Edition de texte numérique: les propositions de la Text Encoding Initiative

http://lb42.github.io/2017-09-ens/

Sommaire

2 heures ne suffisant guère pour vous présenter les merveilles de la TEI, j'essaie simplement de vous donner un petit goût de quelques aspects signifiants, avec une emphase pratique...

- La théorie sousjacente: qu'est ce qu'on fait en numérisant un texte ?
- Encodage et édition : deux facons d'enregistrer une lecture
- Balisage XML; conceptes de validation et de schéma
- *Pratique de balisage avec un editeur XML*
- Architecture de la TEI: modules, concept de ODD
- *Exploration de la TEI avec Roma; création d'un schéma personnalisé, par ex pour transcription d'un doc archival*
- Traitement des documents TEI XML : concepts CSS, XSLT, Xquery
- Variétés d'outil
- Le communauté TEI actuel: comment s'y prendre

En attendant

Installez le logiciel oXygen XML editor ...

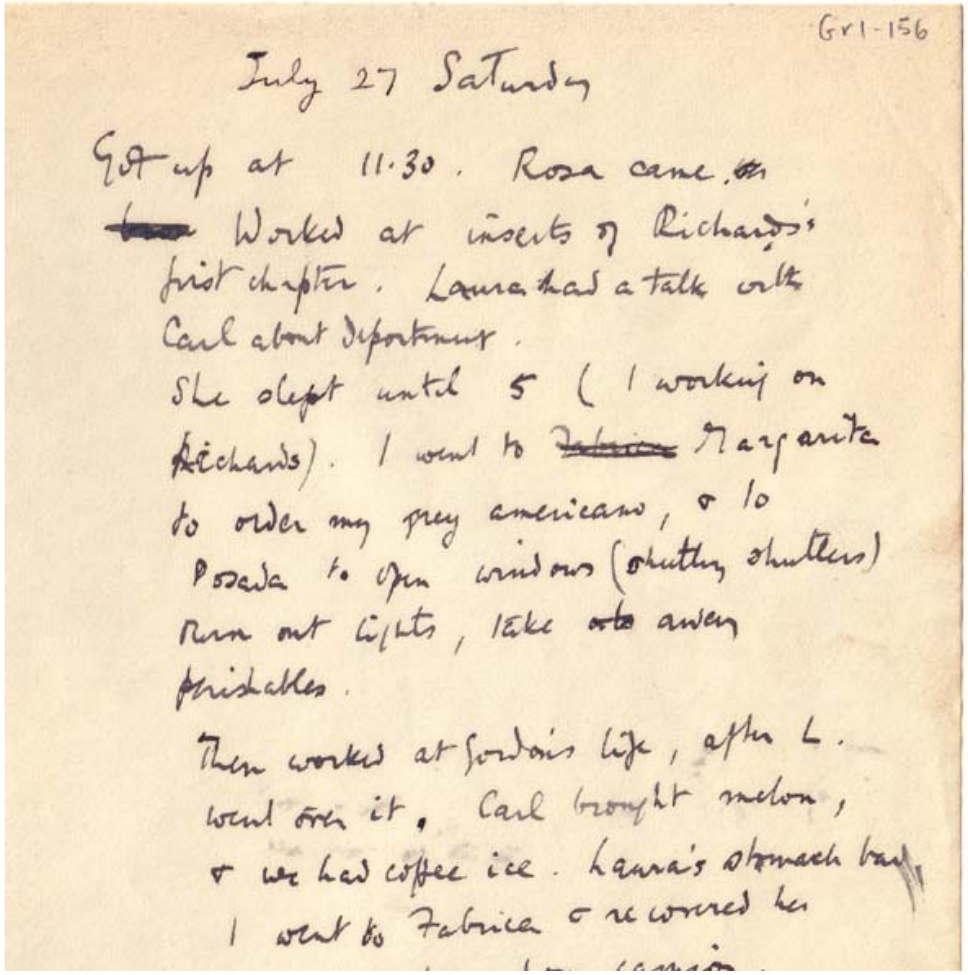
- visitez https://www.oxygenxml.com/download.html
- Selectionnez XML Editor et cliquez Download
- Selectionnez votre plateforme (Windows, Mac, Linux...) et cliquez Download
- Completez le formulaire en fournissant votre adresse email, votre pays, votre desir de pub, et le code secret en bas
- Enregistrez l'installeur et l'activez
- Controllez votre email pour le licence d'essai: vous en aurez besoin la premiere fois que vous activez oXygen

Texts 'r us

Quoique votre définition des 'humanités numériques', je suis certain qu'elle implique des considérations du *texte* et des textes numerisées...

'texte' dans le sens le plus complet: pas simplement des facsimilés numériques mais également leur transcriptions ; pas simplement des transcriptions, mais également des annotations et interprétations la dessus ; pas simplement des annotations, mais également des metadonnées ...

Par exemple ...



GvI-156

July 27 Saturday

Got up at 11.30. Rosa came. ~~to~~
~~[redacted]~~ Worked at inserts of Richards's
first chapter. Laura had a talk with
Carl about departure.
She slept until 5 (I working on
Richards). I went to ~~Fabrica~~ Margarita
to order my grey americana, & to
Rosa to open windows (shutting shutters)
turn out lights, take ~~out~~ away
perishables.
Then worked at Sondais life, after L.
went over it, Carl brought melon,
& we had coffee ice. Laura's stomach bad.
I went to Fabrica & recovered her

un ‘substitut’ (surrogate) représentant l’apparence d’un document existant

Exemple continue ...

Diary of Robert Graves 1935-39 and ancillary material

New Search | Diary Scans

« Return to Search Results

July 1935

« June | « Abstract » | August »

Sun	Mon	Tue	Wed	Thu	Fri	Sat
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

DISPLAYED DIARY SCAN(S)

July 27 Saturday

» Annotated markup

» Full-sized Image

» Gallery Scan

July 27 Saturday

Got up at 11.30. **Rosa** came.~~xxxxx~~[crossed out]

~~xxxxx~~[crossed out] Worked at inserts of **Richards**' first chapter. **Laura** had a talk with **Carl** about deportment.

She slept until 5 (I working on **Richards**[1]). I went to ~~Fábrica~~[RG] **Margarita** to order my grey *americano* [2], & to **Posada** to open windows (shutting shutters) turn out lights, take ~~into~~[RG] away perishables.

Then worked at **Gordon**'s Life[3], after **L.** went over it. **Carl** brought melon, & we had coffee ice. **Laura**'s stomach bad. I went to **Fábrica** & recovered her parasol & fan from *camión* [4]. More work on **Gordon**[5]. Bed at 12.

Gelat expects no result of law suit for two months. **Concordia** ceiling finished: tiles green & yellow, being laid diagonally.

EDITORIAL NOTES

[1] **Old Soldier Sahib**. eds.
[2] Spanish [slang?] for "jacket" KG; KG also replaces the final "o" with "a". eds.
[3] See **A Mistake Somewhere**. eds
[4] bus. KG
[5] i.e. Gordon's autobiography: see above. eds.

une représentation du contenu linguistique, et structure, avec annotations sur sa portée, son contexte..

.. et en dessous

```
−<div type="diaryentry" n="1935-07-27">
  <head> July 27 Saturday </head>
  −<p>
    Got up at 11.30.
    <rs type="person" key="Ro1">Rosa</rs>
    came.
    <unclear reason="crossed out"/>
  </p>
  −<p>
    <unclear reason="crossed out"/>
    Worked at inserts of
    <rs type="person" key="FR2">Richards</rs>
    ' first chapter.
    <rs type="person" key="LR1">Laura</rs>
    had a talk with
    <rs type="person" key="KG1">Carl</rs>
    about deportment.
  </p>
  −<p>
    She slept until 5 (I working on
    <rs type="person" key="FR2">Richards</rs>
    −<note>
      −<bibl>
        −<rs type="cita" key="OSS">
```

La tournée numérique

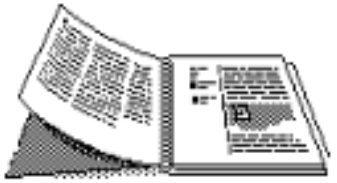
Les sciences humaines et sociales s'occupent surtout du *texte* ...

- (majoritairement non-numérique) les livres, les manuscrits, les fonds d'archive ...
- ainsi que d'autres manifestations culturelles/communicatives (de plus en plus numerisées) par ex., les sons, les images, les cahiers de recherches, les tweets

Les 'digital humanities' se préoccupent avec les outils et les techniques qui permettent de manipuler d'une manière integrée toutes ces manifestations, et donc de gérer ce patrimoine de plus en plus signifiant.

Le balisage (markup, encodage) est une composante incontournable de ces manipulations

Traîtements numériques du texte



- word processing
- indexing
- database

aardvark
abacus
abracadabra
acidulation
adumbrate
aetherial
affection
agony
aha
ai
ajacanthus

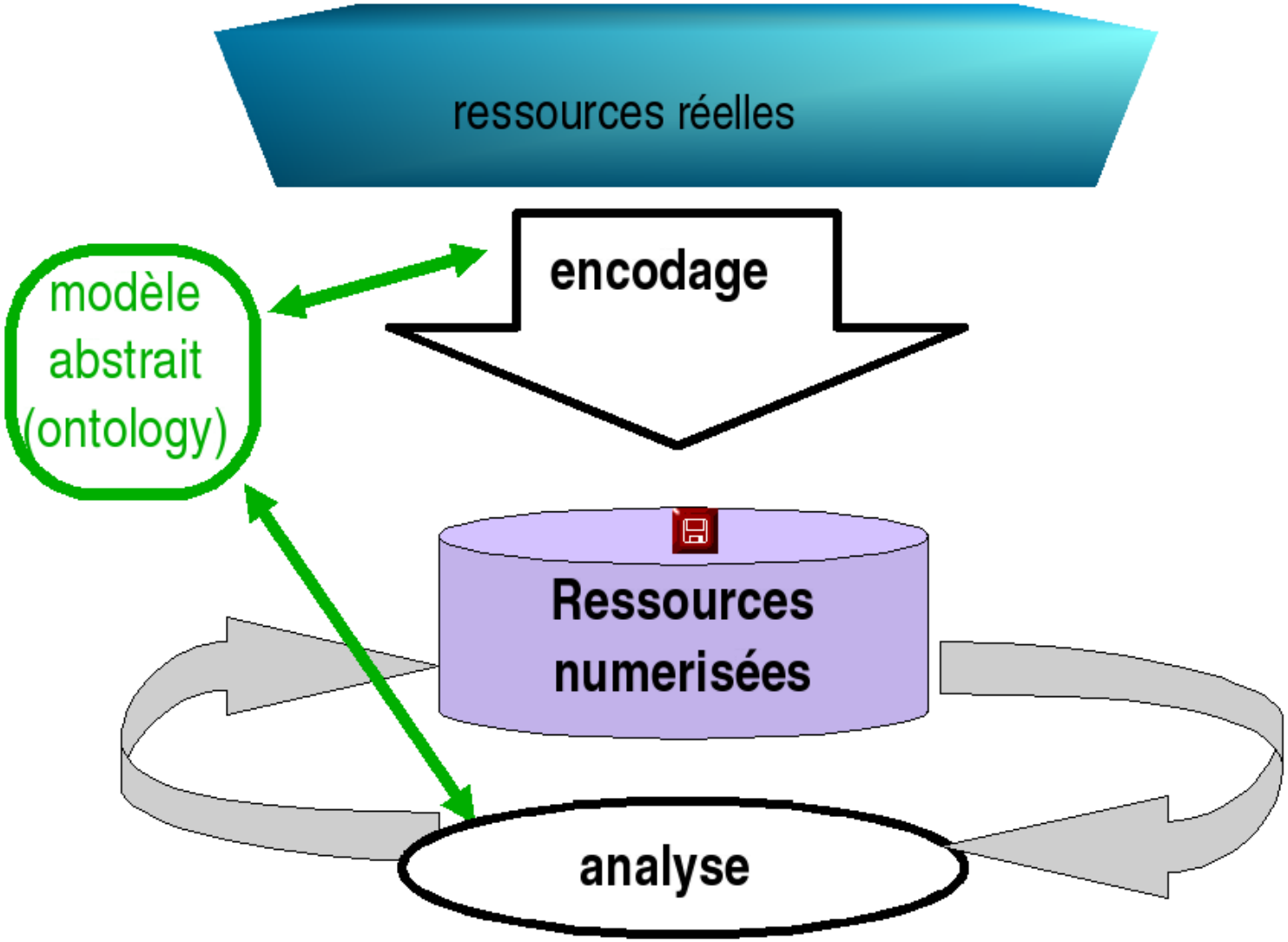
Texte et texte numérique

Un texte peut être considéré selon trois axes :

- Un texte a une existence physique, ayant des traits visuels qu'on peut (plus ou moins) transférer automatiquement d'une instance à une autre
- Un texte possède des propriétés linguistiques et structurelles, qu'on ne peut transcrire, traduire, ou transmettre qu'avec une compréhension humaine
- Un texte présente des informations sur le monde réel, qu'on peut comprendre (ou non) ou annoter, et qui nous permet de générer de nouveaux textes

Un balisage effectif devrait donc opérer dans tous ces trois axes.

Qu'est-ce qu'on fait en numérisant un texte?



Ceci n'est pas un arbre

Un modèle textuel réduit les complexités des textes, en utilisant un syntaxe simple pour les exprimer

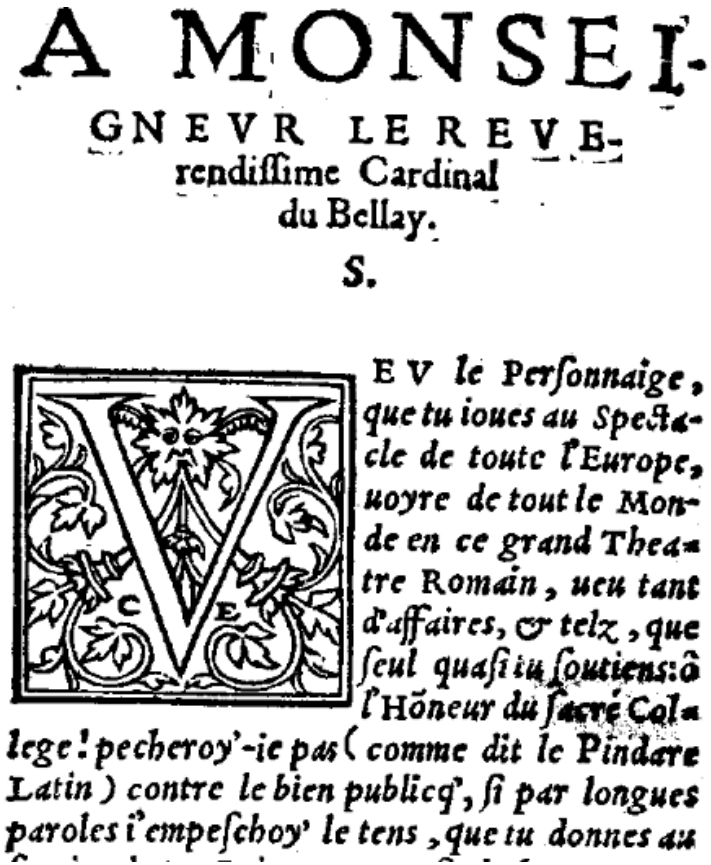
XML, par exemple

```
texte texte <tag attribut="valeur">element</tag> texte
```

On introduit des balises dans le flux de texte

- pour identifier un empan de ce flux
- pour lui associer un nom ou type ...
- ... et peut être des attributs

Par exemple



A MONSEI-
GNEVR LE REVE-
rendiſſime Cardinal
du Bellay.
S.

VEV le Perſonnaige, que tu ioues au Spectacle de toute l'Europe, uoyre de tout le Monde en ce grand Theatre Romain, ueu tant d'affaires, & telz, que ſeul quaſi iu ſoutiens: ô l'Hõneur du ſacré College! pecheroy'-ie pas (comme dit le Pindare Latin) contre le bien publicq', ſi par longues paroles i'empeſchoy' le tens, que tu donnes au

Qu'est ce qu'on balisera?

Balisage du mise en evidence


A MONSEI-
GNEVR LE REVE-
rendiſſime Cardinal
du Bellay.
S.

EV le Perſonnaige,
que tu ioues au Specta-
cle de toute l'Europe,

```
<pb n="4"/>A MONSEI-
<lb/>GNEUR LE REVE-
<lb/>rendissime Cardinal
<lb/>du Bellay.
<lb/>S

<lb/>
<c rend="lettrine">V</c>EV le
Personnaige,
<lb/>que tu ioues au Specta-
<lb/>cle de toute l'Europe...
```

Balisage de structure

A MONSEI-

GNEVR LE REVE-
rendiſſime Cardinal
du Bellay.

S.

EV le Perſonnaige,
que tu ioues au Specta-
cle de toute l'Europe,

```
<div type="dedicace">
 <head>A MONSEIGNEUR LE
REVERENDISSIME CARDINAL DU
BELLAY</head>
 <salute>S<ex>ire</ex>
 </salute>
 <p>
  <c rend="lettrine">V</c>EU
le Personnaige, que tu joues
au Spectacle de toute
   l'Europe... </p>...
</div>
```

Mais on peut aller plus loin...

```
<pb n="4"/>
<s>
 <w pos="PPJ" lemma="voir">VEU</w>
 <w pos="ART" lemma="le">le</w>
 <w pos="SBC" lemma="personnage">Personnaige</w>
 <pc>,</pc>
 <w pos="COO" lemma="que">que</w> ...
</s>
```

ou bien

```
<s>
 <choice>
  <reg>Vu</reg>
  <orig>Veu</orig>
 </choice> le
<choice>
  <reg>Personnage</reg>
  <orig>Personnaige</orig>
 </choice>, que tu joues
au Spectacle...
</s>
```

à ne rien dire de ...

```
<head> A MONSEIGNEUR LE REVERENDISSIME
<persName ref="#dubellay03">CARDINAL DU
   BELLAY</persName>
</head>
<!-- .... -->
<person xml:id="dubellay03">
 <persName>Jean du Bellay</persName>
 <birth>
  <date>1492</date>
  <placeName>Souday</placeName>
 </birth>
 <death>
  <date when="15600216">16 February 1560</date>
  <placeName>Roma</placeName>
 </death>
<!-- ... -->
</person>
```

Conclusions preliminaires

- Avant de commencer un exercice de balisage, il faut bien préciser son choix des balises
- Ce choix sera déterminé par les distinctions et méta-informations qu'on considère d'importance
- L'XML nous aide en définissant un syntaxe formel pour notre balisage
- La TEI nous aide en fournissant un lexique tres complet des balises disponibles

Revenons d'abord sur le syntaxe XML

La bonne soupe d'acronymes

SGML	Standard Generalized Markup Language
HTML	Hypertext Markup Language
W3C	World Wide Web Consortium
XML	eXtensible Markup Language
DTD	Document Type Definition (or Declaration)
CSS	Cascading Style Sheet
Xpath	XML Path Language
XSLT	eXtensible Stylesheet Language - Transformations
RelaxNG	Regular Expression Language for XML (New Generation)

à ne pas oublier TEI, la *Text Encoding Initiative*

XML: ce que c'est et pourquoi on devrait le connaitre

- XML est une manière de représenter les données structurées sous forme de chaîne de caractères
- XML est extensible
- un document XML doit être *bien formé*
- un document XML peut être *valide*
- XML est indépendant de l'application, de la plateforme et du vendeur
- XML rend le pouvoir aux fournisseurs de données, et facilite l'intégration des ressources diverses et polyglottes

(Presque) tout ce qu'il faut savoir au sujet de l'XML, sur un seul transparent

- Un document XML contient au moins un *élément*
- Un élément possède une *balise d'ouverture*, facultativement de *contenu* et une *balise de fermeture*
- Un élément peut d'ailleurs porter des *attributs*, chacun portant un *nom* et une *valeur*
- Un document XML est *obligatoirement* 'well formed' (bien-formé) i.e. il doit suivre la syntaxe XML
- Un document bien-formé peut *facultativement* être *valide* i.e. il est conforme aux règles d'un *schéma* quelconque

```
<?xml version="1.0" ?>
<root>
  <element attribute="value"> content </element>
  <!-- comment -->
</root>
```

Un petit document XML

```xml
<?xml version="1.0" encoding="UTF-8"?>
<cookBook>
    <recipe n="1">
        <head>Soupe de pierre</head>
        <ingredientList>
            <ingredient>un oignon</ingredient>
            <ingredient>deux carottes</ingredient>
            <ingredient>de l'eau</ingredient>
               <!-- d'autres ingrédients -->
            <ingredient>une pierre</ingredient>
            <ingredient>des paysans naïfs</ingredient>
        </ingredientList>
        <procedure>
            <step>mettre l'eau à bouillir dans un grand chaudron</step>
           <!-- d'autres étapes -->
            <step>enlever la pierre et servir</step>
        </procedure>
    </recipe>
    <recipe n="2">
        <!-- deuxieme recette ici -->
    </recipe>
    <!-- hic desunt multa -->
</cookBook>
```

XML: règles du jeu

- En effet, un document XML représente une arborescence composée de *noeuds*
- il y a un seul noeud racine qui contient tous les autres
- chaque noeud peut être
  - une arborescence
  - un *élément* (qui porte facultativement des *attributs*)
  - une chaîne de caractères
- Chaque élément a un nom et du contenu
- Chaque attribut a un nom et une valeur
- Les noms sont liés avec un *namespace* (espace de noms)

Représentation d'une arborescence XML

- Un document XML linéarisé commence par une instruction de traitement spécial
- Les occurrences d'élément sont marquées entre *balises ouvrantes* et *balises fermantes*
- Les paires nom/valeurs qui constituent les attributs d'un élément peuvent apparaître sans ordre à l'intérieur d'une balise ouvrante
- Les caractères < et & sont Magiques et doivent être cachés au moyen de références entité (< et & respectivement)
- L'espace de noms auquel appartient un élément peut être signalé par un *namespace-prefix* (p.e. xml:) prédéfini
- Les *commentaires* sont délimités par <!-- et -->
- Les *références entité* sont délimités par & et ;
- Les *sections CDATA* sont délimités par <![CDATA[et]]>

Syntaxe XML: le "fine print"

Pour qu'un document soit *bien formé*, il faut que:

1. une seul racine contienne le document entier
2. chaque arborescence soit proprement imbriquée
3. tous les noms soient sensibles à la casse
4. chaque balise ouvrante ait sa balise fermante (sauf qu'on peut combiner les deux, le noeud étant vide)
5. les valeurs d'attribut soient présentées correctement entre guillemets

Bien formé? Oui ou non?

- `<seg>some text</seg>`
- `<seg><foo>some</foo> <bar>text</bar></seg>`
- `<seg><foo>some <bar></foo> text</bar></seg>`
- `<seg type="text">some text</seg>`
- `<seg type='text'>some text</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>`
- `<seg type="text">some text</Seg>`

Bien formé? Oui ou non?

- `<seg>some text</seg>`
- `<seg><foo>some</foo> <bar>text</bar></seg>`
- `<seg><foo>some <bar></foo> text</bar></seg>`
- `<seg type="text">some text</seg>`
- `<seg type='text'>some text</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>`
- `<seg type="text">some text</Seg>`

Bien formé? Oui ou non?

- `<seg>some text</seg>`
- `<seg><foo>some</foo> <bar>text</bar></seg>`
- `<seg><foo>some <bar></foo> text</bar></seg>`
- `<seg type="text">some text</seg>`
- `<seg type='text'>some text</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>`
- `<seg type="text">some text</Seg>`

Bien formé? Oui ou non?

- `<seg>some text</seg>`
- `<seg><foo>some</foo> <bar>text</bar></seg>`
- `<seg><foo>some <bar></foo> text</bar></seg>`
- `<seg type="text">some text</seg>`
- `<seg type='text'>some text</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>`
- `<seg type="text">some text</Seg>`

Bien formé? Oui ou non?

- `<seg>some text</seg>`
- `<seg><foo>some</foo> <bar>text</bar></seg>`
- `<seg><foo>some <bar></foo> text</bar></seg>`
- `<seg type="text">some text</seg>`
- `<seg type='text'>some text</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>`
- `<seg type="text">some text</Seg>`

Bien formé? Oui ou non?

- `<seg>some text</seg>`
- `<seg><foo>some</foo> <bar>text</bar></seg>`
- `<seg><foo>some <bar></foo> text</bar></seg>`
- `<seg type="text">some text</seg>`
- `<seg type='text'>some text</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>`
- `<seg type="text">some text</Seg>`

Bien formé? Oui ou non?

- `<seg>some text</seg>`
- `<seg><foo>some</foo> <bar>text</bar></seg>`
- `<seg><foo>some <bar></foo> text</bar></seg>`
- `<seg type="text">some text</seg>`
- `<seg type='text'>some text</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>`
- `<seg type="text">some text</Seg>`

Bien formé? Oui ou non?

- `<seg>some text</seg>`
- `<seg><foo>some</foo> <bar>text</bar></seg>`
- `<seg><foo>some <bar></foo> text</bar></seg>`
- `<seg type="text">some text</seg>`
- `<seg type='text'>some text</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>`
- `<seg type="text">some text</Seg>`

Bien formé? Oui ou non?

- `<seg>some text</seg>`
- `<seg><foo>some</foo> <bar>text</bar></seg>`
- `<seg><foo>some <bar></foo> text</bar></seg>`
- `<seg type="text">some text</seg>`
- `<seg type='text'>some text</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>`
- `<seg type="text">some text</Seg>`

Bien formé? Oui ou non?

- `<seg>some text</seg>`
- `<seg><foo>some</foo> <bar>text</bar></seg>`
- `<seg><foo>some <bar></foo> text</bar></seg>`
- `<seg type="text">some text</seg>`
- `<seg type='text'>some text</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>`
- `<seg type="text">some text</Seg>`

Bien formé? Oui ou non?

- `<seg>some text</seg>`
- `<seg><foo>some</foo> <bar>text</bar></seg>`
- `<seg><foo>some <bar></foo> text</bar></seg>`
- `<seg type="text">some text</seg>`
- `<seg type='text'>some text</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>`
- `<seg type="text">some text</Seg>`

Bien formé? Oui ou non?

- `<seg>some text</seg>`
- `<seg><foo>some</foo> <bar>text</bar></seg>`
- `<seg><foo>some <bar></foo> text</bar></seg>`
- `<seg type="text">some text</seg>`
- `<seg type='text'>some text</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>`
- `<seg type="text">some text</Seg>`

Validation XML

Pour qu'un document XML bien formé soit considéré *valide*, il faut qu'il conforme à des règles supplémentaires. Ces règles constituent un *schéma*

Un schéma spécifie :

- les noms de tous les éléments légaux
- les noms et les types de valeurs de tous les attributs légaux
- des règles concernant l'imbrication et le contenu des éléments
- les noms des éléments qui peuvent constituer la racine d'un document
- et quelques autres menus propos...

Un schéma donc vous permet de contrôler par exemple que 'tout chapitre ait son titre', que 'toute recette comporte une liste d"ingredients', que 'le valeur de tout attribut *@when* soit conforme au standard ISO' ... etc.

Un espace de noms, par contraste, ne vous permet que de labelli le vocabulaire d'où est derivé un ensemble d'éléments.

Exercice 1

Nous allons expérimenter un logiciel spécialisé pour créer et modifier des fichiers XML...

- Manipulation d'un document XML bien formé
- Creation d'un document XML valide
- Importation d'un document non-XML

Lire, c'est encoder...

- L'interprétation des mots d'un texte n'est pas aléatoire : elle est guidée par les signes de ponctuation, par les changements de police, par leur disposition spatiale etc !
- Pour indiquer les même choses (et d'autres) dans un texte numérique, une balisage devient essentiel
- Le balisage sert ainsi à exprimer nos lectures préalables
- Le balisage rend possible une polyvalence des ressources textuelles et induit des reflexions profondes sur la materialité des textes qu'elles impliquent

La Text Encoding Initiative (TEI) peut nous aider...

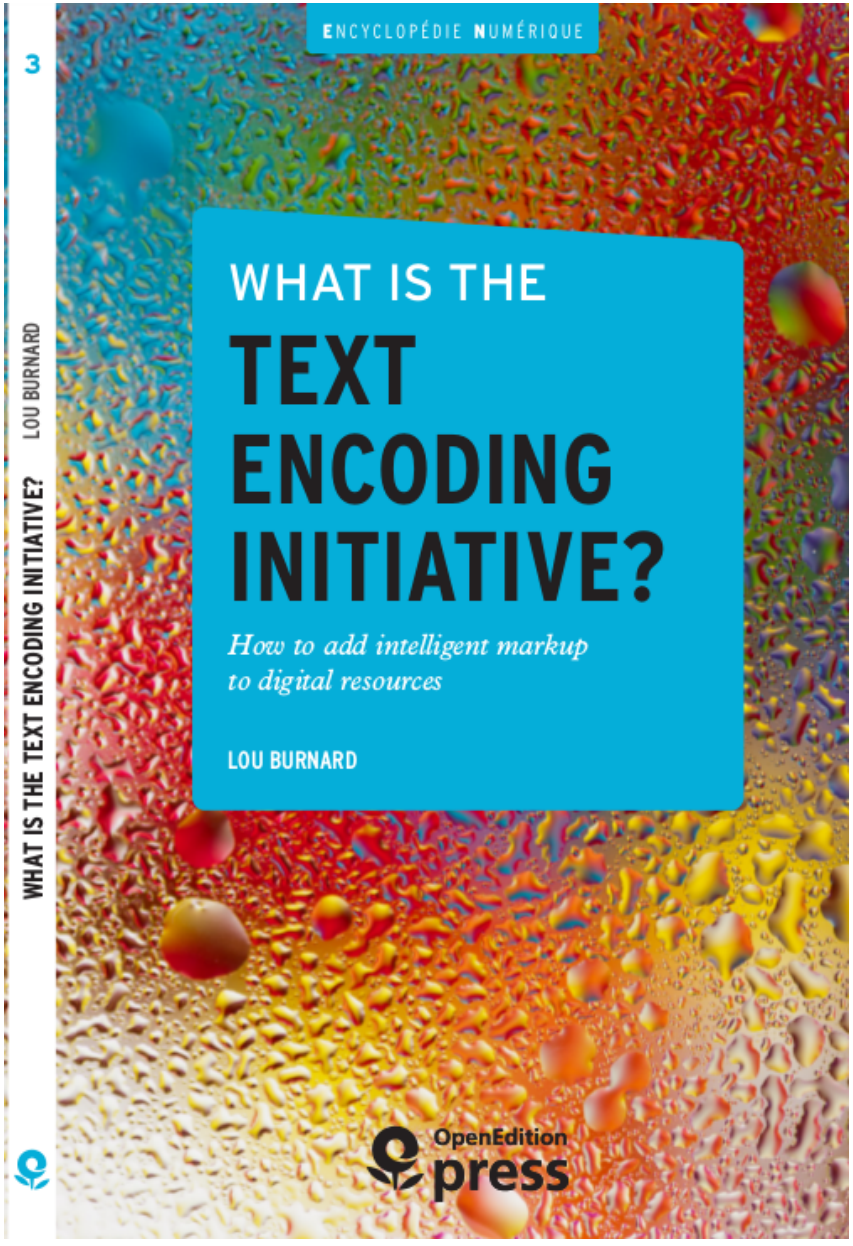
Elle représente un modèle conceptuel de texte bien établi qui facilite :

- la conversion des données existantes
- la création des données nouvelles
- l'intégration des données déja existantes mais répandues dans plusieurs sources

Elle est basée sur des formats ouverts et des technologies ouvertes

Elle s'appuye sur une théorie explicite de l'ontologie textuelle

Qu'est-ce que la Text Encoding Initiative ?



- Une organisation, une institution ?
- un 'club', une mode, une religion ?
- une spécification technique ?
- un gabarit pour la construction des spécifications techniques?

version francaise :
http://books.openedition.org/oep/123

Concrètement la Text Encoding Initiative c'est ...

une 'Initiative pour l'Encodage Textuel'....

- un ensemble de 'recommandations' pour l'encodage des ressources numériques avec XML
- un infrastructure internationale responsable de la maintenance, de l'évolution, et de la distribution de ces recommandations
- une communauté internationale d'utilisateurs de ces recommandations

Plutot un cadre permettant de réflechir sur ce que c'est qu'un texte numérisé qu'un "standard" fixe.

Les enjeux de la TEI

- Faciliter la création, l'échange, et l'intégration des données textuelles informatisées
  - toute sorte de textes
  - toutes les langues
  - toute origine temporelle ou culturelle
- La TEI s'adresse également ...
  - aux débutants, cherchant des solutions bien connues et consensuelles
  - aux experts, cherchant à créer de nouvelles solutions

Les buts de la TEI

- faire des recommandations qui se basent sur un consensus existant
- préférer les solutions générales à celles spécifiques à une discipline
- en même temps permettre la spécialisation et l'extension

D'où est sortie la TEI ?

- En automne 1987 aux États-Unis, la NEH finance une réunion internationale sur la possibilité de définir des "text encoding guidelines"
- C'était un projet de recherche en "humanities computing"...
- Influences majeures
  - bibliothèques et archives numerisées
  - ingénierie linguistique
  - édition des sources littéraires ou historiques

Vous souvenez-vous de l'an 1987 ?

La Text Encoding Initiative est née dans un monde très différent...

- C'était l'été de *Joe le taxi*, premier tube de Vanessa Paradis...
- le world wide web n'existait pas
- le tunnel sous la manche était en construction
- un état nommé l'Union Soviétique venait de lancer une station spatiale appelée "Mir" .. et de subir un désastre à Tchernobyl
- l'informatique sérieuse s'éffectuait uniquement sur les grosses machines dites 'mainframes '

Vous souvenez-vous de l'an 1987 ?

La Text Encoding Initiative est née dans un monde très différent...

- C'était l'été de *Joe le taxi*, premier tube de Vanessa Paradis...
- le world wide web n'existait pas
- le tunnel sous la manche était en construction
- un état nommé l'Union Soviétique venait de lancer une station spatiale appelée "Mir" .. et de subir un désastre à Tchernobyl
- l'informatique sérieuse s'éffectuait uniquement sur les grosses machines dites 'mainframes '

Vous souvenez-vous de l'an 1987 ?

La Text Encoding Initiative est née dans un monde très différent...

- C'était l'été de *Joe le taxi*, premier tube de Vanessa Paradis...
- le world wide web n'existait pas
- le tunnel sous la manche était en construction
- un état nommé l'Union Soviétique venait de lancer une station spatiale appelée "Mir" .. et de subir un désastre à Tchernobyl
- l'informatique sérieuse s'éffectuait uniquement sur les grosses machines dites 'mainframes '

Vous souvenez-vous de l'an 1987 ?

La Text Encoding Initiative est née dans un monde très différent...

- C'était l'été de *Joe le taxi*, premier tube de Vanessa Paradis...
- le world wide web n'existait pas
- le tunnel sous la manche était en construction
- un état nommé l'Union Soviétique venait de lancer une station spatiale appelée "Mir" .. et de subir un désastre à Tchernobyl
- l'informatique sérieuse s'éffectuait uniquement sur les grosses machines dites 'mainframes '

Vous souvenez-vous de l'an 1987 ?

La Text Encoding Initiative est née dans un monde très différent...

- C'était l'été de *Joe le taxi*, premier tube de Vanessa Paradis...
- le world wide web n'existait pas
- le tunnel sous la manche était en construction
- un état nommé l'Union Soviétique venait de lancer une station spatiale appelée "Mir" .. et de subir un désastre à Tchernobyl
- l'informatique sérieuse s'éffectuait uniquement sur les grosses machines dites 'mainframes '

Vous souvenez-vous de l'an 1987 ?

La Text Encoding Initiative est née dans un monde très différent...

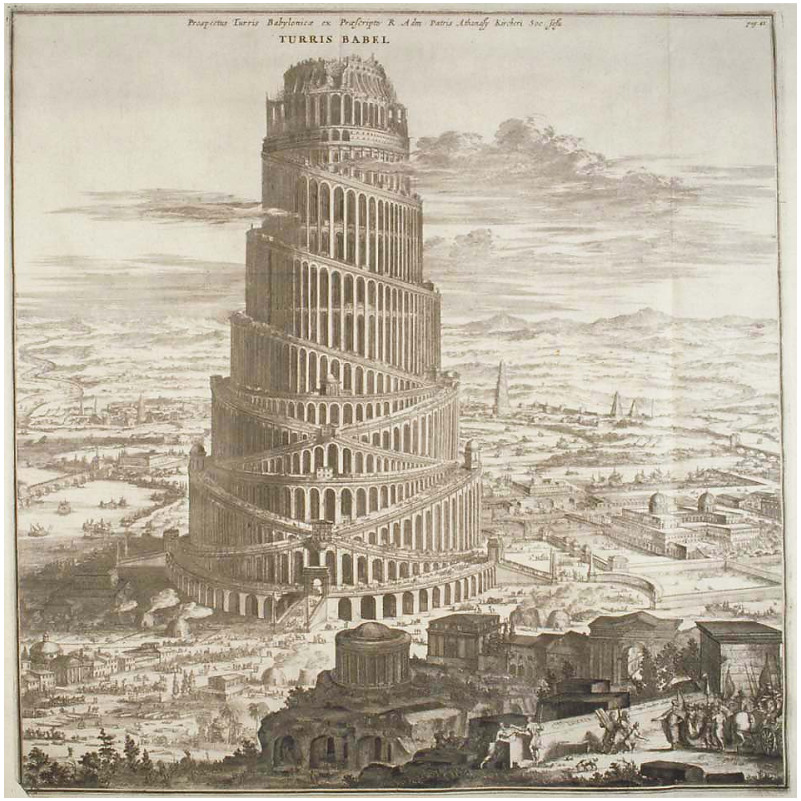
- C'était l'été de *Joe le taxi*, premier tube de Vanessa Paradis...
- le world wide web n'existait pas
- le tunnel sous la manche était en construction
- un état nommé l'Union Soviétique venait de lancer une station spatiale appelée "Mir" .. et de subir un désastre à Tchernobyl
- l'informatique sérieuse s'éffectuait uniquement sur les grosses machines dites 'mainframes '

…mais aussi dans un monde un peu familier…

- Les disciplines ”linguistique de corpus” et ”intelligence artificielle” avaient établi la nécessité de travailler avec des ressources numérisées et à grande échelle
- Des avancées en traitement de texte commençaient à avoir un effet sur la lexicographie et les systèmes de gestion documentaire (TeX, Scribe, tRoff..)
- L’ Internet existait, et les théories sur comment en profiter d’une manière ‘hypertextuelle’ abondaient
- On confrontait déjà les problèmes de pérennisation des données et d’incompatabilités technologiques (ex. les CD).

Pourquoi cet effort ?

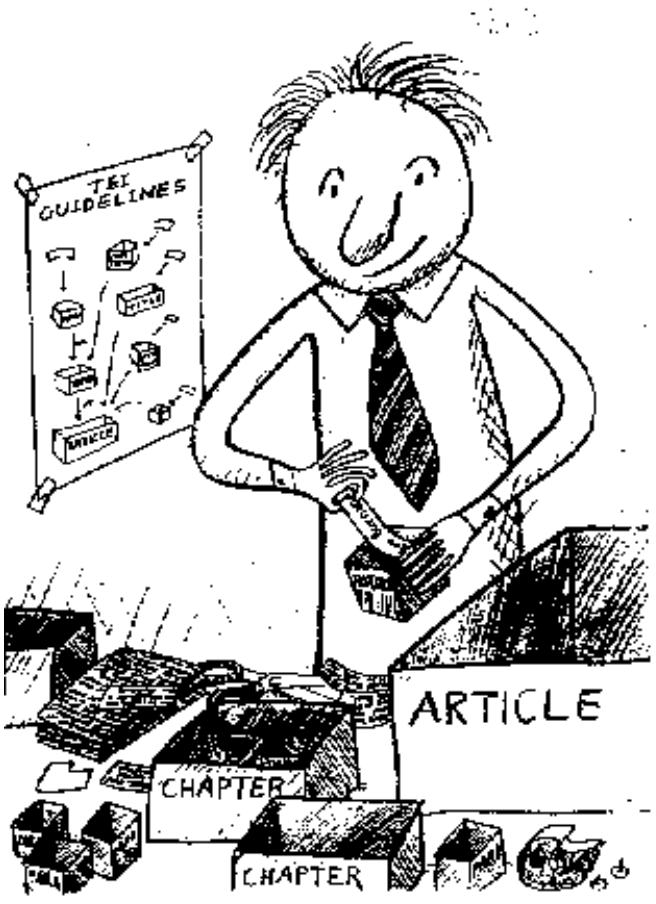
- Parce qu'on s'est aperçu qu'on risquait une nouvelle confusion de langues avec l'arrivée de l'informatique dans la représentation des données textuelles !
- Mais aussi peut-etre un désir de mettre à jour les traditions philologiques de la gestion des textes?



TEI chronologie

1988 - 1990	$€ Recherche financé phase 1: production de TEI P1
1990 - 1992	$$ €€ Recherche financé phase 2: production des fascicules de TEI P2
1993 - 1994	$$$ €€ integration des chapitres de P2 comme TEI P3
1995 - 1999	$ Promotion et prise en main TEI
2000	Établissement du Consortium TEI (incorporé dec)
2001 - 2003	$ Conversion de P3 en XML (TEI P4), lancement d'une révision complète
2003 -	TEI P5 : révisions régulières 2 fois par an ; sur github depuis 2005 ; la version 3.3.0 vient d'apparaitre

Personnalisation



ICAME Journal, 1992

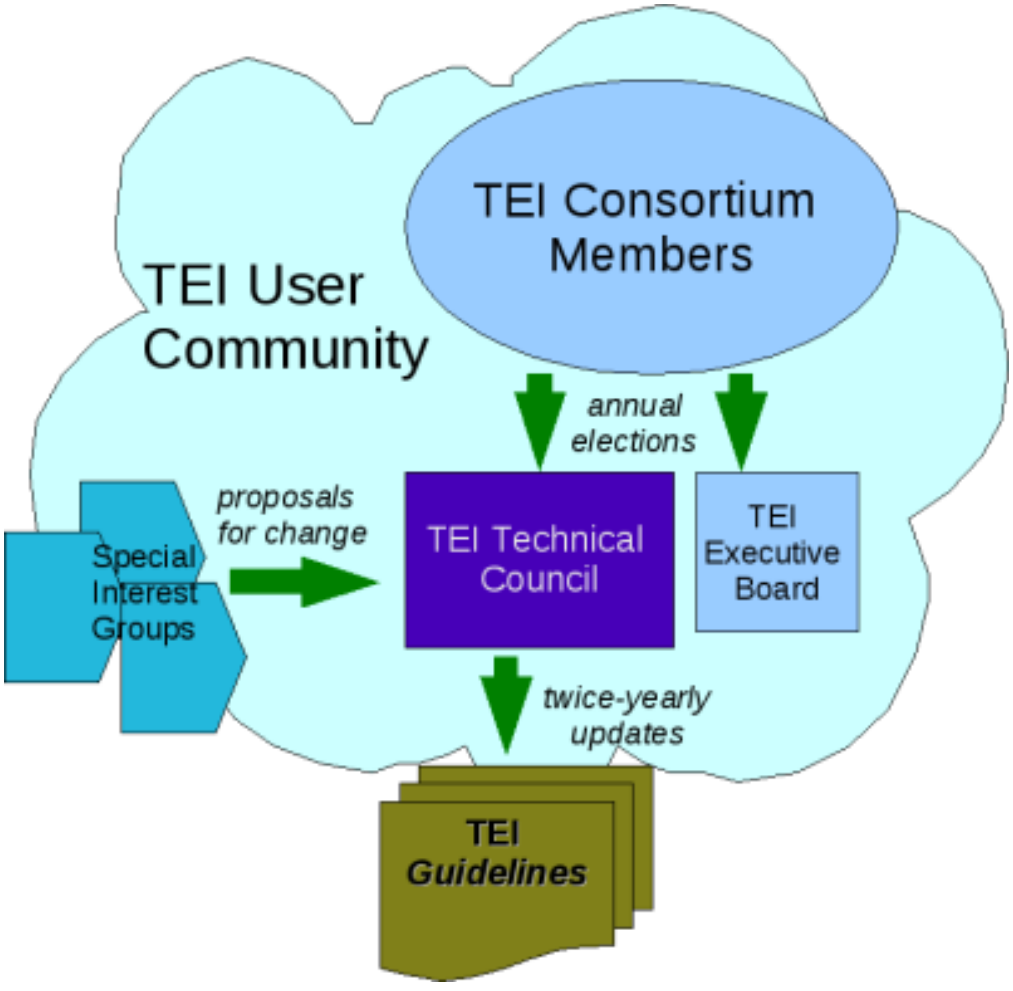
La TEI fournit un gabarit – une espece de kit Lego – pour la construction d'un systeme de balisage adapté aux besoins spécifiques d'un projet particulier, tout en restant comprehensible par d'autres projets ou d'autres systemes. L'essentiel, c'est l'explicitation des choix effectués, et des personnalisations eventuelles.

Qu'est-ce que la TEI aujourd'hui ?

- Consortium international établi en 2000 (voir http://www.tei-c.org/)
- Un ensemble de *Guidelines* (lignes directrices) peu prescriptives
- représentant un consensus au sujet des distinctions significatives dans un vaste ensemble de matériaux textuels
- qui s'expriment en deux gros volumes de prose et un ensemble de définitions formelles
- derrière cet ensemble se trouve un modèle formel ("conceptual schema") de plus en plus élaboré, organisé en système de classes, voire en ontologie

... et une communauté internationale active

Organisation actuelle de la TEI



La TEI facilite un balisage 'intelligent'

La TEI de nos jours s'applique à l'encodage des...

- composants structuraux et fonctionnels d'un texte
- transcriptions diplomatiques des sources historiques, des images, des annotations
- liens, correspondances, alignements
- données et entités : par exemple de temps, personnes, lieux ou événements
- annotations peritextuelles et métatextuelles (correction, suppression, ajouts)
- analyses linguistiques
- métadonnées de plusieurs types
- ... et définitions formelles de schéma XML !

Il faut faire son choix ....

Il n'y a pas de "TEI dtd"

- TEI est un système *modulaire*. On s'en sert pour créer un système d'encodage selon ses propres besoins, en sélectionnant des *modules* spécifiques
- Chaque module définit un groupe d'éléments (et leurs attributs)
- on peut sélectionner les éléments voulus, et même en changer des propriétés
- on peut y mélanger des éléments nouveaux, ou bien natifs ou bien d'autres standards

Exercice 2

On va explorer cela avec Roma : un outil pour la construction des personnalisations TEI.

http://www.tei-c.org/Roma/

Où sont les outils TEI-XML?

- La TEI ne vous fournit ni usine à gaz, ni boîte d'outils..
- Dès le début, elle essaie de se distinguer nettement de la production des outils, pour mieux garantir son indépendence
- Les Guidelines sont conçus comme expression concrète d'un modèle abstraite des objets – pour la plupart textuels – qui sont d'interêt scientifique dans les SHS...
- Mais c'est réservé aux communautés d'utilisateurs de se décider comment traiter ces objets, et donc de construire les outils pour en profiter.

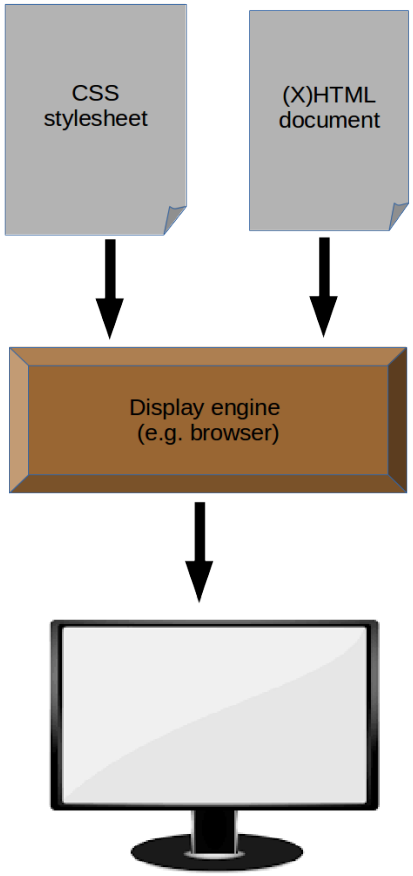
OK, mais quand même...

Les TEI Guidelines s'expriment en XML, comme toute application TEI

- en principe, alors, n'importe quel logiciel conforme au norme XML nous suffirait
- mais il faut le construire!

Heureusement, la construction des outils devient beacoup plus simple à cause du disponibilité de quelques technologies standardisés très génériques... notamment CSS, XSLT, XPath et XQuery

Comment se servir de CSS ?



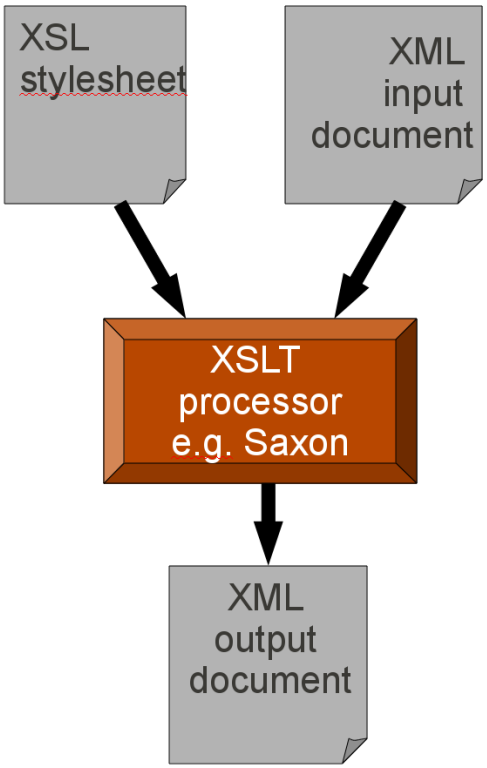
CSS est un langage d'affichage

Principes de CSS

- Sont définis par le standard CSS un ensemble de `propriétés` (style et taille de police, couleur, alignement etc.)
- Une feuille de style CSS définit des règles associant des valeurs pour ces propriétés avec tout élément de type x, ou tout élément dans tel contexte, ou avec telle valeur d'attribut etc
- Les propriétés applicables à un élément sont hérités par défaut de son parent
- Les règles d'association peuvent etre spécifiés par un document externe, ou explicitement dans un doc HTML

Bémol: on ne peut rien changer du document traité: CSS ne définit que son affichage

Comment se servir de XSLT ?



XSLT est un langage de transformation

Principes de XSLT

- Un document XML est transformé en traversant toute l'arborescrence, sous controle d'une feuille de style XSLT contenant des `templates`
- Si un noeud du document correspond à un de ces templates, le template est actionné
- Un template peut accéder à tout partie du document, en utilisant un langage standardisé XPath
- Si aucun template n'est specifié pour un élément, par défaut le contenu textuel de l'élément est emis

Une transformation typique

A partir de ceci :

```
<div type="recette" n="34">
 <head>Pasta pour les debutants</head>
 <list>
  <item>pates</item>
  <item>fromage râpé</item>
 </list>
 <p>Faire bouiller les pates, et melanger avec le
fromage.</p>
</div>
```

on veut produire :

```
<html>
 <h1>34: Pasta pour les novices</h1>
 <p>Ingrédients: pates fromage râpé</p>
 <p>Faire bouiller les pates, et melanger avec le
fromage.</p>
</html>
```

Comment exprimer cela en XSL?

```
<xsl:stylesheet xpath-default-namespace="http://www.tei-
c.org/ns/1.0"
 version="2.0">
 <xsl:template match="div">
  <html>
   <h1>
    <xsl:value-of select="@n"/>:
<xsl:value-of select="head"/>
   </h1>
   <p>Ingrédients:
<xsl:apply-templates select="list/item"/>
   </p>
   <p>
    <xsl:value-of select="p"/>
   </p>
  </html>
 </xsl:template>
</xsl:stylesheet>
```

Quid de XPath et XQuery ?

XPath est un langage standardise pour addresser les parties dune arborescence XML

```
document("books.xml")/bookstore/book[price>30]../title
```

Xquery est un langage d'interrogation de base de donnees XML

```
 for $x in
doc("books.xml")/bookstore/book where $x/price>30
order by $x/title
return $x/title
```

Comment distribuer ses ressources XML-TEI ?

La politique du moindre effort...

- Voici nos fichiers XML-TEI. Debrouillez-vous.
  `http://www.cnrtl.fr/corpus/estrepublicain/`
- Le palimpsest d'archimede`http://archimedespalimpsest.net`
- Oxford Text Archive (`http://ota.ox.ac.uk/`)

Systèmes d'édition

Pour une chaine de production traditionnel, on peut utiliser un modèle simple:

- création/modification avec outil bureautique classique, sous contrôle des styles
- transformation XSLT vers TEI
- validation et modification
- transformation vers format de saisie d'une système PAO par ex inDesign

Pour la gestion, stockage, affichage etc. d'un ensemble de documents TEI sur le web on a maintenant plusieurs choix :

- des plug-ins pour les CMS les plus populaires (par ex. Drupal, Zotero, Omeka)
- des systèmes conçus pour TEI (par ex. Kiln, TEI Boilerplate, Lodel, Cetecean)
- des systèmes génériques de gestion de documents par ex xt

Exemples d'applications XTF

- Ecrivains féminins victoriens `http://webapp1.dlib.indiana.edu/vwwp/projectinfo/technical.doc`
- Les noms et les lieux dans le corpus de Rabelais `http://renom.univ-tours.fr/`

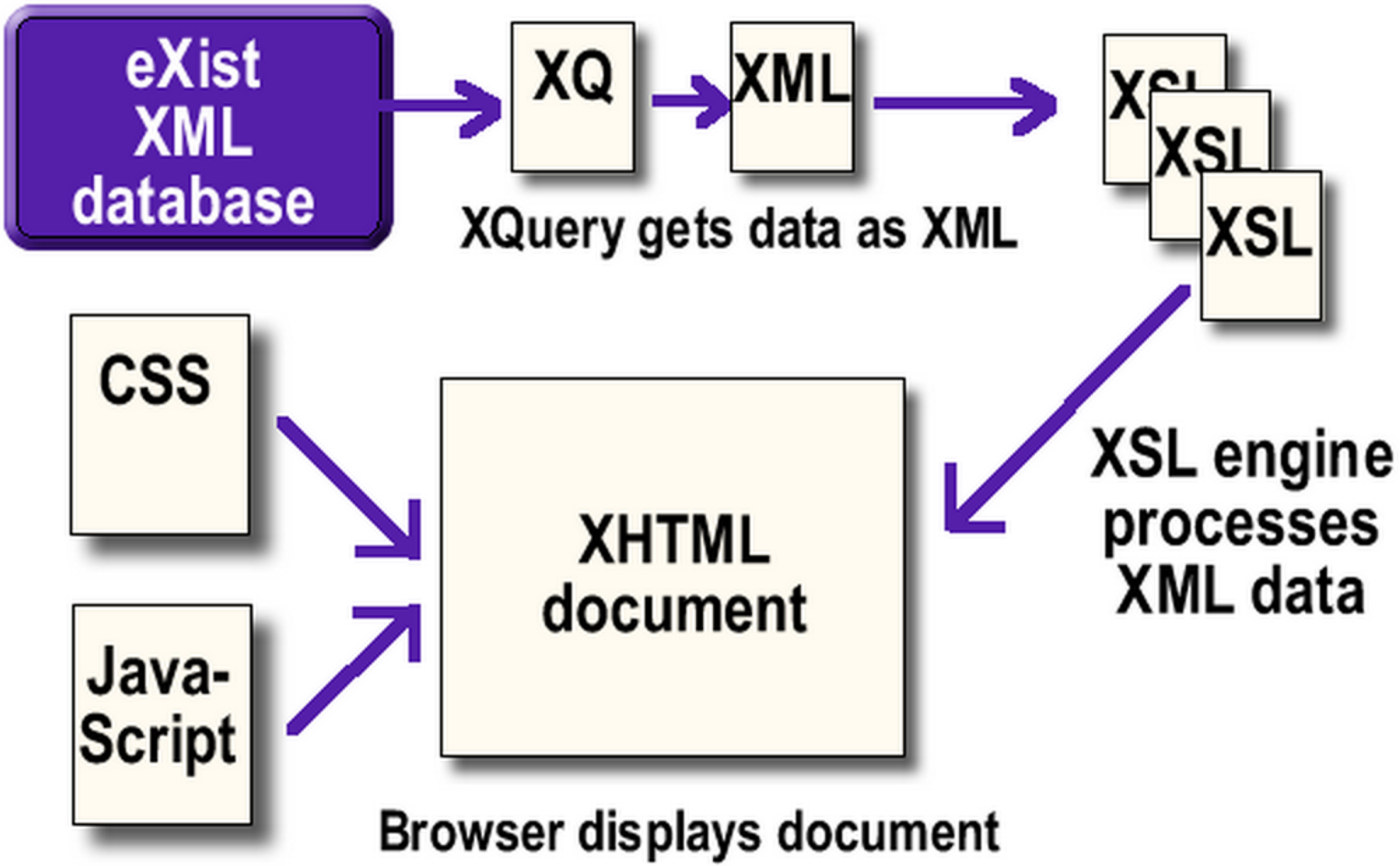
Rechercher, analyser..

Bases de données XML génériques comme

- baseX `http://basex.org`
- eXist `http://exist-db.org`

La solution par excellence pour les projets disposant d'un quantité important de documents/objets XML TEI

Architecture typique



Exemples...

- Colonial Despatches:
  `http://bcgenesis.uvic.ca/docsByDate.htm`
- Carl Maria van Weber
  Archive`http://weber-gesamtausgabe.de/en/A002068/Correspondence`

Outils scientifiques

Outils élaborés par/pour des communautés savantes spécialistes, par ex:

- 'textometrie' : analyse statistique des mots
- élaboration d'un apparat critique
- analyse de l'oral
- analyses linguistiques

quelques exemples suivent ...

http://textometrie.ens-lyon.fr/

D'autres outils "TEI-friendly"

- Ancient Wisdoms : kiln `http://www.ancientwisdoms.ac.uk/method/software-install/`
- Shelley-Godwin archive : shared canvas viewer `http://shelleygodwinarchive.org/about`
- Letters and mss of 19th c Berlin `http://tei.ibi.hu-berlin.de/berliner-intellektuelle/manuscript?Sandmann+en#5`
- Bibliotheque Virtuelle des Humanistes : philologic `http://www.bvh.univ-tours.fr/Epistemon/philologic.asp`

L'esprit TEI

Qu'est-ce que cela veut dire : « être conforme » à la TEI ?

- une pratique de balisage consensuelle
- un lexique commun
- un respect de l'autonomie

La standardisation ne devrait pas signifier « fais comme moi » ; elle veut dire « explique-moi ce que tu fais. »

… d'où les variations TEI

Par exemple : éléments pour description bibliographique : On a le choix entre

- <bibl> qui contient n'importe quel mélange de composants bibliographiques … ou aucun
- <biblStruct> qui contient une sélection prédéfinie d'éléments, strictement structurés

Etre conforme à la TEI veut dire quoi?

- **être honnet** : Les éléments XML qui se déclarent comme appartenant au namespace TEI doivent respecter les définitions TEI de ces éléments
- **être explicite** : Pour valider un document TEI, un ODD est fortement conseillé, parce que cela mettra en évidence toutes les modifications effectuées

L'objet de ces règles est de faciliter le "blind interchange" des documents – mais ils ne le garantissent pas.

Pourquoi continuer de s'intéresser à la TEI ?

Deux raisons pour lesquelles les standards échouent le plus souvent :

- ils sont basés sur une théorie pas encore mûre
- 'not invented here' : la communauté envisagée est trop diverse ou fragmentée

Comment faire mûrir une théorie ?

Dans son TEI ODD, on peut :

- limiter les valeurs possibles d'un attribut plus ou moins strictement
- proposer des règles Schematron sur le contenu (p.e. co-dependency)
- enlever quelques éléments facultatifs
- ajouter de nouveaux éléments, labellisés dans votre propre espace de noms

Donc on peut évoluer et tester sa théorie, en restant toujours TEI-conforme.

Not Invented Here?

- TEI P5 a des possibilités très extensives pour l'I18N...
- TEI héberge volontairement d'autres espaces de noms
- Donc on peut se servir des autres schémas existants :
  - SVG pour les graphiques
  - MathML pour les maths
  - DCMI pour les metadonnées
  - …
- La définition d'un élément TEI peut inclure (s'il y en a) son mapping avec d'autres ontologies, formalisé par un élément <equiv> (équivalent)

L'évolution darwinienne, ça marche…

- faites vos modifications dans votre espace de nom
- documentez-les dans un ODD
- faites discuter vos propositions sur la liste TEI-L, ou dans un SIG !
- proposez les modifications efficaces au Conseil Scientifique de la TEI, en faisant une "feature request" sur sourceforge
- Il y a une version nouvelle de TEI P5 deux fois par an…

... et n'oubliez pas de vous abonner au Consortium !

Pour en savoir plus

- Site du consortium : http://www.tei-c.org
- Depot du consortium : https://github.com/TEIC/TEI
- S'inscrire a TEI-L : http://listserv.brown.edu/archives/cgi-bin/wa?SUBED1=tei-l&A=1
- S'inscrire a tei-fr : https://groupes.renater.fr/wiki/tei-fr/
- TEI Wiki: http://wiki.tei-c.org/
- Technical Council: http://lists.tei-c.org/pipermail/tei-council/
- Twitter: @teiconsortium
- Facebook: http://www.facebook.com/groups/TEIconsortium/