

Question pratique : Evaluation

Modélisation et interopérabilité : Semaine 40, cours 3

Benoît Valiron <benoit.valiron@monoidal.net>

<http://inf356.monoidal.net/>

- Trois choses :
 - Un examen final (coef 2/3)
 - Un examen partiel (coef 1/6)
 - Mardi 10 Novembre, de 12h à 15h
 - Salle à déterminer
 - Contenu : format XML, DTD, Manipulation de formats, Relax NG.
 - Documents autorisés (mais pas de livres...)
 - Un projet (coef 1/6)
 - À rendre la dernière semaine de cours
 - Support numérique : vous assurer que je peux le lire !

1

2

Le projet

- Sujet en ligne ce week-end
- Il vous sera demandé :
 - Création d'une base de donnée au format XML
 - Horaires des TER en Rhône-Alpes
 - Fichiers de validation DTD et Relax NG
 - Documentation du format que vous avez développé :
 - Inclus dans le Relax NG
 - En format docbook
 - Des fonctions XPath pour récupérer des infos
 - Des transformations XSLT pour un affichage de résultats
 - Un affichage graphique (SVG et google-map)

3

4

Aujourd'hui, petit tour du propriétaire.

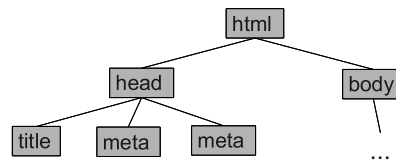
XML : où le trouver ?

- Pages internet : XHTML
- Images : SVG
- Maths : MathML
- Données textuelles
 - Données GPS : openstreet, GPX, google KML...
- Flux RSS
- Méta-données
 - RDF et Dublin Core
- Traitements de texte :
 - ODF : OpenOffice
 - OOXML : MS Word
- Protocole d'échange de données entre applications : SOAP
- ...

5

XHTML

- Application XML lisible par un navigateur.
- Nom qualifié : <http://www.w3.org/1999/xhtml>
- Format (simplifié) :



```
<!ELEMENT html (head, body)>  
<!ELEMENT head (meta*, title, meta*)>  
<!ELEMENT title (#PCDATA)>  
<!ELEMENT meta EMPTY>
```

6

3 DTDs pour XHTML

- **XHTML Strict :**
`<!DOCTYPE html PUBLIC
 "-//W3C//DTD XHTML 1.0 Strict//EN"
 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">`
- **XHTML Transitional :**
`<!DOCTYPE html PUBLIC
 "-//W3C//DTD XHTML 1.0 Transitional//EN"
 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">`
- **XHTML Frameset :**
`<!DOCTYPE html PUBLIC
 "-//W3C//DTD XHTML 1.0 Frameset//EN"
 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">`

7

<http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd>

```
... Entités paramètres ...
<!--==== Document Structure =====>
<!ELEMENT html (head, body)>
<!ATTLIST html
  %i18n;
  id          ID          #IMPLIED
  xmlns       #URI;      #FIXED 'http://www.w3.org/1999/xhtml'
  >
<!--==== Document Head =====>
<!ENTITY % head.misc "(script|style|meta|link|object)*">
<!ELEMENT head (%head.misc;,  

  ((title, %head.misc;,(base, %head.misc;)? ) |  

  (base, %head.misc;,(title, %head.misc;))))>
<!ATTLIST head
  %i18n;
  id          ID          #IMPLIED
  profile     #URI;      #IMPLIED
  >
<!ELEMENT title (#PCDATA)>
<!ATTLIST title
  %i18n;
  id          ID          #IMPLIED
  >
.... Plus de règles ...
```

Le fichier fait 978 lignes de code, dont 200 lignes de commentaires.
 • 47 ELEMENT
 • 77 ATTLIST
 • 54 ENTITY

8

Élément body

- Texte structuré :
 - Titres : h1 h2 h3 h4 h5
 - Paragraphes : p
 - Sauts de ligne : br
 - Style de texte : b i tt
 - Tableaux : table tr td
 - Images : img
 - Listes : ul li
 - ...

9

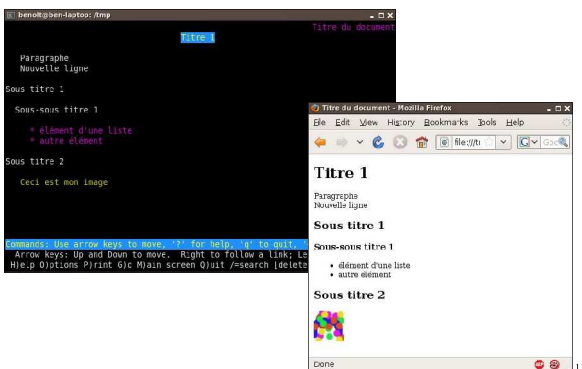
Un document XHTML

- Extension : .html ou .xhtml
- Appel à l'espace de nom :

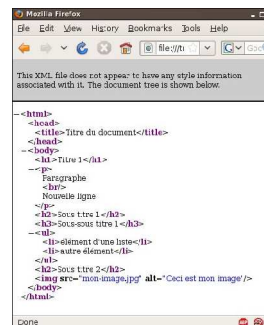
```
<?xml version="1.0"?>
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>Titre du document</title>
  </head>
  <body>
    <h1>Titre 1</h1>
    <p>
      Paragraphe
    <br />
    Nouvelle ligne
    </p>
    <h2>Sous titre 1</h2>
    <h3>Sous-sous titre 1</h3>
    <ul>
      <li>élément d'une liste</li>
      <li>autre élément</li>
    </ul>
    <h2>Sous titre 2</h2>
    
  </body>
</html>
```

10

Visualisation



Sans espace de nom...

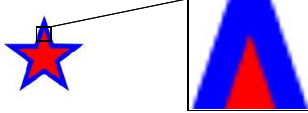


- Firefox est perdu.
- Lynx est plus malin et infère le format du fichier.

12

Format d'image

- Les formats JPG, GIF, PNG... sont des formats binaires : description de l'image pixel par pixel.
- Ne supportent pas bien le zoom

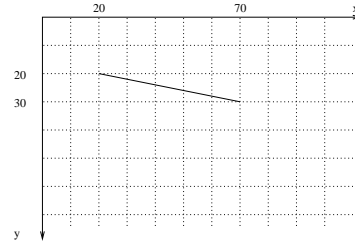


- Vectoriel : image décrite par des points et des zones du plan
- Meilleure résistance au zoom ; plus modulaire.

13

Image vectorielle

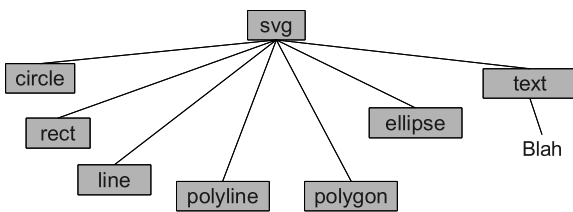
- Le trait va être caractérisé par son épaisseur et ses extrémités (20, 20) et (70, 30).



14

Format SVG

- Application XML.
- Nom qualifié : <http://www.w3.org/2000/svg>
- Exemple d'arbre SVG (sans attribut) :



15

Attributs SVG

- Pour les formes et le texte:

- Positions et longueurs :

```
<circle cx="70" cy="100" r="50" />
<rect x="150" y="50" width="135" height="100" />
<line x1="325" y1="150" x2="375" y2="50" />
<polyline points="50,250 75,350 100,250 125 350" />
<polygon points=" 250,250 297,284 279, 340" />
<ellipse cx="400" cy="300" rx="72" ry="50" />
<text x="100" y="100">Hello</text>
```

- Trait et couleurs :

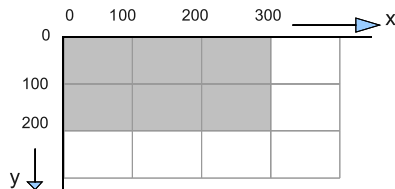
- stroke : couleur du trait
- fill : couleur de l'intérieur de la forme
- stroke-width : épaisseur du trait

16

Attributs SVG (suite)

- Pour la racine :
 - viewBox : la plage du plan (ex : "x1 y1 x2 y2")
 - width : la largeur désirée (ex : "2cm")
 - height : la hauteur désirée (ex : "4cm")

```
<svg viewBox="0 0 200 300" width="4cm"> ... </svg>
```

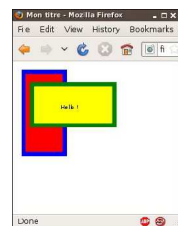


17

Exemple : rectangles

- Dans un fichier, extension .xml ou .svg
- Avec espace de nom

```
<svg viewBox="0 0 300 300"
width="4cm"
xmlns="http://www.w3.org/2000/svg">
<rect
fill="red"
stroke="blue"
stroke-width="10"
x="10" y="10"
width="100"
height="200"/>
<rect
fill="yellow"
stroke="green"
stroke-width="10"
x="30" y="40"
width="200"
height="100"/>
<text x="100" y="100">Hello!</text>
</svg>
```



18

Dans un fichier seul

- Extension : .svg ou .xml
- Lisible directement par firefox.
- Internet explorer a besoin d'un plugin.

```
<?xml version="1.0" encoding="utf-8"?>
<svg xmlns="http://www.w3.org/2000/svg"
  viewBox="0 0 300 300" width="4cm">
  <rect fill="red" stroke="blue" stroke-width="10"
    x="10" y="10" width="100" height="200"/>
  <rect fill="yellow" stroke="green" stroke-width="10"
    x="30" y="40" width="200" height="100"/>
  <text x="100" y="100">Hello!</text>
</svg>
```

19

Inclusion dans un fichier XHTML

- Extension du fichier : toujours .xhtml
- Utilisation des espaces de nom

```
<?xml version="1.0" encoding="utf-8"?>
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>Mon titre</title>
  </head>
  <body>
    <svg xmlns="http://www.w3.org/2000/svg"
      viewBox="0 0 300 300" width="4cm">
      <rect fill="red" stroke="blue" stroke-width="10"
        x="10" y="10" width="100" height="200"/>
      <rect fill="yellow" stroke="green" stroke-width="10"
        x="30" y="40" width="200" height="100"/>
      <text x="100" y="100">Hello!</text>
    </svg>
  </body>
</html>
```

20

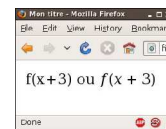
Avec des préfixes...

```
<?xml version="1.0" encoding="utf-8"?>
<html:html xmlns:html="http://www.w3.org/1999/xhtml">
  <html:head>
    <html:title>Mon titre</html:title>
  </html:head>
  <html:body>
    <svg:svg xmlns:svg="http://www.w3.org/2000/svg"
      viewBox="0 0 300 300" width="4cm">
      <svg:rect fill="red" stroke="blue" stroke-width="10"
        x="10" y="10" width="100" height="200"/>
      <svg:rect fill="yellow" stroke="green" stroke-width="10"
        x="30" y="40" width="200" height="100"/>
      <svg:text x="100" y="100">Hello!</svg:text>
    </svg:svg>
  </html:body>
</html:html>
```

21

MathML

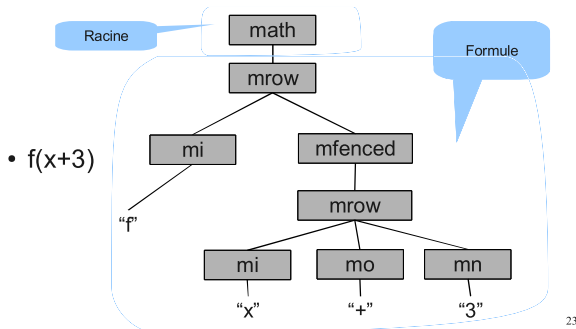
- Encoder des mathématiques dans du texte
 - Pas juste une suite de lettres
 - Exposants
 - Fractions
 - Racines
 - Espacement entre les symboles particulier suivant leur sens.



22

Idée derrière MathML

- Concevoir une formule comme un arbre :



23

La notion de formule

- Une formule est un arbre, attaché à la racine.
- Pour accoler plusieurs éléments en ligne:


```
<mrow> ... </mrow>
```
- Les formules élémentaires sont :
 - Les identifiants (des variables) :


```
<mi>x</mi> <mi>f</mi> <mi>y</mi>
```
 - Les opérateurs :

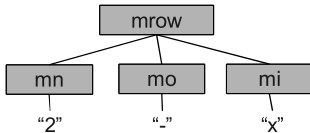

```
<mo>+</mo> <mo>-</mo> <mo>&middot;</mo>
          <mo>=</mo> <mo>&lt;</mo> ...
```
 - Les nombres :


```
<mn>1234</mn>
```

24

Exemple

```
<mrow>
  <mn>2</mn>
  <mo>-</mo>
  <mi>x</mi>
</mrow>
```



25

Formules complexes

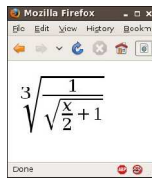
- Fractions :
`<mfrac>` numérateur dénominateur `</mfrac>`
- Racine :
`<msqrt>` radical `</msqrt>`
`<mroot>` radical exposant `</mroot>`
- Exposant :
`<msup>` formule exposant `</msup>`
- Parenthèses :
`<mfenced>` formule `</mfenced>`

26

Exemple

- On veut représenter la formule

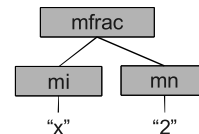
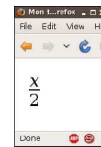
$$\sqrt[3]{\frac{1}{\sqrt{\frac{x}{2}+1}}}$$



27

Exemple

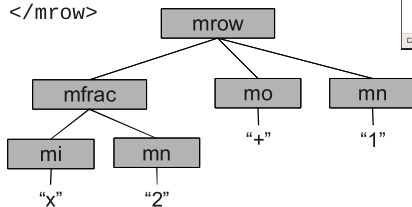
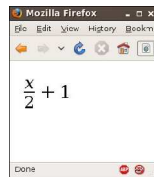
```
<mfrac>
  <mi>x</mi>
  <mn>2</mn>
</mfrac>
```



28

Exemple

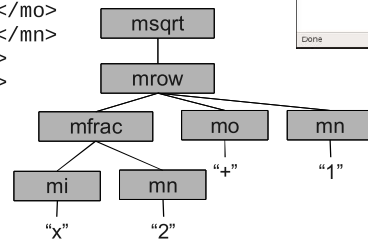
```
<mrow>
  <mfrac>
    <mi>x</mi>
    <mn>2</mn>
  </mfrac>
  <mo>+</mo>
  <mn>1</mn>
</mrow>
```



29

Exemple

```
<msqrt>
  <mrow>
    <mfrac>
      <mi>x</mi>
      <mn>2</mn>
    </mfrac>
    <mo>+</mo>
    <mn>1</mn>
  </mrow>
</msqrt>
```



30

Exemple

```

<mfrac>
  <mn>1</mn>
  <msqrt>
    <mrow>
      <mfrac>
        <mi>x</mi>
        <mn>2</mn>
      </mfrac>
      <mo>+</mo>
      <mn>1</mn>
    </mrow>
  </msqrt>
</mfrac>

```

The diagram shows the tree structure of the MathML expression. The root is `mfrac`, which branches into `mn` (value "1") and `msqrt`. `msqrt` branches into `mrow`, which further branches into `mfrac`, `mo` ("+"), and `mn` ("1"). The inner `mfrac` branches into `mi` ("x") and `mn` ("2").

Exemple

```

<root>
  <mfrac>
    <mn>1</mn>
    <msqrt>
      <mrow>
        <mfrac>
          <mi>x</mi>
          <mn>2</mn>
        </mfrac>
        <mo>+</mo>
        <mn>1</mn>
      </mrow>
    </msqrt>
  </mfrac>
  <mn>3</mn>
</root>

```

The diagram shows the tree structure of the MathML expression. The root is `mfrac`, which branches into `mn` ("1") and `msqrt`. `msqrt` branches into `mrow`, which further branches into `mfrac`, `mo` ("+"), and `mn` ("1"). The inner `mfrac` branches into `mi` ("x") and `mn` ("2"). The root also branches into `mn` ("3").

Un fichier MathML

- Un document xml
- Element racine `$...$`
- Espace de nom : <http://www.w3.org/1998/Math/MathML>

```

<?xml version="1.0"?>
<math xmlns="http://www.w3.org/1998/Math/MathML">
  <root>
    <mfrac>
      <mi>1</mi>
      <msqrt>
        <mrow>
          <mfrac>
            <mi>x</mi>
            <mn>2</mn>
          </mfrac>
          <mo>+</mo>
          <mn>1</mn>
        </mrow>
      </msqrt>
    </mfrac>
    <mn>3</mn>
  </root>
</math>

```

Du XHTML et du MathML

```

<?xml version="1.0"?>
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:m="http://www.w3.org/1998/Math/MathML">
  <head><title>Des mathématiques</title></head>
  <body>
    <h1>Une équation</h1>
    <p>
      <math>
        <mi>1</mi>
        <mo>+</mo>
        <mn>1</mn>
      </math>
      alors la formule
      <math>
        <mi>3</mi>
        <msqrt>
          <mrow>
            <mfrac>
              <mi>x</mi>
              <mn>2</mn>
            </mfrac>
            <mo>+</mo>
            <mn>1</mn>
          </mrow>
        </msqrt>
      </math>
      se simplifie pour donner
      <math>
        <mi>1</mi>
      </math>
    </p>
  </body>
</html>

```

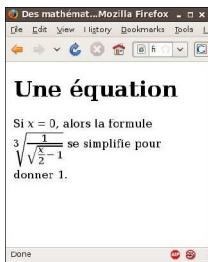
The diagram shows the tree structure of the XHTML document. The root is `html`, which branches into `head` and `body`. `head` contains `title` ("Des maths"). `body` contains `h1` ("Une équation") and `p`. The `p` element contains several `m:math` elements and text: "alors la formule", "se simplifie pour", and "donner".

Du XHTML et du MathML

```

<?xml version="1.0"?>
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:m="http://www.w3.org/1998/Math/MathML">
  <head><title>Des mathématiques</title></head>
  <body>
    <h1>Une équation</h1>
    <p>
      <math>
        <mi>1</mi>
        <mo>+</mo>
        <mn>1</mn>
      </math>
      alors la formule
      <math>
        <mi>3</mi>
        <msqrt>
          <mrow>
            <mfrac>
              <mi>x</mi>
              <mn>2</mn>
            </mfrac>
            <mo>+</mo>
            <mn>1</mn>
          </mrow>
        </msqrt>
      </math>
      se simplifie pour donner
      <math>
        <mi>1</mi>
      </math>
    </p>
  </body>
</html>

```



Flux de données : RSS

- Plusieurs définitions :
 - Really Simple Syndication
 - Rich Site Summary
- Pour un flux d'informations :
 - Sites d'information
 - Liste de contenu
 - Blogs
 - Nouveautés sur un site
- Pas d'espace de nom par défaut.

Format RSS

- Exemple :

```
<rss>
<channel>
<title>Liste des leçons</title>
<item>
  <title>XML et DTDs</title>
  <pubDate>Semaine 38</pubDate>
  <description> ... </description>
</item>
<item>
  <title>Espaces de noms</title>
  <pubDate>Semaine 39</pubDate>
  <description> ... </description>
</item>
</channel>
</rss>
```

- DTD (simplifiée):

```
<!ELEMENT rss (channel+)>
<!ELEMENT channel
  (title, description?, item*)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT description (#PCDATA)>
<!ELEMENT item
  (title|pubDate|description)*>
<!ELEMENT pubDate (#PCDATA)>
```

37

Les métadonnées.

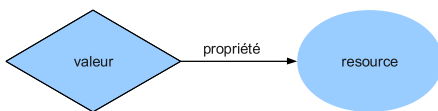
- Problème classique : Ajout dans un XML d'informations qui ne sont pas du contenu.

- Auteur(s)
- Date de création, de mise à jour
- Description
- Lien externe
- ...

38

Format RDF

- Modèle de donnée simple :



```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Description about="http://inf356.monoidal.net">
    <author>Benoît Valiron</author>
  </rdf:Description>
</rdf:RDF>
```

Quel langage ?

- Chaque sous-élément de Description est une propriété qui contient une valeur textuelle ou XML.

39

Format Dublin Core

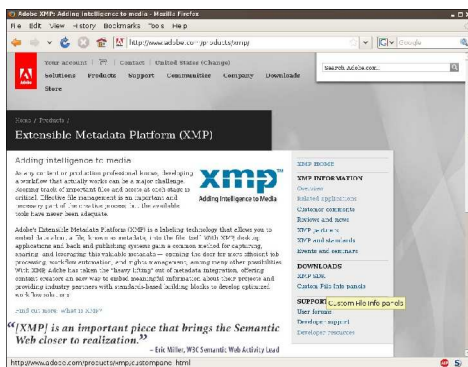
- Donner une structure standardisée à des métadonnées
- En particulier aux propriétés de RDF.
- Pas d'arbre par défaut, juste un ensemble de 15 éléments avec un sens bien défini

- | | |
|---------------|--------------|
| • title | • format |
| • creator | • identifier |
| • subject | • source |
| • description | • language |
| • publisher | • relation |
| • contributor | • coverage |
| • date | • rights |
| • type | |

40

Méta-données dans un PDF

Adobe utilise un format de métadonnées insérées dans les PDF : le format XMP.



41

Du contenu XML dans un PDF

```
<?xpacket begin="ï»¿id="W5M0MpCehhHzreSzNTczkc9d"?">
<X:xmpmeta xmlns:x="adobe:meta/">
  <x:xmpk="Adobe XMP Core 4.0-c316 44.253921, Sun Oct 01 2006 17:08:23" />
</X:xmpmeta>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Description rdf:about="" xmlns:xap="http://ns.adobe.com/xap/1.0/">
    <xap:CreateDate>2009-07-10T11:03:46+02:00</xap:CreateDate>
    <xap:ModifyDate>2009-07-10T11:03:46+02:00</xap:ModifyDate>
    <xap:MetadataDate>2009-07-10T11:03:46+02:00</xap:MetadataDate>
    <xap:CreatorTool>Vector Data Adobe Photoshop 7.0</xap:CreatorTool>
  </rdf:Description>
  <rdf:Description rdf:about="" xmlns:dc="http://purl.org/dc/elements/1.1/">
    <dc:format>application/pdf</dc:format>
  </rdf:Description>
  <rdf:Description rdf:about=""
    xmlns:xapMM="http://ns.adobe.com/xap/1.0/mm/">
    <xapMM:DocumentID>uuid:d1d6b8c-3219-44f0-b663-460ea508aaf7</xapMM:DocumentID>
    <xapMM:InstanceID>uuid:fd9e6ee6-1963-436f-b908-8ddc64f92924</xapMM:InstanceID>
  </rdf:Description>
  <rdf:Description rdf:about="" xmlns:pdf="http://ns.adobe.com/pdf/1.3/">
    <pdf:Producer>Adobe Photoshop for Windows</pdf:Producer>
  </rdf:Description>
</rdf:RDF>
</X:xmpmeta>
```

42

```

<?xpacket begin="" id="w5M0MpCehiHzreSzNTczkc9d"?>
<x:xmpmeta xmlns:x="adobe:meta/" x:xmpk="3.1-701">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
<rdf:Description rdf:about=""
xmlns:pdf="http://ns.adobe.com/pdf/1.3/">
<pdf:Producer>Acrobat Distiller 7.0 (Windows)</pdf:Producer>
</rdf:Description>
<rdf:Description rdf:about="" xmlns:xap="http://ns.adobe.com/xap/1.0/">
<xap:CreateDate>2009-06-16T09:53:18+02:00</xap:CreateDate>
<xap:CreatorTool>PScript5.dll Version 5.2.2</xap:CreatorTool>
<xap:ModifyDate>2009-06-16T09:53:18+02:00</xap:ModifyDate>
</rdf:Description>
<rdf:Description rdf:about=""
xmlns:dc="http://purl.org/dc/elements/1.1/">
<dc:format>application/pdf</dc:format>
<dc:title>
<rdf:Alt>
<rdf:li xml:lang="x-default">Microsoft PowerPoint -
090615 DP PTTN Lancement Nord Correntin</rdf:li>
</rdf:Alt>
</dc:title>
<dc:creator>
<rdf:Seq>
<rdf:li>hfarnham</rdf:li>
</rdf:Seq>
</dc:creator>
</rdf:Description>
<rdf:Description rdf:about="" xmlns:xapMM="http://ns.adobe.com/xap/1.0/mm/">
<xapMM:DocumentID>uuid:15594a7b-a8a6-43d3-9b98-481e4015c8e2</xapMM:DocumentID>
<xapMM:InstanceID>uuid:51799985-3abe-4c00-81e6-0167593cd477</xapMM:InstanceID>
</rdf:Description>
</rdf:RDF>
</x:xmpmeta>

```

Méta-données ailleurs ?

- Dans du RSS : Exemple au TD-2/TP-2
- Dans du XHTML :
 - Pas de standard.
 - Balise meta dans l'entête
 - Inclusion avec RDF dans l'entête : casse la DTD.
- Pour donner un sens à tout fichier de données non-standard.