

XPath

<http://www.techscore.com/tech/XML/XPath/xpath01-2.html>

<http://www.w3.org/TR/xpath>

Java XPath

XPath のデータモデル

XML データを以下の 7 種類のノードから構成されるツリーと考える。

- ・ルートノード：最上位ノード
- ・要素ノード：XML の要素を表すノード
- ・テキストノード：開始タグと終了タグで挟まれた文字列データ
- ・属性ノード：要素内で指定された属性を表すノード
- ・名前空間ノード：名前空間 (Namespace) を表すノード
- ・処理命令ノード：処理命令 (Processing Instruction) を表すノード (処理命令とは <? と ?> で挟まれた一文のこと)。XML 宣言は処理命令の形をしていますが、XML 1.0 の構文上、処理命令ではないため、XML 宣言は処理命令ノードとしては扱いません。
- ・コメントノード：コメントを表すノード (ただし、ドキュメント型宣言内に記述するコメントを除きます)

コンテキストノード

- ・通過点を指定することにより目的のデータに至る経路のことをロケーションパスという

```
aaa
+ bbb
  + ccc
    + ddd
```

という構造を bbb->ccc->ddd とたどるとき、起点となる aaa をコンテキストノード、bbb/ccc/ddd をロケーションパスという

ロケーションパス (簡略な記述法)

- ・ロケーションパスの書き方には、厳密かつ詳細に記述する書き方と、簡略に記述する書き方の 2 つがある
- ・以下の構文で示されたものを 1 個以上 "/" で繋いだものがロケーションパスとなる。
- ・ひとつのロケーションパス式の評価が終わるとコンテキストノードは最初にあったノードに戻る

構文	意味
para	コンテキストノードの子要素 para を選択
*	コンテキストノードのすべての子要素を選択
text()	コンテキストノードのすべての子テキストノードを選択
@name	コンテキストノードの name 属性を選択
@*	コンテキストノードのすべての属性を選択
para[1]	コンテキストノードの 1 番目の子要素 para を選択
para[last()]	コンテキストノードの最後の子要素 para を選択

//para	ルートノードの子孫ノードから要素ノード para をすべて選択。つまり、コンテキストノードと同じ文書内のすべての para 要素を選択
.(ピリオド)	コンテキストノードを選択
..(ダブルピリオド)	コンテキストノードの親を選択
chapter[title="Introduction"]	コンテキストノードの子要素 chapter のうち、文字列値が Introduction である子要素 title を 1 個以上持つものを選択
chapter[title]	コンテキストノードの子要素 chapter のうち、子要素 title を 1 個以上持つものを選択
employee[@secretary and @assistant]	secretary 属性と assistant 属性の両方をもつコンテキストノードの子要素 employee をすべて選択

ロケーションパス

ロケーションパスは "/" で区切られた 1 つ以上のロケーションステップと呼ばれる単位からなっており、以下の 3 つのパートからなる。

パート	説明
軸	選択するノードとコンテキストノードとの間のツリー上の位置関係を指定する
ノードテスト	選択するノードのノード型と名前を指定する
述語	選択するノードの集合を、任意の式を使用してさらに細かく選別する

ここまで、メモした

<http://www.techscore.com/tech/XML/XPath/xpath03.html>

XPath の基本

```
<Media>
  <Book Author="a1" Title="t1"/>
  <Book Author="a2" Title="t2"/>
  <Book Author="a3" Title="t3"/>
  <CD Artist="a4" title="t4"/>
  <DVD Directory="d1" Title="t5">
    <Actor>A1</Actor>
    <Actor>A2</Actor>
  </DVD>
</Media>
```

- ・「/」は最も一般的な演算子で目的の要素へのパスを構築できる (eg:Media/CD)
 - ・ Media/Book を選択すると以下が生成される

```
<Book Author="a1" Title="t1"/>
<Book Author="a2" Title="t2"/>
<Book Author="a3" Title="t3"/>
```

- ・ XPath によって、ノードのリストまたは単一のノードが生成されるという考え方は XML バインディングを学ぶ上で極めて重要

- ・ XML では要素の属性の両方が XML ノードとみなされる
- ・ XPath は実際には要素だけでなくノードを選択することによって機能する
- ・ 属性名を参照するには「@」を演算子を使用する
 - ・ Media/Book/@Title を使用すると、次の内容が返る (XmlAttributeNode 型)

```
Title="t1"
Title="t2"
Title="t3"
```

- ・ 「*」演算子を使用すると、任意の名前付きノード（属性または要素）を取得できる
- ・ 「[]」演算子を使用すると、位置または属性によってノードを選択できる（インデックス 1 ベース）
 - ・ Media/Book[1] を選択すると以下が生成される

```
<Book Author="a1" Title="t1"/>
```

- ・ 属性による選択
 - ・ Media/Book[@Author="t1"]

```
<Book Author="a1" Title="t1"/>
```

<u>XPath</u>	説明	例
/	ルート以下を選択	/
//NAME	任意の子孫の NAME というタグにマッチ	//Book
@NAME	NAME という属性にマッチ	//Book/@ Author
*	任意のタグにマッチ	//*/@ Author
@*	任意の属性にマッチ	//Book/@*
NAME	NAME というタグにマッチ	/Media
[]	位置または属性によって子タグを選択	//Book[1],//Book[@Author='a1']