8 – Références croisées et liens

Il est possible de coder les références croisées, ou des liens explicites entre des points différents d'un même document SGML, au moyen des éléments décrits en section 8.1 « Références croisées simples ». Les références ou liens vers des éléments situés dans un autre document SGML, ou vers des parties de documents non-SGML, peuvent être codés au moyen des pointeurs étendus TEI décrits en section 8.2 «Pointeurs/indicateurs étendus». Les liens implicites (telle l'association entre deux textes parallèles ou celle entre un texte et son interprétation) peuvent être codés au moyen des attributs de liaison expliqués en section 8.3 « Attributs de liaison ».

8.1. Références croisées simples

Une référence croisée reliant deux endroits situés dans le même document peut être codée au moyen de l'un ou l'autre des éléments suivants :

<ref> référence à un autre emplacement dans le document courant, en termes d'un ou plusieurs éléments identifiables, le tout pouvant éventuellement être modifié par un commentaire ou du texte supplémentaire ;

<ptr> pointeur vers un autre emplacement dans le document courant, en termes d'un ou plusieurs éléments identifiables.

Ces éléments partagent les attributs suivants :

target indique la destination du pointeur en termes d'un ou plusieurs identifiants SGML ;

type catégorise le pointeur d'une manière quelconque, au moyen de tout ensemble de catégories approprié ;

targType
indique le ou les types d'éléments désignés par ce pointeur ;

crDate indique la date de création du pointeur ;

resp indique le créateur du pointeur.

La différence entre ces deux éléments est que <*ptr*> est un élément vide qui marque simplement un endroit à partir duquel un lien est à créer, tandis que <*ref*> peut contenir également du texte – typiquement le texte de la référence croisée elle-même. L'élément <*ptr*> serait plutôt employé pour une référence croisée signalée par des dispositifs non-verbaux tels qu'un symbole ou une icône, ou dans un texte électronique par un bouton. Il sert également dans des systèmes de production de document, où le logiciel de formatage a la possibilité de générer la forme verbale correcte de la référence croisée.

Les deux formes présentées ci-après, par exemple, sont logiquement équivalentes (en supposant que nous avons documenté ailleurs la forme verbale exacte des références croisées représentées par les éléments <*ptr*>) :

```
Voir notamment  <ref target=SEC12>la section 12, page 34</ref>.

Voir notamment <ptr target=SEC12>.
```

la valeur de l'attribut *target* doit être un identifiant SGML dans le document SGML courant. Ceci implique que le passage ou l'expression pointé doit porter un identifiant, et donc être balisé comme un élément d'un certain type. Dans l'exemple suivant, la référence croisée pointe vers un élément <*div1*> :

```
...
voir notamment <ptr target=SEC12>.
...
<div1 id=SEC12><head>En ce qui concerne les identificateurs
...
```

Puisque l'attribut *id* est global, tout élément dans un document peut être pointé de la même façon. Dans l'exemple suivant, un paragraphe a été pourvu d'un identifiant pour qu'on puisse pointer sur lui :

```
...
on en reparlera dans <ref target=pspec>la section sur
les liens</ref>
...
<p id=pspec>Les liens peuvent être composés de n'importe
quel type d'éléments
...
```

Il est possible d'utiliser l'attribut *targType* pour préciser que l'élément pointé doit être d'un type particulier, comme dans l'exemple qui suit.

```
...
on en reparlera dans  <ref target=dspec targType='div1 div2'>
la section sur les liens</ref>
```

Cette référence devrait échouer si l'élément portant l'identifiant `dspec` n'est ni un <*div1*> ni un<*div2*>. À noter cependant que cette vérification ne peut être effectué par un analyseur syntaxique SGML seul puisque ce dernier ne peut que vérifier l'existence d'un élément *dspec* donné.

L'attribut *type* peut être employé pour catégoriser le lien représenté par le pointeur, par tout moyen approprié. Les attributs *resp* et *crDate* peuvent également être utilisés pour identifier la personne ou l'entité responsable de la création du lien, ainsi que la date de création de ce dernier, comme dans l'exemple suivant :

```
on en reparlera dans
<ref type=xref resp=auto crDate=950521 target=dspec
 targtype='div1 div2'>
la section sur les liens</ref>
```

Selon toute vraisemblance, ces attributs seront les plus utiles dans des systèmes hypertextes contenant de nombreux pointeurs ayant de multiples fonctions et créés par toute sorte de moyens.

Parfois, la cible d'une référence croisée ne correspond à aucune caractéristique particulière d'un texte, et ne peut donc pas être balisée comme étant d'un certain type d'élément. Si la cible désirée est simplement un endroit dans le document courant, la façon la plus facile de le marquer est d'introduire un élément <*anchor*> à l'endroit correspondant. Si la cible est une séquence de mots non balisée autrement, l'élément<*seg*> (segment) peut être introduit pour la marquer. Ces deux éléments sont décrits comme suit :

<*anchor*>
: indique un emplacement ou un endroit sur lequel on peut pointer dans un document ;

<*seg*>
: identifie un passage ou un segment de texte au sein d'un document, sur lequel on peut pointer ; exemple d'attribut :

  type catégorise le segment.

Dans l'exemple (imaginaire) suivant, des éléments <*ref*> ont été employés pour représenter des points dans le texte devant être attachés d'une certaine manière à d'autres parties du texte, un point précis dans le premier cas et une séquence de mots dans le second :

```
Si je reviens sur <ref target=ABCD>le passage sur lequel je
me suis endormi</ref>, je note que <ref target=EFGH>trois
mots</ref> ont été entourés de rouge par un précédant lecteur.
```

Ce codage nécessite que des éléments ayant les identifiants spécifiés (`ABCD` et `EFGH` dans cet exemple) existent quelque part ailleurs dans le document courant. En supposant qu'aucun élément n'existe déjà pour porter ces identifiants, les éléments <*anchor*> et <*seg*> (segment) peuvent être employés :

```
.... <anchor type=bookmark id='ABCD'> ....
 ....<seg type=target id='EFGH'> ... </seg> ...
```

L'attribut *type* devrait être employé (comme ci-dessus) pour préciser les différents rôles de ces éléments généraux dans un texte. D'autres applications sont présentées ci-après en section 8.3 « Enchaînement d'attributs ».

8.2. Pointeurs étendus

Les éléments <*ptr*> et <*ref*> servent uniquement pour des références croisées ou des liens dont les cibles se trouvent dans le même document SGML que leur source. En outre, ils ne peuvent indiquer que des éléments SGML. Les éléments présentés dans la présente section ne sont pas limités de cette façon.

<*xptr*> définit un pointeur vers un autre emplacement dans le document courant ou dans un document externe ;

<*xref*> définit un pointeur vers un autre emplacement dans le document courant ou dans un document externe, modifié éventuellement par un commentaire ou un texte supplémentaire.

En plus des attributs applicables aux pointeurs présentés dans la section 8.1 « Références croisées simples » ci-dessus, ces éléments partagent les attributs supplémentaires suivants, dont le but est de spécifier la cible de la référence croisée ou du lien, au lieu de l'attribut *target* :

doc indique le document dans lequel l'emplacement requis doit être trouvé ; par défaut, c'est le document courant ;

from indique le début de la destination du pointeur, sous forme d'une expression dans la syntaxe du pointeur étendu TEI : par défaut, c'est l'ensemble du document indiqué par l'attribut *doc* ;

to indique le point final de la destination du pointeur, sous forme d'une expression dans la syntaxe du pointeur étendu TEI ; cet attribut ne peut être spécifié que si l'attribut *from* l'a été.

Une spécification complète du langage formel employé pour exprimer la cible des pointeurs étendus TEI dépasserait les limites de ce document ; nous nous bornerons donc à indiquer seulement quelques-unes des caractéristiques les plus généralement utiles. Pour plus de détails, il convient de consulter la version complète des recommandations de la TEI.

Un élément <*xptr*> (ou <*xref*>) peut indiquer l'ensemble d'un autre document simplement en donnant un nom d'entité comme valeur de l'attribut *doc*, comme dans cet exemple :

```
voir <xref doc=P3>The TEI Guidelines, passim</xref>
```

Cet exemple suppose qu'une entité système ou publique avec le nom *P3* a été déclarée. Cette déclaration peut être placée dans le fichier d'extension *litemods.ent* ou effectuée par une autre méthode spécifique au logiciel auteur SGML employé – voir section 15 «Chiffres et Graphiques».

L'attribut *from* est employé pour préciser un emplacement donné au sein du document spécifié par l'attribut *doc*. La spécification utilise un langage spécial, appelé syntaxe du pointeur étendu TEI, dont seulement quelques détails sont donnés ici. Dans ce langage, les emplacements sont définis comme une série de pas *(steps)*, chacun identifiant une certaine partie du document, souvent en termes des emplacements identifiés par le pas précédent. Par exemple, on pointera vers la troisième phrase du deuxième paragraphe du chapitre deux en sélectionnant le chapitre deux dans le premier pas, le deuxième paragraphe dans le deuxième pas, et la troisième phrase dans le dernier pas. Un pas peut être défini en termes de concepts SGML (tels que `parent`, `descendant`, `précédant`, etc.) ou, plus vaguement, en termes de structure de texte, de positions de mots ou de caractères. On peut également utiliser une notation étrangère (non-SGML), ou préciser un emplacement dans un graphique en termes de son système de coordonnées.

Les attributs *from* et *to* utilisent la même notation. Chacun indique une certaine portion du document cible ; le pointeur étendu dans son ensemble indique la section qui commence au début de *from* et qui se termine à la fin de *to*.

La première étape du chemin vers un emplacement sera souvent de spécifier l'identifiant d'un certain élément dans le document cible, comme dans cet exemple :

```
<xptr doc=P3 from='id (SA)'>
```

Ceci sélectionne l'ensemble de l'élément qui porte l'identifiant *SA* à l'intérieur de l'entité *P3*. Si une cible plus fine est nécessaire, d'autres pas peuvent suivre. Les mots clés suivants sont disponibles afin de permettre la sélection d'autres éléments en termes de leur relation avec un élément identifié lors du pas précédent :

child éléments contenus par cet élément ;

ancestor éléments englobant cet élément, directement ou indirectement ;

previous éléments ayant le même parent que cet élément mais situés avant dans le document ;

next éléments ayant le même parent que cet élément mais situés après dans le document ;

preceding
éléments dans le document qui commencent avant cet élément, indépendamment de leurs parents ;

following
éléments dans le document qui commencent après cet élément, indépendamment de leurs parents.

Chacun de ces mots clés implique un ensemble particulier d'éléments (un ensemble d'enfants, un ensemble d'ancêtres, un ensemble de descendants, etc.). Afin de préciser lequel des éléments d'un ensemble est pointé, le mot clé peut éventuellement être suivi d'une liste entre parenthèses contenant :

– un nombre positif ou négatif, indiquant lequel des nombreux éléments éventuellement trouvés est le bon (`+1` indiquant le premier élément rencontré, commençant à l'emplacement actuel, et $-$`1` indiquant le dernier), ou le mot clé `all` (tous) indiquant que tous les éléments dans l'ensemble doivent être pointés ;

– un identifiant générique, indiquant le type d'élément requis, ou une étoile `*` indiquant que n'importe quel type d'élément fait l'affaire ;

– un ensemble de noms et de valeurs d'attribut, indiquant que l'élément sélectionné devrait avoir des attributs ayant les noms et les valeurs spécifiés, s'il en existe.

En reprenant l'exemple ci-dessus, la référence suivante sélectionne le troisième élément <*p*> contenu directement par tout élément ayant l'identifiant *SA* :

```
<xptr doc=P3 from='id (SA) child (3 p)'>
```

De la même manière, en supposant que l'entité *P3* est en fait une référence à la version SGML des recommandations de la TEI, alors la référence suivante sélectionne la section 14.2.2 de cette publication qui se trouve être celle où la syntaxe du pointeur étendu est formellement définie :

```
Pour plus de détails, voir
<ref doc=P3 from='id (SA) child (2 div2) child (2 div3)'>
  TEI Extended pointer syntax definition
</ref>
```

Normalement, l'étendue d'une référence croisée est suffisamment définie par l'attribut *from*. Pour certains documents cependant, il peut être plus commode de définir une plage de début et une plage de fin. Comme indiqué ci-dessus, l'attribut *to* est fourni dans ce but. Par exemple :

```
<xptr doc=P1 from='id (xyz)' to='id (abc)'>
```

est un pointeur étendu dont la cible est la séquence qui commence au début de l'un quelconque des éléments du document *P1* ayant l'identifiant *XYZ*, et se terminant à la fin de n'importe quel élément du même document ayant l'identifiant *ABC*. Tout élément se trouvant dans l'intervalle est également inclus, indépendamment de sa structure. Le pointeur est erroné si la fin de *ABC* précède le début de *XYZ*.

L'utilisation de cette syntaxe permet la construction facile de spécifications très complexes. Par exemple, la référence suivante sélectionne l'élément <*head*> le plus proche ayant un attribut *lang* avec la valeur `LAT`, et apparaissant avant le début de l'élément ayant l'identifiant *SA* :

```
<xptr doc=P3 from='id (SA) preceding (1 head lang lat)'>
```

Si aucune valeur n'est fournie pour l'attribut *doc* , on suppose qu'il s'agit du document courant. Ainsi, les références suivantes sont sémantiquement équivalentes. Les deux indiquent l'élément ayant l'identifiant *X1* dans le document courant :

```
<ptr target=X1>
<xptr from='id  (X1)'>
```

8.3. Attributs de liaison

Les attributs de liaison spécifiques suivants ont été définis pour chaque élément de la DTD *TEI Lite* :

ana relie un élément à son interprétation ;

corresp relie un élément à un ou plusieurs autres éléments correspondants ;

next relie un élément au prochain élément dans un agrégat ;

prev relie un élément à l'élément précédent dans un agrégat.

L'attribut *ana* (analyse) est destiné à être utilisé lorsqu'un ensemble d'interprétations ou d'analyses abstraites a été défini quelque part dans un document, comme nous le verrons de façon plus détaillée en section 16 « Interprétation et analyse. » Par exemple, une analyse linguistique de la phrase « John aime Nancy » pourrait être codé comme suit :

```
<seg type=sentence ana=SVO>
  <seg type=lex ana=NP1>John</seg>
  <seg type=lex ana=VVI>aime</seg>
  <seg type=lex ana=NP1>Nancy</seg>
</seg>
```

Ce codage implique l'existence, ailleurs dans le document, d'éléments ayant les identifiants *SVO*, *NP1*, et *VV1* où la signification de ces codes particuliers est expliquée. On notera l'emploi de l'élément *<seg>* (segment) pour marquer des composants particuliers de l'analyse, différenciés par l'attribut *type*.

L'attribut *corresp* (correspondant) est un moyen simple de représenter une certaine forme de correspondance entre deux éléments dans un texte. Par exemple, dans un texte plurilingue, il peut être employé pour relier les diverses traductions d'un même passage, comme dans l'exemple suivant :

```
<seg lang=FRA id=FR1 corresp=EN1>Jean aime Nancy</seg>
<seg lang=ENG id=EN1 corresp=FR1>John loves Nancy</seg>
```

Le même mécanisme peut être employé à beaucoup d'autres fins. Dans l'exemple suivant, il est employé pour représenter des correspondances anaphoriques entre *the show* et *Shirley*, et entre *NBC* et *network* :

```
<p><title id=shirley>Shirley</title>, which made
its Friday night debut only a month ago, was
not listed on <name id=nbc>NBC</name>'s new schedule,
although <seg id=network corresp=nbc>the network</seg>
says <seg id=show corresp=shirley>the show</seg>
still is being considered.
```

Les attributs *next* et *previous* sont un moyen simple de relier les composants d'un élément discontinu, comme dans l'exemple suivant :

```
<q id=Q1a next=Q1b>Who-e debel you?</q>
&mdash he at last said &mdash
<q id=Q1b prev=Q1a>you no speak-e,
damme, I kill-e.</q>  And so saying,
the lighted tomahawk began flourishing
about me in the dark.
```