L' approche Xweb

Dans cette partie nous étudions un modèle de conception d'applications Web basées sur XML (Xweb) proposé par Lo et Hocine [19].

Plusieurs travaux ont été menés dans le but de proposer des méthodologies et des modèles de conception d'applications Web. On peut distinguer les méthodologies de conception d'applications hypermédia et les systèmes de gestion de sites Web dynamiques.

Les premiers travaux réalisés dans le domaine avaient pour but de résoudre les problèmes liés aux systèmes hypermédias. Il s'agissait de trouver des méthodologies adaptées à ce type d'applications qui ont des spécificités par rapport aux applications traditionnelles. Toutes ces méthodologies sont fondées sur la séparation entre l'analyse du domaine, la spécification de l'espace navigationnel et la conception de l'interface utilisateur.

Les solutions proposées dans ces travaux peuvent être adaptées au contexte du Web et ont inspiré la plupart des travaux menés pour la conception d'applications Web. En plus des problèmes liés au contexte hypermédia, d'autres problèmes liés à la spécificité du Web se posent au concepteur d'application Web : intégration de données de sources diverses, interopérabilité, nature dynamique du Web, nécessité de couplage avec des SGBD, etc. C'est dans ce cadre que plusieurs systèmes ont vu le jour. On peut citer Strudel[7], Araneus[11], WebML[4].

Tous ces systèmes (Strudel, Araneus, WebML) sont basés sur HTML et s'appuient sur la notion de médiateur. Un médiateur est un module logiciel qui exploite de la connaissance codée sur quelques ensembles et sous ensembles de données afin de créer une information pour une couche applicative de niveau supérieur.

Les systèmes de médiation traitent de l'intégration de données issues de sources hétérogènes. Ils permettent l'exploitation de plusieurs sources d'information à partir d'une vue intégrée de données.

Cependant, ils utilisent une approche virtuelle et ne sont pas très adaptés à certains types d'application comme les systèmes d'information environnementaux qui présentent la spécificité de ne pas nécessiter une mise à jour régulière de leurs données. En effet, la réalisation de ce type d'applications nécessite plus des solutions permettant la construction de bases d'information exploitables à travers le Web que de solutions de médiation.

Dans **l'approche virtuelle**, les données sont laissées dans leurs sources respectives sans être dupliquées et les requêtes sont décomposées et appliquées en temps réel sur les sources. Cette forme d'intégration ne facilite :

- ni la mise en forme des données intégrées pour leur diffusion sur le Web,
- ni l'ajout d'un module de recherche d'informations pertinentes.

En plus, la définition du schéma intégré et sa liaison avec les schémas locaux sont des difficultés liées à cette approche.

Par ailleurs, les avantages offerts par le langage XML comme format de représentation et d'échange de données sur le Web font de ce standard un outil incontournable pour la réalisation de systèmes d'information pour le Web. Il faut dire que cela offre de nombreux avantages[20]. D'abord, XML permet une représentation naturelle de données provenant de n'importe quelle source. Il offre la possibilité de développer des adaptateurs permettant de transformer toute donnée source ayant un format quelconque en XML. Ensuite, XML facilite l'interopérabilité entre les différents composants d'un système d'intégration en offrant une syntaxe de données standard.

Dans cette section nous nous focalisons sur la présentation des deux modèles principaux (structurel et médiatique), autour desquels s'articule la méthode de conception et de

développement d'applications Web. Cette méthode s'appuie sur les nouveaux standards XML (eXtensible Markup Language) [2] et sur la notion de dataweb, un dataweb étant une collection de données (i) structurées comme c'est le cas de celles qui sont stockées dans les bases de données relationnelles ou objets, ou (ii) semi-structurées comme celles qui sont disponibles sur le web [9].

1.Le modèle structurel

La démarche de conception du modèle structurel qu'ils proposent s'effectue en plusieurs étapes :

- Identifier les sources de données (structurées et semi-structurées) du dataweb,
- Utiliser le modèle entité-association (E-A) pour modéliser chaque source de données structurées, et XML pour chaque source de données semi-structurées [1],
- Traduire les modèles E-A obtenus à l'étape précédente en schémas relationnels,
- Obtenir pour chaque source, grâce à une technique de mapping (relationnel – XML) qui est détaillée dans [9], un schéma structurel représentant la structure logique de la base de données relationnelle en XML.
- Insérer dans une base de documents XML les données issues des bases de données relationnelles.

L'ensemble des schémas structurels issus des différentes sources de données structurées du dataweb, et les schémas des documents XML issus des sources de données semi-structurées, forment le modèle structurel du dataweb.

Pour illustrer simplement l'approche, nous avons utilisé Xweb pour réaliser le site web d'une centrale d'achat. Nous avons considéré deux sources de données :

- les informations sur les fournisseurs et les produits de la centrale d'achat Ndakarou khéweul, modélisées à l'aide du modèle E/A (données structurées) ;
- les informations relatives aux services web décrites dans un document texte, formalisées dans un document XML (données semi-structurées). Rappelons que les services web ne sont pas pris en compte dans l'approche proposée par Lo et Hocine.

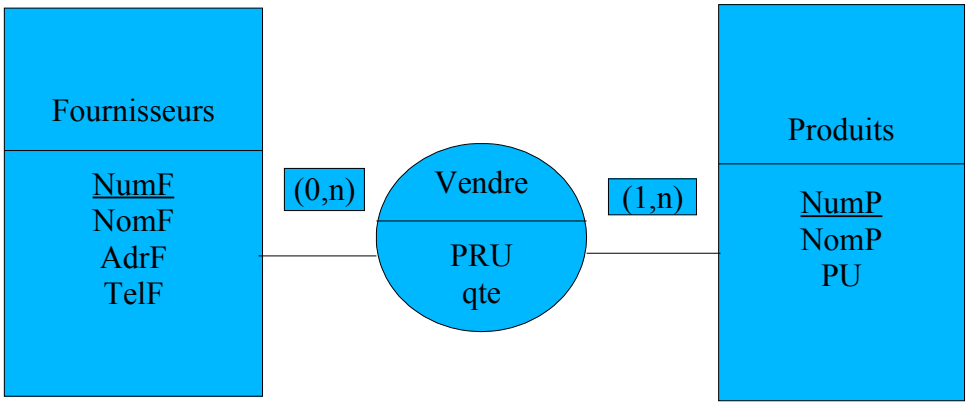

Figure 1 : Diagramme du schéma entité-association de la base de données de la centrale d'achat Ndakarou khéweul

Le modèle structurel du dataweb est donné ci-dessous :

```
<bd_centrale_achat>
    <fournisseurs>
     <fournisseur id="">
        <nom/>
        <adresse/>
        <telephone/>
     </fournisseur>
    </fournisseurs>
    <produits>
     <produit id="">
        <nom/>
        <poids_unitaire/>
     </produit>
    </produits>
    <ventes>
     <vente fournisseur="" produit="">
        <prix_unitaire/>
        <quantite/>
     </vente>
    </ventes>
    <services_web>...</services_web>
</bd_centrale_achat>
```

Le problème de la transformation (ou mapping) des données issues de bases de données relationnelles en documents XML a donné lieu à quelques outils dont DB2XML [15], PLSXML [12] et XML-DBMS [3]. Ces trois systèmes sont des packages Java et ont adopté la même approche : ils utilisent une technique de mapping très simple basée sur le DOM (Document Object Model) [18]. Le problème des clés étrangères n'est pas abordé, ce qui implique une redondance d'informations dans le document obtenu. La structure de ce document est une structure d'arbre et non de graphe orienté comme celle que nous avons proposée (Fig. 2).

XML autorise l'utilisation des types spéciaux d'attributs pour traduire la notion de graphe : un élément peut avoir un attribut de type ID dont la valeur fournit un identificateur unique qui peut être référencé par des attributs de type IDREF ou IDREFS à partir d'autres éléments. Notre modèle de données supporte les attributs IDREF(s) et permet donc de représenter les données XML par un graphe orienté et étiqueté : les attributs IDREF(s) implémentent les associations du modèle E/A.

Exemple de données XML :

Nous utilisons un extrait simplifié de la base de documents XML obtenue après l'opération de mapping :

```
<bd_centrale_achat>
 <fournisseurs>
  <fournisseur id="BL">
   <nom>SBMA</nom>
   <adresse>Lot N4 EST SODIDA DAKAR</adresse>
   <telephone>849-56-06</telephone>
 </fournisseur>
 <fournisseur id="SAT">
   <nom>SATREC</nom>
   <adresse>DAKAR-SENEGAL BP 22130</adresse>
   <telephone>828-14-33</telephone>
 </fournisseur>
 <fournisseur id="KIR">
   <nom>KIRENE</nom>
   <adresse>BP 7020 DAKAR-SENEGAL</adresse>
   <telephone>956-22-97</telephone>
 </fournisseur>
 <fournisseur id="FONT">
   <nom>FONTAINE</nom>
   <adresse>DAKAR-SENEGAL</adresse>
   <telephone>828-45-96</telephone>
 </fournisseur>
 <fournisseur id="COL">
   <nom>COLGATE PALMOLIVE</nom>
   <adresse>DAKAR-SENEGAL</adresse>
   <telephone>8825-15-16</telephone>
 </fournisseur>
 <fournisseur id="MED">
   <nom>MEDIANE</nom>
   <adresse>Dakar-SENEGAL</adresse>
   <telephone>825-18-26</telephone>
 </fournisseur>
 <fournisseur id="CRO">
   <nom>CROCO</nom>
   <adresse>RUFISQUE</adresse>
   <telephone>848-28-16</telephone>
 </fournisseur>
</fournisseurs>
<produits>
```

```
- <produit id="EKYR">
    <nom>Eau Minerale KIRENE</nom>
    <poids_unitaire>330g</poids_unitaire>
  </produit>
- <produit id="EFONT">
    <nom>Eau Minerale FONTAINE</nom>
    <poids_unitaire>500g</poids_unitaire>
  </produit>
- <produit id="TCOL">
    <nom>Pate dentifrice COLGATE</nom>
    <poids_unitaire>120g</poids_unitaire>
  </produit>
- <produit id="TMED">
    <nom>Pate dentifrice MEDIANE</nom>
    <poids_unitaire>120g</poids_unitaire>
  </produit>
- <produit id="SCRO">
    <nom>Savon CROCO</nom>
    <poids_unitaire>900g</poids_unitaire>
  </produit>
- <produit id="LSAT">
    <nom>Lait en Poudre (VITALAIT)</nom>
    <poids_unitaire>22.5g</poids_unitaire>
  </produit>
- <produit id="LBL">
    <nom>Lait en Poudre (BARALAIT)</nom>
    <poids_unitaire>22.5g</poids_unitaire>
  </produit>
 </produits>
 ...
</bd_centrale_achat>
```

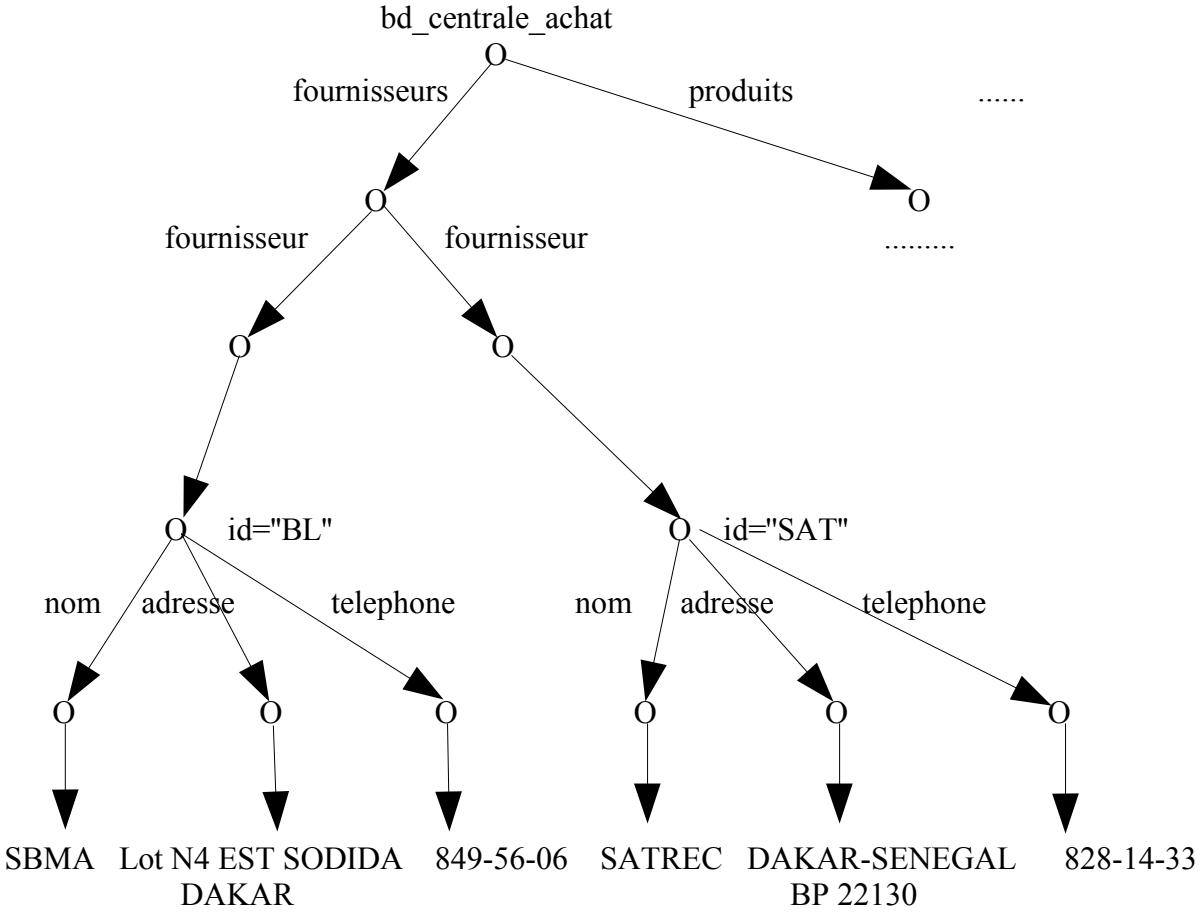

Figure2: Exemple de graphe XML représentant une partie du document XML ci-dessus

2.Le modèle médiatique

Le modèle médiatique est une vue du modèle structurel. Il décrit les unités médiatiques du dataweb et leur structure navigationnelle.

Une unité médiatique (U.M.) est une «unité informative qui a une certaine autonomie du point de vue d'un utilisateur (i.e. qui a un sens en soi et donc présente une idée ou un concept cohérent) et qui mérite d'être sollicitée dans plusieurs démarches de consultation » [5].

Dans le dataweb, une U.M. est représentée en XML et constitue le contenu d'une page Web. L'ensemble des unités médiatiques et leurs liens navigationnels (ie une **base médiatique**) est obtenu à partir d'un algorithme de génération automatique s'appuyant sur le modèle médiatique. Ce modèle, décrit à l'aide de balises XML spécifiques, est étudié dans cette section.

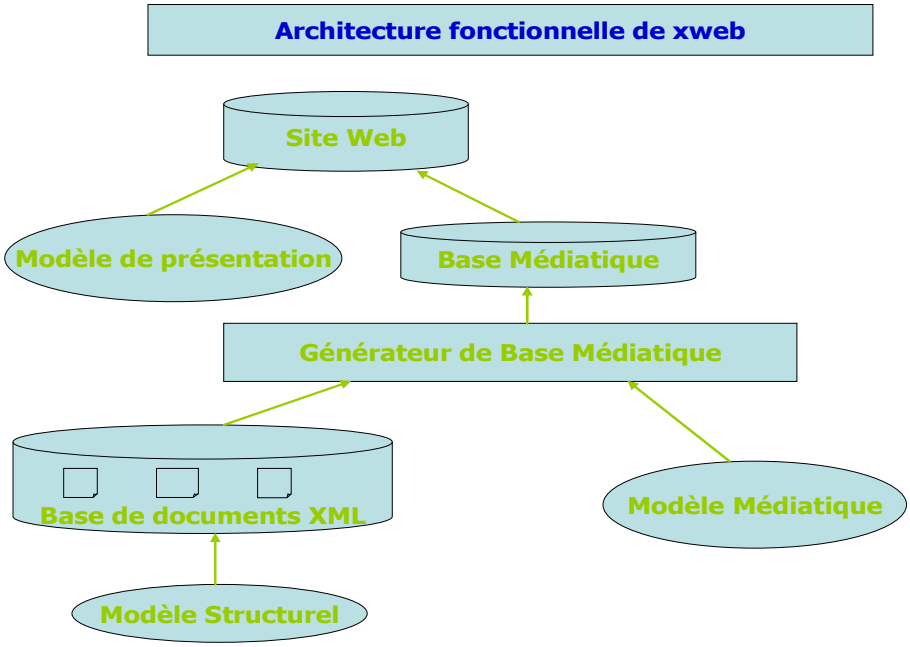

Figure3: Architecture fonctionnelle de Xweb

2.1Les types d'unités médiatiques et leur description en XML

Lo et Hocine[19] proposent huit types d'unités médiatiques : Xobjets, contextes navigationnels, index, menus, liens, textes, images et pages web.

A - Les *Xobjets* : un Xobjet est un extrait de la base de documents XML, obtenu à partir de l'application d'un filtre.

Un filtre peut être engendré automatiquement à partir d'une vue définie en XML. A titre d'exemple, voici une vue permettant d'obtenir des Informations sur les produits des fournisseurs de la centrale d'achat Ndakarou khéweul :

```
<um id="u2" type="Xobjet">
  <vue_filtre id="f_produits_xobjet" type="produit" definition="bd_centrale_achat/produits/produit">

        <nom/>
        <poids_unitaire/>

  </vue_filtre>

</um>
```

Les éléments prédéfinis de l'objet vue_filtre sont : <vue_filtre>, <definition>. Ils permettent d'engendrer automatiquement une requête (appelée filtre) écrite en XSLT (eXtensible Style Language Transformation) [17], XSLT étant un langage permettant de transformer des documents XML en d'autres documents XML.

Nous donnons ci-dessous le code XSLT obtenu à partir de l'objet vue_filtre :

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
    <xsl:output method="xml" indent="yes"/>
     <xsl:template match="/">
      <Xobjet>
         <xsl:for-each select="bd_centrale_achat/produits/produit ">
            <produit>
              <nom> <xsl:apply-templates select="nom"/></nom>
              <poids_unitaire> <xsl:apply-templates select="poids_unitaire"/></poids_unitaire>
            </produit>
        </xsl:for-each>
      </Xobjet>
    </xsl:template>
 </xsl:stylesheet >
```

L'application de ce filtre sur la base de documents XML donne le résultat suivant :

```
<xobjet>
   <produit>
     <nom>Eau Minerale KIRENE</nom>
   <poids_unitaire>330g</poids_unitaire>
     </produit>
     <produit>
       <nom>Eau Minerale FONTAINE</nom>
       <poids_unitaire>500g</poids_unitaire>
     </produit>
     <produit>
       <nom>Pate dentifrice COLGATE</nom>
       <poids_unitaire>120g</poids_unitaire>
     </produit>
     <produit>
       <nom>Pate dentifrice MEDIANE</nom>
       <poids_unitaire>120g</poids_unitaire>
     </produit>
     <produit>
       <nom>Savon CROCO</nom>
       <poids_unitaire>900g</poids_unitaire>
     </produit>
     <produit>
       <nom>Lait en Poudre (VITALAIT)</nom>
      <poids_unitaire>22.5g</poids_unitaire>
     </produit>
     <produit>
       <nom>Lait en Poudre (BARALAIT)</nom>
      <poids_unitaire>22.5g</poids_unitaire>
     </produit>
</xobjet>
```

B - Les contextes navigationnels : un contexte navigationnel est un ensemble de Xobjets accessible à partir d'un index.

On distingue deux types de contexte navigationnels :

- Les *contextes simples* dont le filtre, permettant d'obtenir les xobjets, fait intervenir une seule entité du modèle E/A ;
- Les *contextes composés* dont les filtres font intervenir plusieurs entités (donc au moins une association) du modèle E/A.

B1 – Exemple de contexte simple : les fournisseurs de produits par nom.

Un tel contexte permet d'obtenir plusieurs Xobjets représentant chacun la page d'un fournisseur de produits et un index donnant accès à chacun de ces Xobjets. On le décrit comme suit :

```
<um id="fournisseurs" type="contexte_simple">
  <vue_filtre id="f_four_cont" type="fournisseur"
definition="bd_centrale_achat/fournisseurs/fournisseur">
    <info_index>
      <index>nom</index>
      <titre>fournisseurs</titre>
    </info_index>
    <nom />
    <adresse />
    <telephone />
  </vue_filtre>
</um>
```

Les éléments prédéfinis de l'objet vue_filtre sont : <vue_filtre>, <definition>, <info_index> et <index>. Cette vue est ensuite traduite en un filtre XSLT dont l'application permet d'obtenir les divers Xobjets (ie les fournisseurs) et de construire l'index. On obtient alors le résultat suivant :

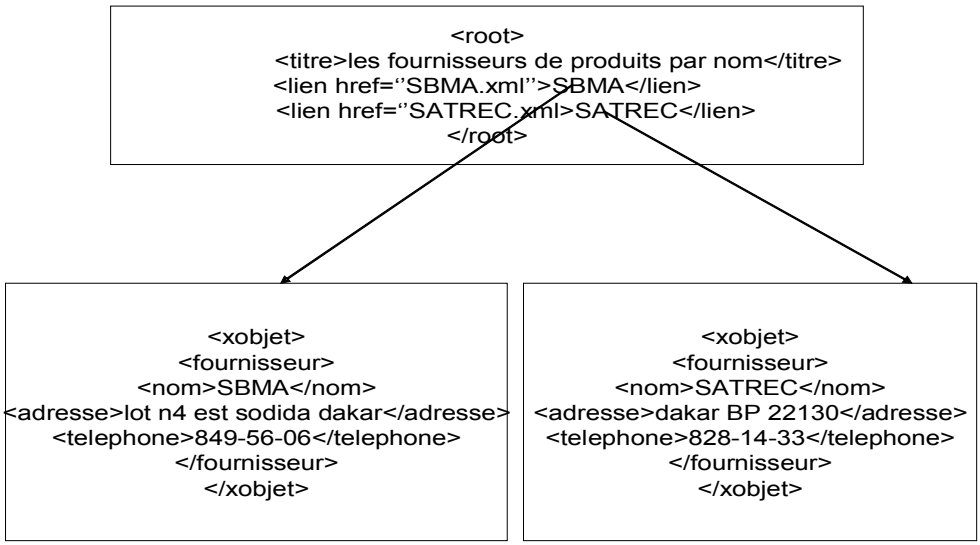

Figure4 : Schématisation d'un exemple de contexte navigationnel simple

B2 – Exemple de contexte composé : les produits vendus par fournisseur.

Contrairement aux contextes simples, les contextes composés font intervenir plusieurs entités et donc au moins une association du modèle E/A. Dans notre première approche, seules deux entités et une association sont prises en compte.

L'exemple fait intervenir l'association "V*endre*" du modèle E/A (Fig.1) reliant les entités "F*ournisseurs*" et "P*roduits*", donc les balises <fournisseurs> et <produits> du modèle structurel. Deux filtres sont alors utilisés :

- Un premier filtre, appliqué à la balise <fournisseurs>, permet d'obtenir la liste des

fournisseurs (id, nom, adresse, telephone). L'identifiant (id), qu'on peut avantageusement remplacer par l'attribut nom, détermine l'index principal.

- Un deuxième filtre, appliqué à la balise <produits> conditionnellement à chaque valeur de l'index principal (ie conditionnellement à chaque fournisseur), permet d'obtenir les produits vendus par fournisseur (les Xobjets représentés ci-dessous) et un index secondaire (les produits de ce fournisseur).

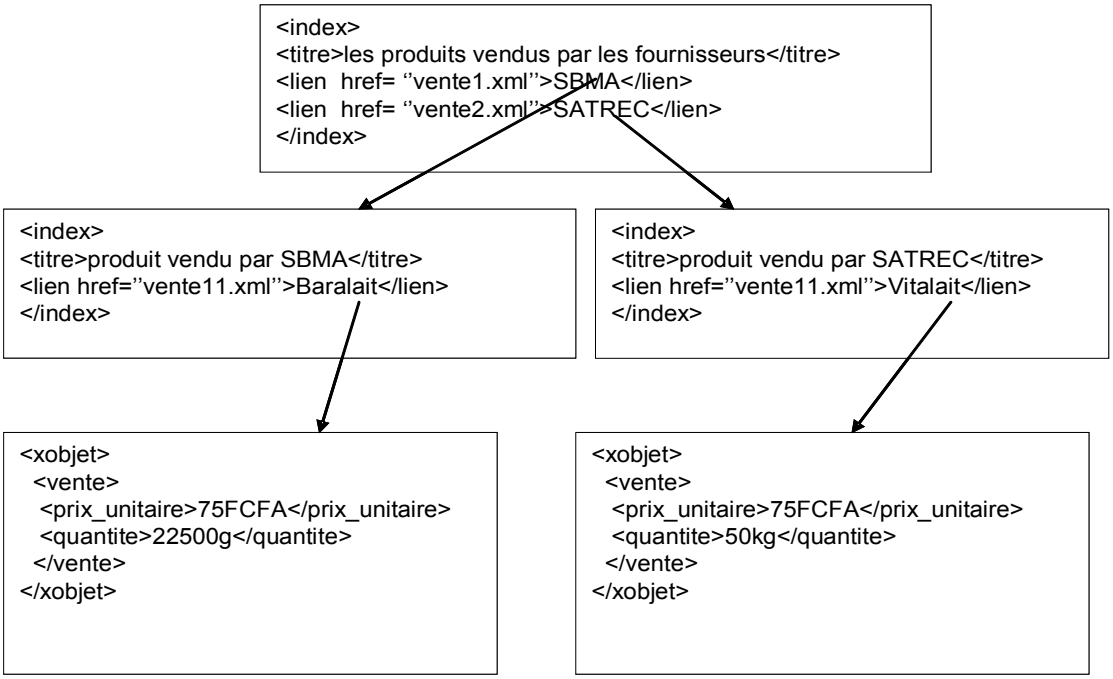

Figure5 : Schématisation d'un exemple de contexte navigationnel composé

C - Les index : ce sont des unités médiatiques composées de liens menant vers des Xobjets appartenant à une thématique donnée. Un index dépend d'un contexte navigationnel.

D - Les menus : un menu est un ensemble de liens (référentiels) vers d'autres unités médiatiques.

```
<um id="menu_gen" type="menu" >
    <lien id="..\resultats\produits.xml">Produits</lien>
    <lien id="..\resultats\fournisseurs.xml">Fournisseurs</lien>
    <lien id="..\resultats\uddi.xml">L'annuaire de services web</lien>
</um>
```

E - Les liens : ces U.M. permettent de décrire les liens hypertextes dans le dataweb. On distingue trois sortes de liens :

E1 - Les liens structurels : ils décrivent la structure navigationnelle des U.M. A partir d'une unité médiatique, un lien structurel permet d'aller vers les unités connexes dans la structure navigationnelle (unités suivante, précédente, page d'accueil, etc...).

E2 - Les liens contextuels : ils proviennent du modèle structurel et permettent de connecter des unités médiatiques ayant des liens sémantiques, par exemple les liens se trouvant dans les index.

E3 - Les liens référentiels : ils permettent d'aller à n'importe quelle U.M., et même à des URL extérieures.

Notons que seuls les liens référentiels sont décrits dans les U.M. ; les autres types de liens sont générés automatiquement à partir de la structure navigationnelle et du modèle structurel.

F - Les textes : une U.M. de type texte est un texte libre ajouté par le concepteur.

```
<texte>Ce travail est réalisé dans le LANI</texte>
```

G - Les images : une U.M. de type image permet d'insérer une image à partir d'un fichier.

```
<image>ndakarou_khéweul.gif</image>
```

H - Les unités pages : ce sont des unités englobantes, décrites en XML, qui seront associées à des feuilles de style pour générer les pages web du site.

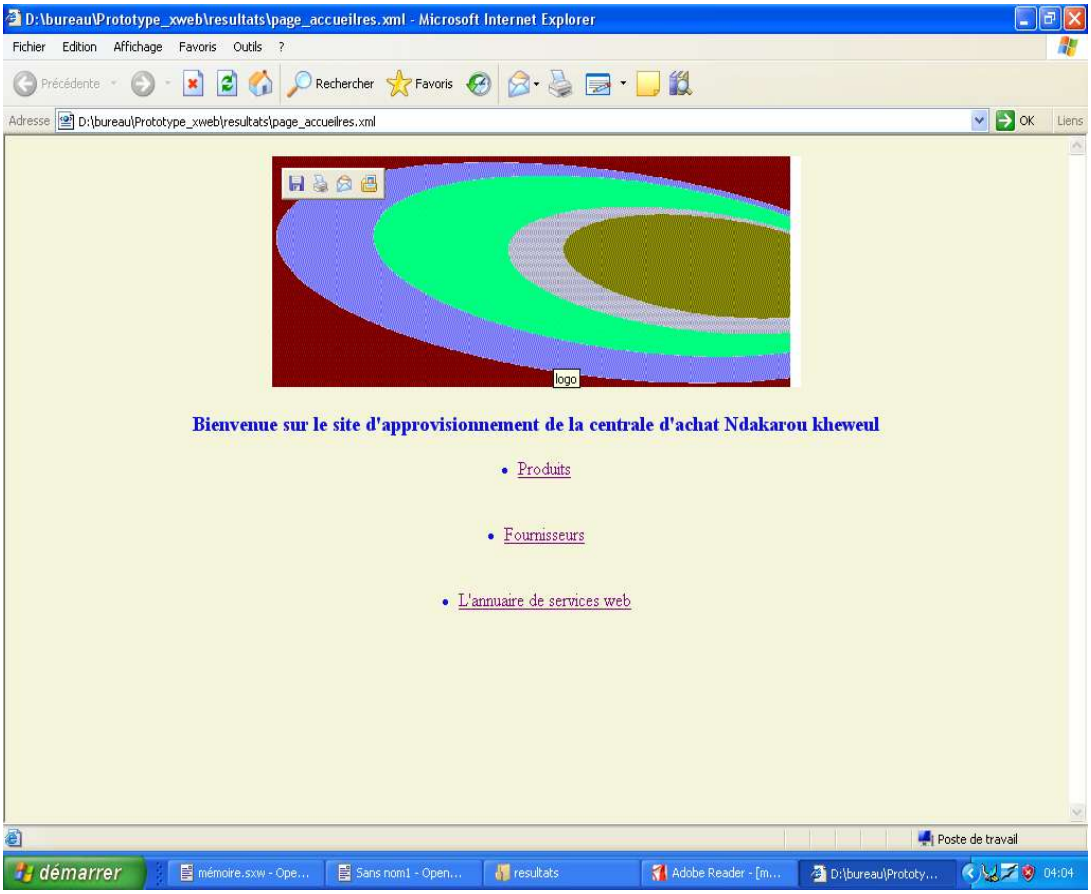

Figure 6 *: Page d'accueil du site web de la centrale d'achat Ndakarou Khéweul*

La figure 6 montre la page d'accueil de la centrale d'achat Ndakarou Khéweul. Cette page résulte de l'application d'une forme médiatique (ici une feuille de style) à l'unité médiatique suivante :

```
<racine>
 <image href="..\donnees1\ndakarou_kheweul.gif"/>
 <titre>Bienvenue sur le site d'approvisionnement de la centrale d'achat Ndakarou kheweul</titre>
 <lien href="..\resultats\produits.xml">Produits</lien>
 <lien href="..\resultats\fournisseurs.xml">Fournisseurs</lien>
 <lien href="..\resultats\uddi.xml">L'annuaire de services web</lien>
</racine>
```

Cette unité médiatique, de type "unité page", a été engendrée à partir de la description suivante :

```
<um id="page_accueil" presentation="menu.xsl">
 <image>ndakarou kheweul</image>
 <titre>Bienvenue sur le site d'approvisionnement de la centrale d'achat Ndakarou kheweul</titre>
 <lien id="..\resultats\produits.xml">Produits</lien>
 <lien id="..\resultats\fournisseurs.xml">Fournisseurs</lien>
 <lien id="..\resultats\uddi.xml">L'annuaire de services web</lien>
</um>
```

2.2 Les différentes étapes pour la construction de la base médiatique

Pour construire une base médiatique, Lo et Hocine proposent les étapes suivantes :

a) Déterminer les "unités pages" (les nommer et décrire leurs fonctions) ;
b) Déterminer les liens navigationnels entre unités pages ;
c) Décrire formellement en XML les unités médiatiques à construire ;
d) Engendrer la base médiatique à partir de la description des U.M. et de la base de documents XML

3.Génération de la base médiatique :

La génération de la base médiatique résulte d'un processus relativement complexe (notamment pour les unités médiatiques de type « contexte composé »).

L'algorithme de génération reçoit en entrée trois types d'informations :

- La base de documents XML (Fig 2).
- La description des unités médiatiques en XML (section 2.1), c'est à dire la description de chaque unité médiatique de type « unité page».
- Un catalogue de métadonnées permettant notamment d'établir une correspondance entre les noms logiques des ressources (images, URL de destination des liens hypertextes, filtres…) et leurs noms physiques : on peut par exemple préciser que l'U.M. correspondant au logo de la centrale d'achat, nommée « ndakarou kheweul » (voir paragraphe G, section 2.1), correspond au fichier « ndakarou_kheweul.gif ». Ce catalogue est lui aussi écrit en XML.

Le résultat de l'algorithme est une collection d'unités médiatiques, essentiellement des unités médiatiques de type « Xobjet » et « unités pages ». Pour construire les pages Web du site à partir de la base médiatique, il reste à associer une forme médiatique à chaque unité médiatique. Cela peut se faire en utilisant des feuilles de style (CSS ou XSL) ou même en manipulant directement les documents XML contenant les U.M. à l'aide d'un éditeur WYSIWYG.

Dans la suite de cette section, nous étudions l'algorithme de génération d'une page puisque l'algorithme général en découle directement. Cet algorithme s'appuie sur la structure arborescente de tout document XML [18], en l'occurrence sur la structure arborescente d'une unité médiatique de type "unité page".

Procédure Générer_Page (Racine : Noeud)

```
Début
   Ouvrir en écriture le fichier (nommé F) dans lequel sera créé l'UM de type «unité page»
      / * le nom physique est déterminé à partir de la balise id et du catalogue */
      Pour chaque noeud fils (nommé nd_fils) de la racine de l'arbre Faire
            Traiter_Noeud (nd_fils, F)
      Fin Pour
 Fin
```

Procédure Traiter_Noeud (nd : Noeud, F : Fichier)

```
Debut
  Selon le type de nd
   Cas "Texte" :
         insérer le texte dans le fichier F
   Cas "Image" :
         trouver le nom physique du fichier image à partir du catalogue
         l'insérer dans le fichier F
   Cas "Lien" :
          trouver le nom physique de l'URL à partir du catalogue
          créer un lien référentiel vers cette URL dans le fichier F
   Cas "Xobjet" :
          exécuter la vue
          insérer le résultat dans le fichier F
   Cas "contexte simple" :
          exécuter la vue
          en déduire les Xobjets et un index
          insérer l'index dans le fichier F
          créer un fichier pour chaque Xobjet
   Cas "contexte composé" : /* version simplifiée : seules 2 vues sont considérées ! */
          exécuter la 1ère vue
          créer un index principal à partir de la 1ère vue et l'insérer dans le fichier F
          Pour chaque élément de l'index principal
                exécuter la 2ième vue conditionnellement à cet élément
                en déduire les Xobjets et un index secondaire
                créer un fichier pour l'index secondaire
                créer un fichier pour chaque Xobjet
          Fin Pour
  Fin Selon
Fin
```

Algorithme 1 : Algorithme de génération d'unités médiatiques