```
<!-- ****************************************

Project code: Web'math (c) ABORD, 1999
Object: XML DTD
File: pexp3.dtd
Usage: Raw data description
Version: 1.0;
Dated: june 1999
Modified: may 2000, september 2000, january 2001, april 2001, april 2002

* Links can be directly used in the document without LINKS
  declaration
* LINKS element is integrated in HEAD element
* internal links (notes, questions) are integrated in "BODY"
  element.
* january 2002, cleaning
* april 2002, new version 1,1 not totally compatible vith 1.0

Note: the file with the leading number is the reference.
Examples can be found in the old version in /PROLOG/PEXP_MIG directory.
See also the files pexp.dtd and pexp<n>.dtd for further comments,
mapping with the old Prof'Expert format (color, etc.).

*********************************************  -->

<!-- *** Les entites des elements de base *** -->

<!ENTITY % elem.mini "#PCDATA | IMG | A">
<!ENTITY % elem.base "#PCDATA | FORMULE | BBOARD | PHRASE | IMG | A | DIV">
<!-- Note: en principe de type text avec BBOARD qui peut devenir bloc et
     DIV qui contient du (X)HTML ! -->
<!ENTITY % elem.plus "%elem.base; | VAL-VAR | VAL-CHAMP | NOTE">
<!ENTITY % elem.spec "INPUT">

<!-- ************************ -->

<!-- *** Le document global *** -->

<!ELEMENT PEXP (HEAD,%content.pexp;)>

<!-- ************************ -->

<!-- *** Entete *** -->

<!ELEMENT HEAD (TITLE, ATTLIST?, CTXTLIST?, VARLIST?, SCRIPT?, LINKS?)>
<!ELEMENT TITLE (#PCDATA)>

<!-- *** Attributs *** -->

<!ENTITY % att.didactique    "cadre|domaine|sdomaine|sujet|notions
                              |dtype|keywords|level">
<!-- attention: notions se rapporte a des mots cles notionnels, keywords
     se refere au contenu (a corriger les documents RMT) -->
<!-- a voir encore le probleme du lexique -->
<!ENTITY % att.technique     "date|nodenr|name|base|variable|file
                              |ttype|window">
<!ENTITY % att.administratif "from|copyright|reference|langue|auteur">

<!ELEMENT ATTLIST (ATT)*>
<!ELEMENT ATT EMPTY>
<!ATTLIST ATT
        NAME (%att.didactique;|%att.technique;|
              %att.administratif;) #REQUIRED
        VALUE  CDATA #IMPLIED
        SCHEME CDATA #IMPLIED>
```

```
<!-- Quand plusieurs "values" sont autorisees (liste), elles sont separees
     par des espaces (dans Prof'Expert, il y avait des ;). A typer
     ulterieurement par NMTOKENS

* didactique
     cadre   : *histoire (culture),                        [hist]
               vie-courante (applications),                [appl]
               fiction (enigme),                           [situ]
               *jeu                                        [jeux]
               *theorie (approche formelle)                [theo]
               *informatique,                              [info]
               didactique (informations didactiques),      [dida]
               *enseignement (expose didactique, exercice),[ense]
               none

         note: * marque les cadres repris comme int,rˆt de visite (??)
               entre [] le nom dans Ermitage


     level   : 1 (cp) 2 (ce) 3 (cm) 4 (sec 6-7) 5 (sec 8-9) 6 (ly)
               7 (specialiste) 8 (didacticien)
               note: ce level est attribue lors de l'inscription, mais
               tous ne sont pas libres de choix !
               Lors de l'inscription existent encore:
               0 (visiteur) 9 (administrateurs)
               si le level est "" (nill) on garde l'ancien

         note: a voir pour dehierarchiser

     domaine : arithmetique, algebre, geometrie, ensemble-relation,
               logique, analyse, combinatoire, statistique,
               algorithmique, topologie, general
         note: peut dependre du cadre et du niveau

     sous-domaine : depend du domaine
     sujet   : depend du sous-domaine

     Du point de vue "museographique" le domaine correspond … l'etage et
     le cadre a une "aile" du batiment. Des zones plus
     petites correspondent a un sdomaine (groupes de salles),
     un sujet est une salle.

     notions : liste de notions mathematiques
     dtype   : fiche, situation, probleme, exercice, atelier, corrige
               note: a completer depend du domaine et du cadre
     keywords: liste liee au contenu (habillage)
     level   : niveau auquel s'adresse la fiche

        note: actuellement keywords a ete aussi utilise pour 'notion'


* technique
     date    : (SCHEME="std | old") (std = default)
     nodenr  : ancienne numerotation pourra etre remplace par name
     name    : (note: l'ancien printname est devenu TITLE)
     base    : anciennement le fichier hba/ actuellement le directory
     file    : le nom du ficher
     window  : reprend les informations de prof'expert sur la fenetre; a
               voir le statut actuel (elements de style ??)
     ttype   : prevu pour differencier fiche (xml, html), interface, menu, etc.
               (pas utiliser
     langue  : par default fr
     variable: abandonne au profit des variables a part entiere (VARLIST)

     note : window : (avec SCHEME="old") WC/FC/TOP/LEFT/HEIGHT/WIDTH
     WC mod 16 = couleur caractere ;  WC div 16 = couleur fond
```

```
     idem pour FC (FrameColor). Voir les anciens fichiers pour des
     precisions.

* administratif
     from    : pour indiquer la provenance (notamment migration prof'expert)
               avec l'idee de "transmis par" dans le cas ou ce n'est pas
               l'auteur.
     copyright : detenteur du copyright
     auteur    : l'auteur (les 1eres fiches -> aout 2001 n'ont que le
                 copyright
     reference : litterature (;-liste)
     langue    : langue utilisee selon la denomination standard (fr, en, it)

-->

<!-- ************************ -->

<!-- *** Contextes *** -->

<!ELEMENT CTXTLIST (CTXT*)>

<!ELEMENT CTXT EMPTY>
<!ATTLIST CTXT
      NAME (expert|controle|interaction) #REQUIRED
        VALUE  CDATA #IMPLIED
        SCHEME CDATA #IMPLIED>

<!--

     NAME        |      SCHEME     |     VALUE
     ************************************************************
     expert      |      dom        |     none (default)
     expert      |      dom        |
     expert      |      did        |     lacune /
     ************************************************************
     controle    |      tolerance  |     none (default) (textuel)
     controle    |      tolerance  |     calcul  (1+1 est admis pour 2)
     controle    |      remise     |     oui / non
     controle    |      ordre      |     original / aleatoire
     ************************************************************
     interaction |      type       |     lacune
     interaction |      mode       |     crochet / enigme / choix
     ************************************************************
     interaction |      type       |     qrep
     interaction |      mode       |     normal / qcm
     interaction |      feed-back  |     oui / non
     interaction |      reprise    |     oui / non / N
     interaction |      choix      |     N
     ************************************************************

A completer et a voir encore: ton, sexe, option imposee ou non,
tempo (mais le menu … disposition pourrait suffire)
-->

<!-- ************************ -->

<!-- *** Variables *** -->

<!ENTITY % var.element "VAR | LOC">

<!ELEMENT VARLIST (%var.element;)+>

<!-- variables "globales" -->
<!ELEMENT VAR EMPTY>
<!ATTLIST VAR
      NAME CDATA #REQUIRED
```

```
      VALUE CDATA #REQUIRED>

<!-- variables "locales" (liee a une salle) -->
<!ELEMENT LOC EMPTY>
<!ATTLIST LOC
      NAME CDATA #REQUIRED
      VALUE CDATA #REQUIRED>

<!-- ************************ -->

<!-- *** Scripts *** -->

<!ELEMENT SCRIPT (#PCDATA)>
<!ATTLIST SCRIPT
        LANGUAGE CDATA #REQUIRED>

<!-- a voir la liaison avec LINKS -->
<!-- … considerer le <![CDATA[ ... ]]>

<!-- ************************ -->

<!-- *** Links *** -->
<!ENTITY % link.element "LNK | INTLNK">
<!ENTITY % type       "texte|note|lexique|popup">
<!ENTITY % function   "reference|aide|coup-pouce|relance|solution|
                       info|message|feed-back|outil|exercice|defi|
                       hlp-guide|exemple|methode|analyse|next|
                       curiosite|question">

<!-- attention a verifier; curiosite question avaient ete mis
     sous type. -->
<!-- liens sur style et/ou script. Est-ce utile ? -->

<!ELEMENT LINKS (%link.element;)+>
<!ELEMENT INTLNK (%elem.base;)>

<!ELEMENT LNK EMPTY>
<!ATTLIST LNK
      IDREF   CDATA #IMPLIED
      HREF    CDATA #REQUIRED
      LEVEL   CDATA #IMPLIED
      NAME    CDATA #IMPLIED
      SEGMENT CDATA #IMPLIED
      TARGET  CDATA #IMPLIED
      TITLE   CDATA #IMPLIED
        TYPE     (%type;) "texte"
        FUNCTION (%function;) #IMPLIED>

<!-- Warning: IDREF reserve !! -->
<!-- TARGET fenetre, dont extern (ouvre nouvelle fenetre). Utilisation
     a eviter si possible (type & fonction permettent souvent
     de calculer la target

     IDREF si il est non vide cela indique qu'un lien est fait depuis le
     le document. Donc le lien ne sera pas affiche dans une partie
     next ou lien d'une salle (voir aussi "WEBMATH-History.doc" pour
     les details)

     NAME nom du document complete par SEGMENT (ancre)
-->

<!ATTLIST INTLNK
      IDREF  CDATA #IMPLIED
      LEVEL  CDATA #IMPLIED
      TARGET CDATA #IMPLIED
      TITLE  CDATA #IMPLIED
```

```
      TYPE (%type;) "note"
        FUNCTION (%function;) #IMPLIED>

<!-- du point de vue jvs, les donnees de LNK et INTLNK sont transmises
     de la meme facon. Dans le cas de INTLNK le contenu va dans le
     slot reserve a HREF. Un slot 'place' dans les donnees marque
     la difference
-->

<!-- Remarques

  1) info = reference (a preferer info)
  2) lexique: a voir si ce n'est pas une option toujours presente ?
  3) methode: analyse didactique pour apprenant (y compris objectif)
  4) analyse: analyse didactique pour le formateur

  5) A voir lien communication (forum etc.), et ev. d'autres informations
     sur le lien (langue). Il y aura aussi des 'donnees complementaires
     (regle, switch, etc. pour le guide, par exemple) a moins
     que cela ne soit insere dans les data

  6) A voir aussi l'idee de la "methode" qui permet de suivre l'info
     (fichier html, xml, autres techniques ssi)

  TYPE: donne le type d'objet lie et/ou la cible (sera a completer)

  FUNCTION: donne une indication d'usage
            Quelques fonctions peuvent avoir un TYPE par defaut
-->

<!-- ******************** -->

<!-- *** Contenu *** -->

<!ENTITY % content.pexp "(BODY | CONTENT)">

<!ENTITY % elem.comp "INTRO | PROPOSITION | EXPRESSION | EXEMPLE | DEFINITION |
                      CITATION">

<!-- autres elements: CONSIGNE, ENONCE, FORM-REP -->

<!ENTITY % body.content "ANY"> <!-- (X)HTML -->
<!ELEMENT BODY  %body.content;>
<!ATTLIST BODY
      ONFOCUS CDATA #IMPLIED
      ONLOAD CDATA  #IMPLIED
      ONUNLOAD CDATA #IMPLIED>
<!-- futur %body.content; : jeu XHTML reduit a decider -->

<!ELEMENT DIV   %body.content;>
<!-- attribut standard dont une classe permettant de reprendre des
     donnees de pe basees sur des caracteres semi-graphiques -->

<!ENTITY % qcm         "INTRO?, CONSIGNE?, ITEM+">
<!ENTITY % simple-int  "INTRO?, DATA">
<!ENTITY % situation   "ENONCE, CONSIGNE?">
<!ENTITY % exercice    "INTRO?, CONSIGNE?, QUESTION+, FORM-REP?">
<!ENTITY % fiche       "INTRO?, (%elem.comp;)*">
<!ENTITY % analyse     "RESOLUTION, SOLUTIONS">

<!ENTITY % content.content "(%qcm;) | (%simple-int;) | (%situation;) |
                           (%exercice;) | (%fiche;) | (%analyse;)">
<!ENTITY % type.doc  "qcm | simple-int | situation | exercice |
                      fiche | analyse">

<!-- reprend les attributs du doc, utile pour les segments. A verifier
```

```
     la correspondance -->

<!ELEMENT CONTENT (%content.content; | SEGMENT*)>
<!ATTLIST CONTENT TYPE (%type.doc;) #IMPLIED>

<!ELEMENT SEGMENT (%content.content;)>
<!ATTLIST SEGMENT TYPE (%type.doc;) #IMPLIED>

<!-- autres elements: RESOLUTION, SOLUTIONS
                      QUESTION, INTERACTION, ITEM,
                      DATA, SEGMENT, INTERACTION -->

<!-- ************************ -->

<!-- *** Images *** -->

<!ELEMENT IMG EMPTY>
<!ATTLIST IMG
      SRC    CDATA #REQUIRED
      WIDTH  CDATA #IMPLIED
      HEIGHT CDATA #IMPLIED>

<!-- ************************ -->

<!-- *** Ancres *** -->

<!ELEMENT A (%elem.base;)*>
<!ATTLIST A
        IDREF  IDREFS #IMPLIED
      HREF   CDATA #REQUIRED>

<!-- a voir dans le cas sans IDREF a pouvoir distinguer
les differents liens, par exemple :

<A HREF="javascript:parent.simpleReponse('','',4)">
<IMG SRC="images/pava4.gif" WIDTH="81" HEIGHT="81"/></A>

Cela demanderait d'ajouter un attribut TYPE et un
attribut PARAM.

HREF ne serait plus que le cas fourre tout en voie de
disparition.

Remarques concernant les multi-liens

1)
<LNK IDREF="lien1" ...>
<LNK IDREF="lien2" ...>
...
<A IDREF="lien1 lien2">

2) A eviter ?
<LNK IDREF="lien" ...>
<LNK IDREF="lien" ...>
...
<A IDREF="lien">

-->

<!-- ************************ -->

<!-- *** Formules *** -->

<!ELEMENT FORMULE ANY>
<!ATTLIST FORMULE
        FORMAT (latex | mathml) #IMPLIED
```

```
        CODE CDATA #IMPLIED>
<!-- ANY pour mathml -->

<!-- ************************ -->

<!-- *** Tableau *** -->

<!ELEMENT BBOARD (CAPTION?,LABEL*)>
<!ATTLIST BBOARD
      WIDTH  CDATA #REQUIRED
      HEIGHT CDATA #REQUIRED
      PRGM CDATA #REQUIRED>
<!-- A etendre pour en faire differents panels -->

<!ELEMENT CAPTION (#PCDATA)*>

<!ELEMENT LABEL (#PCDATA)*>
<!ATTLIST LABEL
      POS CDATA #REQUIRED>

<!-- ************************ -->

<!-- *** Phrases, notes *** -->

<!ELEMENT PHRASE (%elem.mini;)*>
<!ELEMENT NOTE   (%elem.base;)*>

<!-- ************************ -->

<!-- *** Valeur d'une variable locale a la salle *** -->

<!ELEMENT VAL-VAR EMPTY>
<!ATTLIST VAL-VAR
        NAME CDATA #REQUIRED>
        TYPE CDATA #IMPLIED>
<!-- permettre aussi variable globale
     et l'acces a une valeur retourne par une fonction
     via le type -->

<!-- ************************ -->

<!-- *** Requete sur db *** -->

<!ELEMENT VAL-CHAMP EMPTY>
<!ATTLIST VAL-CHAMP
        NAME       CDATA #REQUIRED
        DATASOURCE CDATA #REQUIRED
        SELECT     CDATA #REQUIRED
        FROM       CDATA #REQUIRED
        WHERE      CDATA #IMPLIED
        TYPE       CDATA #IMPLIED>
<!-- A prevoir aussi eventuellement via un link
     a permettre aussi sous la forme d'un element ou la
     requete est le contenu (a la maniere d'un script ou
     structure en sous elements -->

<!-- ************************ -->

<!-- *** Blocs *** -->

<!ELEMENT INTRO       (%elem.plus;)*>
<!ELEMENT PROPOSITION (%elem.base;)*>
<!ELEMENT NOTION      (%elem.mini;)*>
<!ATTLIST NOTION
        NOM CDATA #IMPLIED>
<!-- les notions donnees par un nom sont des elements vides -->
```

```
<!ELEMENT EXPRESSION  (%elem.base;)*>
<!ELEMENT EXEMPLE     (%elem.base;)*>
<!ELEMENT DEFINITION  (NOTION, EXPRESSION, EXEMPLE*)>
<!ELEMENT CITATION    (%elem.base;)*>
<!ELEMENT CONSIGNE    (%elem.base;)*>
<!ELEMENT ENONCE      (%elem.plus;)*>

<!-- THEOREME ? -->

<!-- *** Formulaire *** -->

<!ELEMENT FORM-REP (%elem.plus; | INPUT)*>
<!ATTLIST FORM-REP
      TYPE  CDATA #IMPLIED
      SIZE  CDATA #IMPLIED
      SCHEME CDATA #IMPLIED>
<!-- type / scheme actuel:
      entier / textuel (usage du modele) ;
      entier / apply (usage de verifReponse() );
      bouton / apply
-->

<!ELEMENT INPUT EMPTY>
<!ATTLIST INPUT
      TYPE CDATA #REQUIRED
      NAME CDATA #REQUIRED
      SIZE CDATA #IMPLIED
      VALUE CDATA #IMPLIED>
<!-- prevu pour le cas TYPE="bouton" pour pouvoir mettre des
     champs additionnels. A voir s'il ne faudrait pas passer
     en parametre le formulaire a verifReponse()
-->

<!-- ************************** -->

<!ELEMENT RESOLUTION (INTRO,ETAPE*)>
<!ELEMENT ETAPE (%elem.plus;)*>
<!ATTLIST ETAPE
        TITRE CDATA #IMPLIED>
<!-- a voir lien avec syst. expert -->

<!ELEMENT SOLUTIONS  (INTRO, UNE-SOL*)>
<!ELEMENT UNE-SOL (%elem.plus;)*>
<!ATTLIST UNE-SOL
        TITRE  CDATA #IMPLIED
        EVAL   CDATA #IMPLIED
        SCHEME CDATA #IMPLIED>

<!ELEMENT QUESTION  (ENONCE?, CONSIGNE?, INTERACTION)>
<!ATTLIST QUESTION
      NAME  CDATA #IMPLIED
        DISPOSITION CDATA #IMPLIED>

<!ELEMENT INTERACTION (INTRO?, CONSIGNE?, ATTLIST?,CTXTLIST?,ITEM*)>
<!ATTLIST INTERACTION
      TYPE  CDATA #REQUIRED
      WIDTH CDATA #IMPLIED
      HEIGHT CDATA #IMPLIED>

<!-- NAME : consigne, remp, type, mode, rows, cols, align, sep
            content -->
<!-- anciennement DATA-COMP a la place de CTX et ATT  -->

<!ELEMENT DATA (DATA-TYPE?, DATA-COMPL*)>
<!ELEMENT DATA-TYPE EMPTY>
<!ELEMENT DATA-COMPL EMPTY>
```

```
<!ATTLIST DATA-COMPL
      NAME CDATA #REQUIRED
      VALUE CDATA #IMPLIED>

<!-- a mettre des infos type: lisibilite (difficulte),
     heritees ou redefinies dans les items -->


<!ELEMENT ITEM (CONSIGNE?, ENONCE?, CHOICE+)> <!-- qcm -->
<!ATTLIST ITEM
   NAME  CDATA #IMPLIED>

<!ELEMENT CHOICE (#PCDATA)>

<!-- a faire ev. des nouveaux elements
     SEGMENT TYPE="analyse" devient ANALYSE -->


<!-- **************** FIN DE LA DTD ***************** -->


<!-- etude des feuilles de style existantes et des modifications
     a apporter selon cette nouvelle version

qrep.xsl  / qrep2.xsl  (exercice)

PEXP
        HEAD
                TITLE
                ATTLIST
                VARLIST
                SCRIPT
                LINKS
        CONTENT
                DIV
                INTRO
                        VAL-VAR
                FORM-REP
                        VAL-VAR
                        INPUT

(a introduite DIV dans INTRO)

qcm2.xsl (qcm)

PEXP
        HEAD
                TITLE
                ATTLIST
                CTXLIST
                LINKS
        CONTENT
                INTRO
                ITEM*
                        TEXT
                        CHOICE*

(remplacer TEXT par ENONCE)

applet.xsl / applet2.xsl (simple-int)

PEXP
        HEAD
                TITLE
                ATTLIST
                CTXLIST (contient nom de l'applet)
```

```
                LINKS
        CONTENT
                INTRO
                DATA
                        DATA-COMPL*

(cas eventuellement a eviter)

qtml.xsl / vtml.xsl

PEXP
        HEAD
                TITLE
                ATTLIST
                VARLIST
                LINKS
        BODY

lacune.xsl (exercice)

PEXP
        HEAD
                TITLE
                ATTLIST
                CTXLIST (contient seul. l'indication de l'interaction lacune
                LINKS
        CONTENT
                INTRO
                QUESTION*
                        DATA
                                DATA-COMPL*
(remplacer l'interieur de question par: INTERACTION avec
 a l'interieur ATTLIST et/ou CTXTLIST)

fiche.xsl (fiche)

PEXP
        HEAD
                TITLE
                ATTLIST
                LINKS
        CONTENT
                INTRO
                DEFINITION*
                        TEXT
                        BBOARD*

(revoir le contenu de definition)

fiche2.xsl (exercice)

PEXP
        HEAD
                ATTLIST
        CONTENT
               INTRO
               QUESTION
                        ENONCE
                        INTERACTION
                                DATA-COMPL

(remplacer DATA-COMPL par ATTLIST et/ou CTXTLIST)

-->
```