claude Silve.
11    </note>
12  </div>
13 </back>

```

2. Attention si le fonds doit être réunie dans un <teiCorpus>, il convient de développer des identifiants uniques pour chaque lettres, il peut par exemple s'agir de la date d'envoi suivie d'un texte générique type "\_\_note" de manière mieux les identifier.

## 6.4 La balisage des noms et dates

Ce balisage permet d'identifier les noms propres et les dates afin de pouvoir les réutiliser par la suite dans un index ou un glossaire.

### 6.4.1 Noms de personnes : <persNames>

Cet élément "contient un nom propre ou une expression nominale se référant à une personne, pouvant inclure tout ou partie de ses prénoms, noms de famille, titres honorifiques, noms ajoutés, etc.". Afin de le lier à l'index ou au glossaire, cet élément doit comporter l'attribut *@ref* renvoyant à l'*@xml:id* de l'entrée concernée ainsi que, pour la première occurrence de la lettre, un *@xml:id* afin de pouvoir le localiser lors de la sortie HTML.

exemple :

```
1 <persName ref="#Horel" xml:id="Horel_a27-10_b3-11">Armand Horel</persName>
```

### 6.4.2 Références nominales

L'élément <persName> doit être utilisé lorsque le nom de la personne est clairement identifié. Autrement il convient de recourir à l'élément <rs> plus générique, il doit être précisé avec l'attribut *@type="person"* et lié à l'index de la même manière.

```
1 <rs type="person" ref="#Horel_Louis" xml:id="Horel_Louis__09101915">mon  
malheureux frère</rs>
```

### 6.4.3 Noms de lieux : <placeName>

Cet élément s'utilise de la même manière que le précédent.

exemple :

```
1 <placeName ref="#Port_Iero" xml:id="Port_Iero_a27-10_b3-11">Port  
2 Iéro</placeName>
```

### 6.4.4 Autres noms propres : <name>

L'élément <name> est plus générique que les deux balises précédentes. Il permettra d'identifier tous les autres noms propres qui ne correspondent pas à une personne ou un lieu. Il sera en plus précisé par un attribut *@type*.

exemple :

```
1 <name type="navire" ref="#Shamrock_II"  
xml:id="Shamrock_II_27041916">Shamrock II</name>
```

### 6.4.5 Mentions de dates : <date>

l'élément <date> permet d'encoder les mentions de date de n'importe quelle forme. Des attribut permettent de préciser et d'uniformiser la valeur.

*@when* permet de standardiser la date sous la forme AAAA, ou si la date est plus précise, AAAA-MM-JJ.

*@notBefore/@notAfter* sont utilisés lorsque la date est imprécise.

*@cert* permet de préciser le degré de certitude ; les valeurs sont "high" (forte), "medium" (moyenne) et "low" (basse).

## 6.5 L'encodage des figures : <figure>

Lorsqu'une illustration est présente il est intéressant de le signaler par l'utilisation de l'élément <figure>. Lorsqu'un facsimilé numérique est disponible, l'élément <graphic> aidé par l'attribut *@url* permet de lier l'image au texte. Enfin l'élément <figDesc> permet de donner une description de cette image.

exemple :

```
1 <figure>
2   <graphic url="../../../FACS/BDIC_FD_1854_20_1_173.jpg"/>
3   <figDesc>
4     Village Molivo
5   </figDesc>
6 </figure>
```

Dans le cas de cartes postales la face illustrée sera encodée avec cet élément.

exemple :

```
1 <pb n="3" type="postcardBreak"/>
2 <figure>
3   <graphic url="../../../FACS/BDIC_FD_1854_20_1_173.jpg"/>
4   <figDesc>
5     Village Molivo
6   </figDesc>
7 </figure>
8 <pb n="4"/>
```

# Chapitre 7

## Index et Glossaire

De manière à ne pas surcharger l'apparat critique, les notes de bas de pages peuvent être limitées par la constitution d'un index-glossaire. Toutes les références nominales (ou tout du moins la première occurrence de chaque lettre) doivent être reliées à l'index afin d'être identifiées. Trois index seront réalisés si besoin ; un index *nominum* (<listPerson>), un index *locorum*, (<listPlace>), et un glossaire (<list> @type="glossaire"). Chaque sous-élément de ces listes (<person>, <place>, <item>) devront être identifiés par un @xml:id, afin de s'y référer lors de l'encodage du texte.

### 7.1 Index nominum : <listPerson>

l'élément **listPerson** permet la constitution d'un index nominum. Il est composé d'un ensemble de sous-éléments <person>, complété à son tour par des éléments <persName>, <birth> et <death> pour les date de naissance et de mort<sup>1</sup>, <event><sup>2</sup> ou <state><sup>3</sup> et enfin un élément <note> permettant de rassembler les sous-éléments <ref> afin de lier, à son tour l'index à chaque occurrence dans les différentes lettre.

exemple :

```
1 <person xml:id="Kitchener">
2   <persName type="normal">
3     <forename>Horatio</forename> <!--prénom-->
4     <surname>Kitchener</surname> <!--nom de famille-->
5     <roleName type="nobility">Lord</roleName> <!--Composante d'un nom
        indiquant son rôle dans le société-->
6     <roleName type="military">Field Marshal</roleName>
7   </persName>
8   <birth>1850</birth>
9   <death>1916</death>
10  <event>
11    <p>
12      Ministre de la guerre en août 1914 [...]
```

1. Ces deux éléments s'utilisent de la même manière que l'élément <date>.

2. contient des données liées à tout type d'évènement significatif dans l'existence d'une personne, d'un lieu ou d'une organisation.

3. description d'un statut ou d'une qualité actuels attribués à une personne, un lieu ou une organisation.



```

13     </p>
14 </event>
15 <note>
16   <ref target="../../Salonique/Salonique_23-11-1915.xml#Kitchener_23111915">
17     <placeName ref="#Salonique">Salonique</placeName>
18     <date when="1915-11-23"/>
19   </ref>
20 </note>
21 </person>

```

L'index-glossaire est particulièrement complexe dans sa réalisation dans la mesure où certaine entrée ne doivent pas se faire au nom de famille mais au prénom (les souverains par exemple). Il est alors nécessaire de typer l'élément `<persName>` avec l'attribut `@type` et de lui donner les valeurs "normal" pour le cas le plus courant et "nobility" dans le cas d'une indexation au prénom.

Notons que l'attribut `@target` de l'élément `<ref>` renvoie à une occurrence située dans un autre fichier.

## 7.2 Index Locorum : `<listPlace>`

`<listPlace>` doit être complété sur le même principe que l'élément `<listPerson>`. Toutefois il faut faire attention aux spécificités régionales et typer les noms de lieux en conséquence. Les éléments `<state>` et `<event>` peuvent aussi être employés.

```

1 <place xml:id="Zante">
2   <placeName type="ville_gr">Zante</placeName>
3   <location>
4     <country>Grèce</country><!--Pour les pays étrangers-->
5     <district type="peripherie">Îles Ioniennes</district>
6     <district type="district">Zante</district>
7     <district type="deme">Zante</district>
8   </location>
9   <note>
10    <ref target="../../Salonique_25-01-1916.xml#Zante_25011916">
11      <placeName ref="#Salonique">Salonique</placeName>
12      <date when="1916-01-25"/>
13    </ref>
14  </note>

```

## 7.3 Glossaire : `<list>`

L'élément liste permet de réaliser des entrées d'index ou de glossaire à partir de termes qui ne peuvent être placés dans les `<listPerson>` et `<listPlace>`. La même principe a encore une fois été retenu.

```

1 <item xml:id="Lutetia">

```

```

2   <label>
3     <name type="navire">Lutétia</name>
4     <rs>le</rs>
5   </label>
6   <desc>
7     Croiseur auxilliaire <date from="1915" to="1917"/>
8   </desc>
9   <listEvent>
10    <event>
11      <p>
12        <date notAfter="1915">Avant 1915</date>
13        Paquebot sud Atlantique
14      </p>
15    </event>
16    <event>
17      <p>
18        <date when="1915-03-19">1915 (mars)</date>
19        Réquisitionné
20      </p>
21    </event>
22  </listEvent>
23  <note>
24    <ref target="../../Salonique/Salonique_19-01-1916.xml#Lutetia_19011916">
25      <placeName ref="#Salonique">Salonique</placeName>
26      <date when="1916-01-19"/>
27    </ref>
28  </note>
29 </item>

```