XML,

Un méta-langage pour des documents structurés

Langages écrits : quelques définitions

L'écriture est un système de représentation graphique d'une langue, au moyen de **signes** (dits aussi « **caractères** ») inscrits ou dessinés sur un support, et qui permet l'échange d'informations sans le support de la voix.

Les écritures *glottographiques* (écriture de langues parlées) peuvent être séparées en deux grands groupes :

* Les écritures de type *phonologique* qui transcrivent la structure phonologique ou phonétique de la chaîne parlée (phonèmes ou phones). Les écritures alphabétiques et syllabiques appartiennent à ce groupe.

* Les écritures *sémantiques* dans lesquelles on écrit les morphèmes d'une langue, c'est-à-dire les unités minimales douées de sens de la chaîne parlée. Les écritures chinoises, hiéroglyphiques ou cunéiformes appartiennent à ce groupe.

Les écritures glottographiques comportent des caractères de balisage pour la césure des mots et des phrases, espaces, tirets, sans effet phonologiques.

En informatique, la *balise* est un **caractère**, ou une **série de caractères**, utilisée pour la structuration d'un document et qui reste invisible par le lecteur final. (source Wikipedia)

La notion de « balisage » (« Markup » en anglais)

Le balisage de textes est une modalité de leur annotation, initialement conçue pour traiter leur forme de présentation, d'abord imprimée, puis ultérieurement affichée sur des écrans.

L'exemple ci-après de document, balisé selon le standard RTF (propriétaire de Microsoft®), montre l'utilisation de polices de caractères, et de changements de taille (fs = font size):

```
{\rtf1\ansi
        {\fonttbl
           {\f0\fnil\fcharset0\fprq0\fttruetype Helvetica;}
           {\f1\fnil\fcharset0\fprq0\fttruetype Bitstream Charter;}}
        {\f1\fs24 Ceci est un texte accentu\'e9}
        \par
        {\f0\fs24 avec des caract\'e8res {\b gras},}
        \par
        {\f1 des {\fs18 petits} et des {\fs32 gros}.}
        }
```

GML : un standard de « balisage » conçu par IBM

Avant Microsoft, la compagnie IBM avait aussi dû créer son propre langage de balisage, de même qu'elle avait développé son propre système d'encodage de caractères (EBCDIC).
L'exemple de document GML (qui l'auto-documente) :

```
:h1.Chapter 1:  Introduction
:p.GML supported hierarchical containers, such as
:ol
:li.Ordered lists (like this one),
:li.Unordered lists, and
:li.Definition lists
:eol.
as well as simple structures.
:p.Markup minimization (later generalized and formalized in SGML),
allowed the end-tags to be omitted for the "h1" and "p" elements.
```

SGML, standardisation de GML

```
<!DOCTYPE book PUBLIC "-//OASIS//DTD DocBook V4.1//EN" [
<!ENTITY truc "<emphasis role='bold'>truc</emphasis>">
<!ENTITY currentdate "<emphasis>Sun Feb 01 20:00:00 CEST 2004</emphasis>">  ]>
<book id="truc">
  <bookinfo>
    <title>&truc; Documentation - &currentdate;</title>
    <authorgroup>
     <author>
       <firstname> Paul </firstname>
       <surname> DURAND </surname>
       <affiliation>
         <address> <email>paul.durand@durand.org</email> </address>
       </affiliation>
     </author>
   </authorgroup>
   <copyright> <year>2004</year> <holder>Durand Paul</holder> </copyright>
   <legalnotice>
    <para> Cette documentation est...</para>
   </legalnotice>
  </bookinfo>
  <toc></toc>
  <chapter id="compilation-ch">
    <title>Compilation</title>
    <sect1>
      <title>Compilation des sources</title>
      <para> Procédure de compilation des sources ici</para>
    </sect1>
 </chapter>
</book>
```

Hytime, SMDL, applications de SGML pour la musique

Pour être efficace, l'expression en SGML de documents musicaux nécessitait de pouvoir mettre en facteur des éléments communs, de répétition, de refrains, ce qui nécéssite de prévoir des renvois dans le flux du texte, qu'une application devait être en mesure de traiter.

Ce fût donc à cette occasion qu'apparût le premier standard d'hyperliens.

un exemple simple de document SGML Hytime

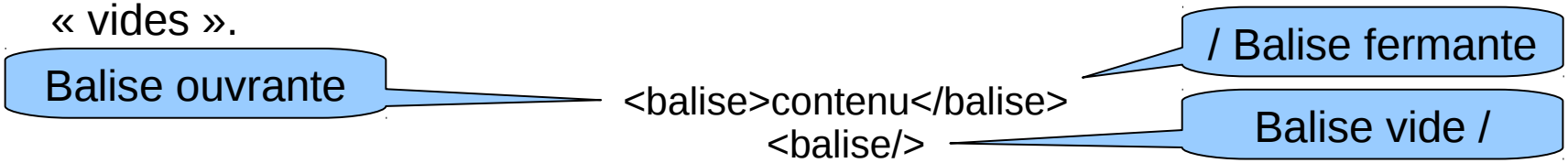
```
<!DOCTYPE    book  [
<!ELEMENT    book - O (citation|location|text)*>
< !ATTLIST    book HyTime (HyDoc) #FIXED HyDoc>
<!ELEMENT    (citation|location|text) - O (#PCDATA)>
<!ATTLIST    textid ID #IMPLIED>
<!ATTLIST    citation
 HyTime     (ilink)   #FIXED   ilink
 anchors    IDREFS    #REQUIRED
 anchrole   CDATA     #FIXED  "start end"
 HyNames    NAMES     #FIXED  "anchors linkends">
<!ATTLIST    location
 HyTime     (dataloc) #FIXED   dataloc
 id         ID        #REQUIRED
 locsrc     IDREF     #REQUIRED
 quantum    (norm)    #FIXED   norm
 reftype    CDATA     #FIXED   "locsrc text">
]>
 <book>
<citation anchors = "firstword lastword">
 <location id = firstword locsrc = phrase>1 1
 <location id = lastword locsrc = phrase>-1 1
 <text id = phrase> this cites this
```

Génèse du méta langage XML

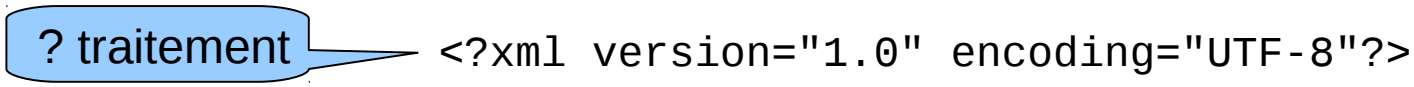
- **XML** est issu d'une filiation,
  - de **SGML**, standard ISO,
    - issu de **GML** Standard propriétaire IBM,
    - dont est issu **HTML,**
      - qui est une DTD **SGML,**
      - qui intègre les fonctionnalités de la DTD **SGML Hytime,**
    - dont il restreint les libertés de conventions informaticiennes,
    - dont il étend indéfiniment les possibilités d'expressions.
- **XHTML** est un redressement de **HTML**, de **SGML** en **XML.**
- **XML** est à l'origine de langages d'expression :
  - XML Schema (dit **XSD**) modélisation de structures **XML,**
  - RDF, RDFS, OWL représentation de réseaux en **XML.**

XML « bien formé »

- De par les libertés informatiques que ses concepteurs s'étaient accordées, notamment pour éluder les fermetures de balises, l'analyse syntaxique d'un document GML ou SGML nécessite l'analyse syntaxique et sémantique préalable de son modèle.
  - *En conséquence les « parsers » SGML sont coûteux à développer et à utiliser.*
- Comme en SGML, un document XML est un arbre, articulé à partir d'une balise racine.
- XML contraint les informaticiens à expliciter la fermeture des balises, et les balises « vides ».



- XML contraint les informaticiens à expliciter en première ligne pour son traitement le type de document XML à effectuer et son encodage.



  - *En conséquence les « parsers » et les outils de traitement XML sont faciles à réaliser, et gratuits à l'utilisation.*

Balises et attributs

- Comme en SGML, une balise XML peut avoir des attributs :
  - déclarés à l'intérieur de la balise ouvrante,
  - à raison d'un attribut par nom de celui-ci,
  - au contenu,
    - assigné par le signe =
    - placés entre deux signes ' ou "

    <balise auteur="moi"> contenu </balise>

discrimination des homonymes

- Pour éviter les difficultés associées aux conflits de noms identiques pour des balises ou des attributs de rôles ou de sémantique différentes, XML intègre de façon explicite la notion linguistique de **champ lexical**.
- *On parle de **champ lexical** pour désigner un ensemble théorique de noms, de substantifs, d'adjectifs et de verbes appartenant à une même catégorie syntaxique et liés par leur domaine de sens.* (source Wikipedia).
- Dans le champ lexical des informaticiens, un champ lexical se désigne par « ***espace de noms*** » ou « ***Namespace*** » en anglais.
- Un espace de noms se déclare comme attribut **xmlns** de la balise racine d'un document, suffixé du préfixe contextuel d'association de noms de balises ou d'attributs à un espace de noms.
- La syntaxe d'un espace de nom est celle d'une URI

```
<baliseracine xmlns:toto="http://MonChampLexical.org" >
    <toto:mabalise><toto:mabalise>
    </baliseracine>
```

XML « valide » : DTD

- Bien formé, un document XML peut être, ou non, contraint au respect d'un modèle de structure.
- La première sorte de modèles de structure est la DTD (Document Type Definition) issue de SGML.
  Elle est déclarée juste après l'instruction de traitement XML :

```
<?xml version="1.0" encoding="UTF-8"?>
```

  - Soit référée à une ressource externe :
    <!DOCTYPE hello SYSTEM "hello.dtd">
  - Soit décrite à cet endroit :

```
<!DOCTYPE hello [
 <!ELEMENT hello (#PCDATA)>
]>
```

XML « valide » XSD

- XML Schema est la recommandation du W3C pour les modèles de validation.

  Un modèle XML Schema est toujours une ressource externe, déclarée comme attribut de la balise racine d'un document :

```
<book
 isbn="0836217462"
 xmlns:xsi="http: // www.w3.org/2000/10/XMLSchema-instance
 xsi:noNamespaceSchemaLocation="file : /// library.xsd">
</book>
```

- Un schéma fournit :
  - des déclarations d'éléments et d'attributs, appartenant à un espace de noms,
  - des définitions de types.
- Chaque instance de document:
  - utilise un ou plusieurs espaces de noms,
  - peut associer chaque espace de nom à un schema XML.

Fin du module