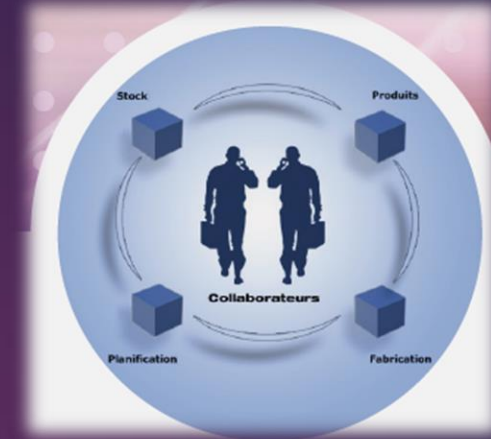


Université de BATNA 2
Département d'informatique



Les données semi-structurées

L3 ISIL

« *Les données semi-structurées* »

Chapitre 2 : PART II

' Modélisation + espaces de noms '

Djennane A,
djennane.am@gmail.com

2020-2021

I. Les scénarios d'utilisation XML

1

Sauvegarde de Données

...

Échanges (AJAX, SOAP,...)

Données : clients, notes, ...

multimédia : livre, hypertexte,

Non-Structurées

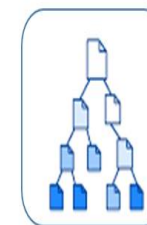
SGBDR

XML native

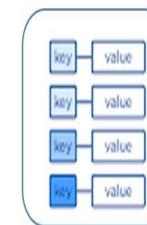
➤ NoSQL (JSON)

Sous-ensemble

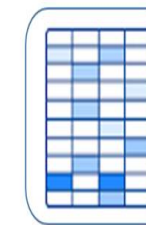
XML : XHTML(textuel), MathML, SMIL, SVG, ...



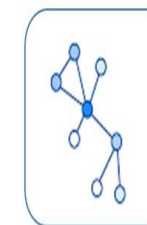
Document Store



Key-Value Store

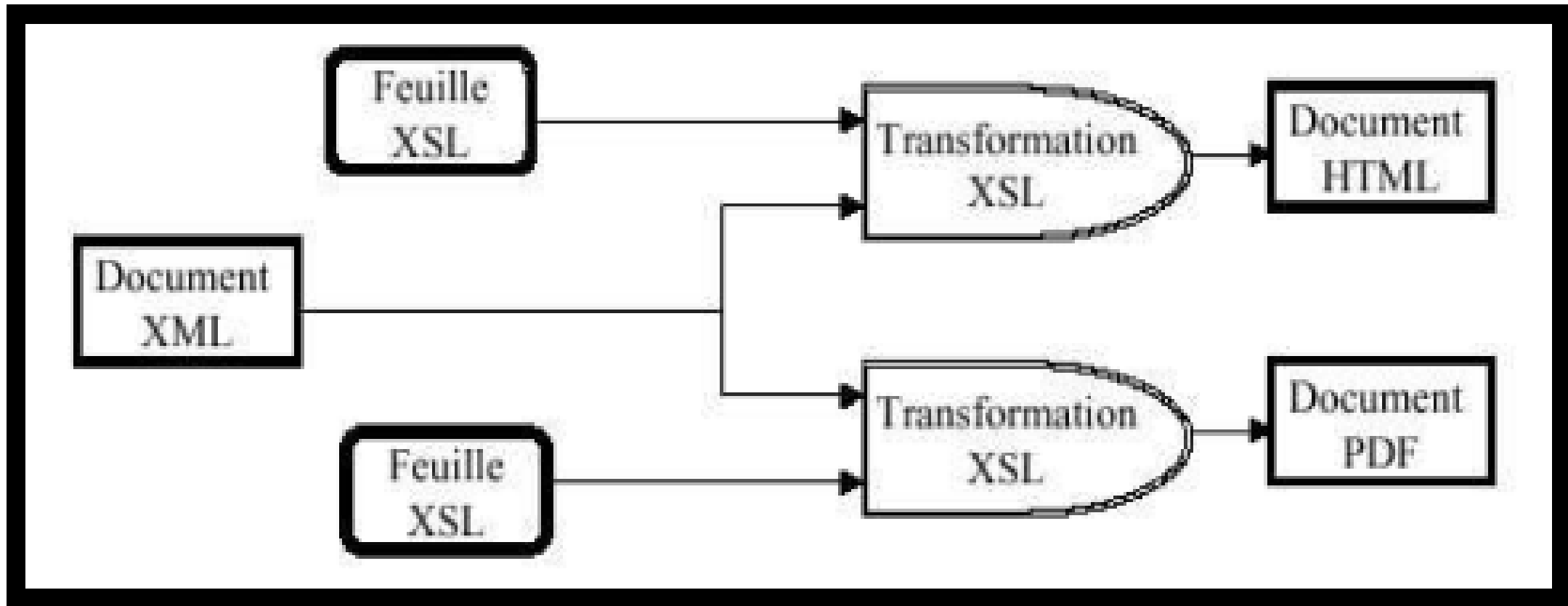


Wide-Column Store



Graph Store

I.a La transformation XML



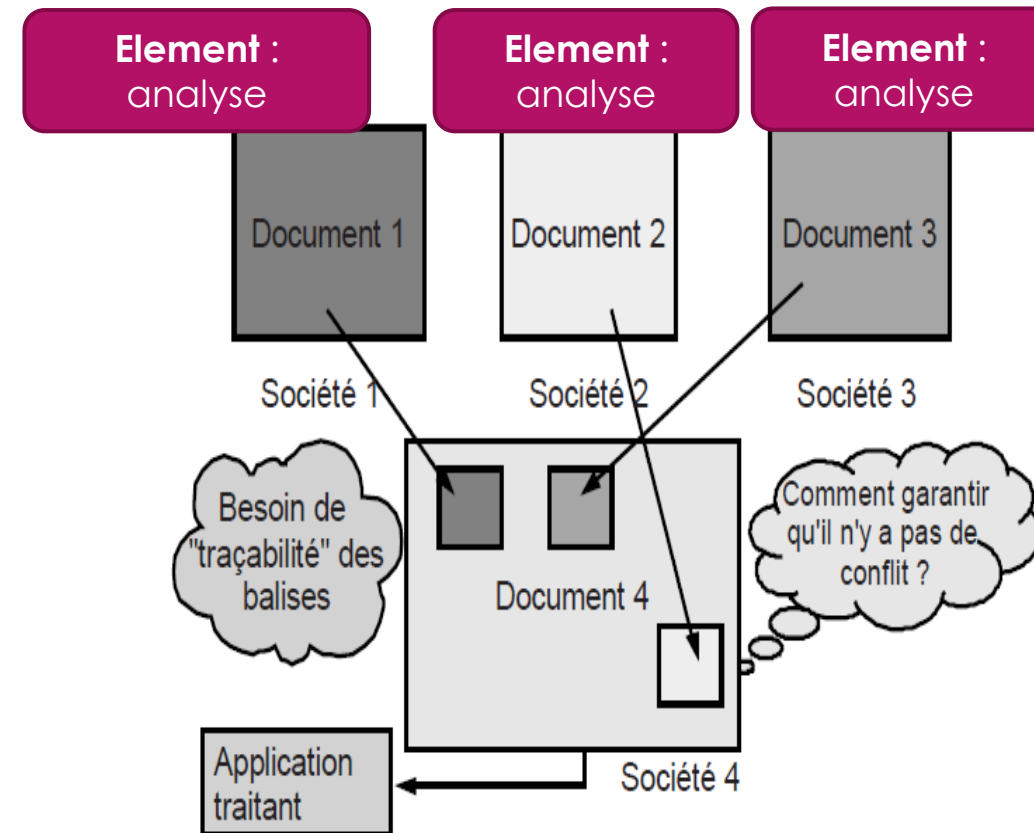
II. La modélisation XML

- Les espaces de noms,
- XML et UML (modélisation de BD).

11.1 Les espaces de noms (Name Spaces) : Résoudre les conflits d'utilisation des balises

- La société 4 **intègre** au document 4, des documents proviennent des sociétés 1, 2 et 3.
- Chaque document définit ces **propres** balises et attributs.
- **Problème**: Comment **interpréter** l'élément **température** dans l'élément **analyse** => température de la salle, du corp humain, ou du processeur ?)
- **Solution** => classer chaque concept (la porté) dans une classe (espace des noms), pour savoir comment il doit être **interprété**

Figure 2-1
Croisement
de documents



```
<table>
  <tr>
    <td>Apples</td>
    <td>Bananas</td>
  </tr>
</table>
```

This XML carries information about a table (a piece of furniture):


```
<table>
  <name>African Coffee Table</name>
  <width>80</width>
  <length>120</length>
</table>
```

If these XML fragments were added together, there would be a name conflict. Both contain a `<table>` element, but the elements have different content and meaning.

A user or an XML application will not know how to handle these differences.

Espaces de nommage

Un espace de nommage (*namespace*) définit une famille de noms afin d'éviter les confusions entre des éléments qui auraient le même nom mais pas le même sens. Cela arrive quand le document XML modélise les informations de plusieurs domaines.

Voici un exemple dans le domaine de la vente de meubles. Le document modélise une table (avec 4 pieds) et aussi un tableau HTML pour afficher ses dimensions. On voit la confusion. 

```
<meuble id="765">
  <table prix="74,99€">acajou</table>
  <table border="1">
    <tr><th>longueur</th><th>largeur</th></tr>
    <tr><td>120cm</td><td>80cm</td></tr>
  </table>
</meuble>
```

Les espaces de noms

Les espaces de noms sont un concept très commun en informatique. Nous les retrouvons dans de nombreux langages de programmation afin de prévenir d'éventuels conflits. Par exemple, dans le langage de programmation Java, les *packages* servent à délimiter la portée d'une classe. En choisissant un package, le développeur lève tout conflit potentiel sur le nom, car la classe sera par la suite utilisée directement ou indirectement via son package et son nom.

Le design de l'espace des noms

- La notion d'espace de noms peut être perçue comme un groupe d'appartenance ou une **famille**.
- L'utilisation des espaces de noms garantit une forme de **traçabilité** de la balise et évite les **ambiguïtés d'usage**.
- Pour que les espaces de noms aient un sens, il faut pour chacun d'eux un **identifiant unique (URI)**.



The diagram consists of a large orange oval containing a smaller blue oval. The orange oval is labeled 'URI (Uniform Resource Identifier)' with the example 'Your Name'. The blue oval is labeled 'URL (Uniform Resource Locator)' with the example 'Your Home Location'. Two vertical blue lines are positioned on the left and right sides of the orange oval, and a purple square is in the top right corner of the slide.

URI
(Uniform Resource Identifier)

Example:
Your Name

URL
(Uniform Resource Locator)

Example:
Your Home Location

Utilisation des espaces de noms dans un document XML

10

L'espace de noms par défaut

```
<chapitre xmlns="http://www.masociete.com">
```

```
  <paragraphe>
```

```
  ...
```

```
  </paragraphe>
```

```
</chapitre>
```

URI (une adresse fictive qui n'as aucun existence)

Ici l'élément `chapitre` est dans l'espace de noms `http://www.masociete.com`. C'est également le cas de l'élément `paragraphe`, puisqu'il est dans l'élément `chapitre`.

Attention

Un espace de noms par défaut ne concerne que les éléments. Les attributs et les textes n'y appartiennent pas. Le texte d'un élément n'est jamais dans un espace de noms puisqu'il représente la donnée.

Espace de noms imbriqués

```
<chapitre xmlns="http://www.masociete.com">  
  <paragraphe xmlns="http://www.autresociete.com">  
    ...  
  </paragraphe>  
</chapitre>
```

L'élément paragraphe n'appartient pas à l'espace de noms `http://www.masociete.com` mais uniquement à l'espace de noms `http://www.autresociete.com`.

L'espace de noms par défaut présente l'inconvénient d'être peu contrôlable sur un document de taille importante. En effet, tout ajout ou modification d'un tel espace va se répercuter sur la totalité du contenu.

L'espace de nom explicite

1. On définit un **préfixe** (EspaceDeNom), avec l'attribut **xmlns**.
2. On associe ce préfixe à une **URI** (qui peut être **fictive** mais qui représente souvent l'organisme auteur de l'espace de nom).
3. On fait ensuite précédé chaque élément concerné, de ce préfixe.

```
<EspaceDeNom:Elément xmlns:EspaceDeNom="URI">donnée</EspaceDeNom:Elément>
```

```
<?xml version="1.0" encoding="ISO-8859-1" ?>  
<fact:facture xmlns:fact="http://www.domaine.com/facture">  
  <fact:montant>10$</fact:montant>  
  <fact:nom>Jean</fact:nom>  
</fact:facture>
```

Combinaison des espaces

On peut déclarer et utiliser plusieurs espaces de noms grâce aux préfixes.

Exemple :

```
<p:res xmlns:p="http://www.masociete.com" xmlns:p2="http://www.autresociete.com">  
  <p2:res>  
  </p2:res>  
</p:res>
```

Le premier élément `res` est dans l'espace de noms `http://www.masociete.com` alors que l'élément `res` à l'intérieur est dans l'espace de noms `http://www.autresociete.com`.

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
<message xmlns='http://etudiant.univ-mlv.fr/~gcharpen/xml/message'
xmlns:autre="http://www.autre.com/namespace">
  <expediteur>
    <identite>
      <prenom>Guillaume</prenom>
      <nom>Charpentier</nom>
      <email>gcharpen@etudiant.univ-mlv.fr</email>
    </identite>
  </expediteur>
  <destinataire>
    <identite>
      <prenom>Gérard</prenom>
      <nom>Dupont</nom>
      <email>gerarddupont@provider.com</email>
    </identite>
  </destinataire>
  <texte>mon message</texte>
  <autre:titre>titre du texte</autre:titre>
  <autre:texte>ceci est un texte</autre:texte>
</message>
```

Combinaison entre espace de noms
implicite et explicite

Namespaces can also be declared in the XML root element:

```
<root xmlns:h="http://www.w3.org/TR/html4/"
xmlns:f="https://www.w3schools.com/furniture">

<h:table>
  <h:tr>
    <h:td>Apples</h:td>
    <h:td>Bananas</h:td>
  </h:tr>
</h:table>

<f:table>
  <f:name>African Coffee Table</f:name>
  <f:width>80</f:width>
  <f:length>120</f:length>
</f:table>

</root>
```

Exemple : code Android

```
<?xml version="1.0" encoding="utf-8" ?>
<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
```

```
<LinearLayout
    android:id="@+id/ll_buttons"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="10dp"
    android:background="@color/colorPrimary"
    android:orientation="horizontal"
    android:padding="10dp"
    tools:layout_editor_absoluteX="10dp"
    tools:layout_editor_absoluteY="10dp">

    <Button
        android:id="@+id/button5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="5dp"
        android:layout_weight="1"
        android:background="@color/colorAccent"
```



Autre exemple introduisant des éléments MathML :

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
  <body>
    <p>Formule mathématique</p>
    <math xmlns="http://www.w3.org/1998/Math/MathML">
      <apply> <log/>
        <logbase>
          <cn> 3 </cn>
        </logbase>
        <ci> x </ci>
      </apply>
    </math>
  </body>
</html>
```

Outre les éléments XHTML, nous trouvons des éléments MathML dont l'espace de noms est <http://www.w3.org/1998/Math/MathML>.

L'exemple précédent

```
<?xml version="1.0" encoding="utf-8"?>
<meuble:meuble id="765"
  xmlns:meuble="urn:iutlan:meubles"
  xmlns:html="http://www.w3.org">
  <meuble:table prix="74,99€">acajou</meuble:table>
  <html:table border="1">
    <html:tr><html:th>longueur</html:th>...</html:tr>
    <html:tr><html:td>120cm</html:td>...</html:tr>
  </html:table>
</meuble:meuble>
```

Conclusion

- Pensez toujours à attacher votre document XML à un espace de noms pour résoudre l'ambiguïté.

Travail de recherche

1. Comment utiliser **les espaces de noms dans les attributs** ?
2. Les protocoles de transport de **flux d'informations** (podcast, news, blogs):
 - **RSS, ATOM.**
 - Remarque - XML n'est pas le seul format de données utilisé pour l'échange d'informations entre systèmes. **JSON** est aussi souvent utilisé.