XML子要素の展開例1

2023-11-29
行方向の展開と列方向への展開サンプル2例組み込み

希望の結果

custommersの子要素群を行方向に、optsの子要素群を列方向に展開する。

この例だと、

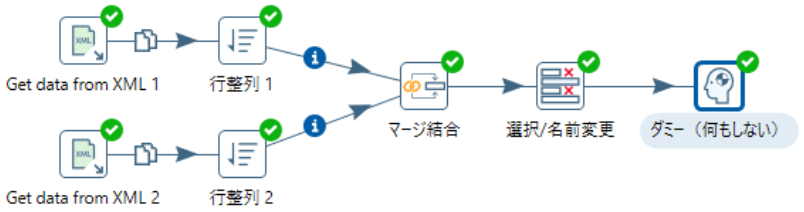
- 要素recsの子要素であるrecの単位で出力する。
- 要素custommersの子要素であるcustommerの数だけ要素recの行数が増える。
- 要素optsの子要素であるoptの個々を独立した項目として扱う。最大3としたので項目数も3個追加された形になる。

入力XML

```
<masterdata>
  <recs>
      <rec>
        <prodid>1000</prodid>
        <prodname>prod A</prodname>
        <custommers>
          <custommer><name>cust
01</name><opts><opt>opt-A</opt></opts>
          □□□□□□□□□□□□□□□□□□ <addr>addr
1</addr></custommer>
          <custommer><name>cust
02</name><opts><opt>opt-A</opt>
<opt>opt-B</opt></opts>
<addr>addr 2</addr></custommer>
          <custommer><name>cust
10</name><opts><opt>opt-A</opt>
<opt>opt-C</opt></opts>
<addr>addr 10</addr></custommer>
        </custommers>
      </rec>
      <rec>
        <prodid>1100</prodid>
        <prodname>prod B</prodname>
        <custommers>
          <custommer><name>cust
02</name><opts></opts>
<addr>addr 2</addr></custommer>
          <custommer><name>cust
20</name><addr>addr
20</addr></custommer>
        </custommers>
      </rec>
      <rec>
        <prodid>9000</prodid>
        <prodname>prod Z</prodname>
        <custommers>
        </custommers>
      </rec>
      <rec>
        <prodid>2000</prodid>
        <prodname>prod F</prodname>
        <custommers>
          <custommer><name>cust
52</name><opts><opt>opt-A</opt>
<opt>opt-B</opt>
<opt>opt-C</opt></opts>
<addr>addr 52</addr></custommer>
          <custommer><name>cust
55</name><opts></opts></custommer>
        </custommers>
      </rec>
  </recs>
</masterdata>
```

出力□CSV想定）

prodid	prodname	custommer	opt1	opt2	opt3	addr
1000	prod A	cust 01	opt-A			addr 1
1000	prod A	cust 02	opt-A	opt-B		addr 2
1000	prod A	cust 10	opt-A	opt-C		addr 10
1100	prod B	cust 02				addr 2
1100	prod B	cust 20				addr 20
9000	prod Z					
2000	prod F	cust 52	opt-A	opt-B	opt-C	addr 52
2000	prod F	cust 55				



実行結果

ログ | 実行履歴 | 実行状況 | パフォーマンスグラフ | メトリクス | データをプレビュー

開始行 ○最終行 ○Off

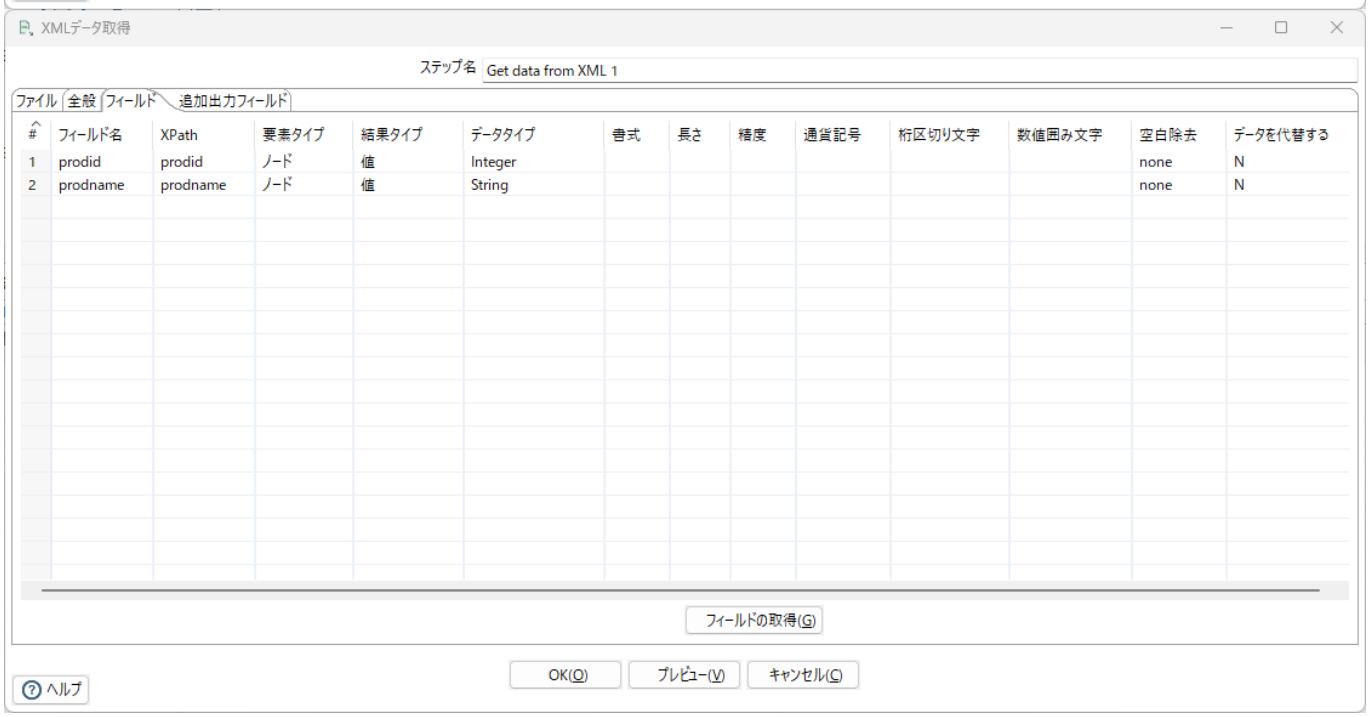
#	prodid	prodname	custommer	opt1	opt2	opt3	addr
1	1000	prod A	cust 01	opt-A			addr 1
2	1000	prod A	cust 02	opt-A	opt-B		addr 2
3	1000	prod A	cust 10	opt-A	opt-C		addr 10
4	1100	prod B	cust 02				addr 2
5	1100	prod B	cust 20				addr 20
6	2000	prod F	cust 52	opt-A	opt-B	opt-C	addr 52
7	2000	prod F	cust 55				
8	9000	prod Z	<null>	<null>	<null>	<null>	<null>

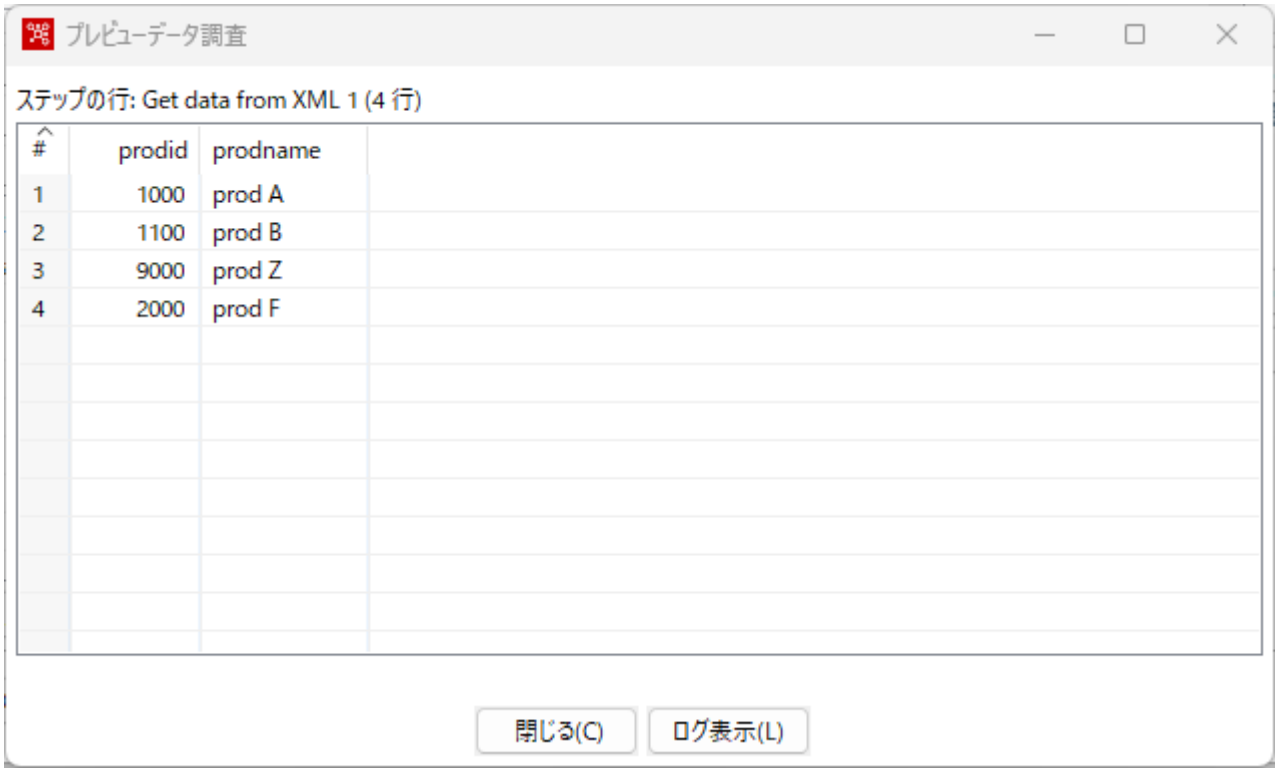
定義

Get data from XML 1

要素recがレコードに変換されるので4レコード生成される事になる。


XMLデータ取得
ステップ名
Get data from XML 1
ファイル
全般
フィールド
追加出力フィールド
設定
ループ XPath
/masterdata/recs/rec
XPathを取得
文字コード
UTF-8
ドキュメントの名前空間を使用する
コメントを無視する
XMLを検証する
トークンを使用する
空のファイルを無視する
ファイルの有無を無視する
エラーを許容する回数
0
大容量データを処理するXPath
拡張設定
ファイル名を出力に含む
フィールド名
レコード番号を出力に含む
フィールド名
オプション設定
ファイル名を結果に含む
ヘルプ
OK(O)
プレビュー(V)
キャンセル(C)


XMLデータ取得
ステップ名
Get data from XML 1
ファイル
全般
フィールド
追加出力フィールド
フィールド名 XPath 要素タイプ 結果タイプ データタイプ 書式 長さ 精度 通貨記号 桁区切り文字 数値囲み文字 空白除去 データを代替する
1 prodid prodid ノード 値 Integer none N
2 prodname prodname ノード 値 String none N
フィールドの取得(G)
ヘルプ
OK(O)
プレビュー(V)
キャンセル(C)



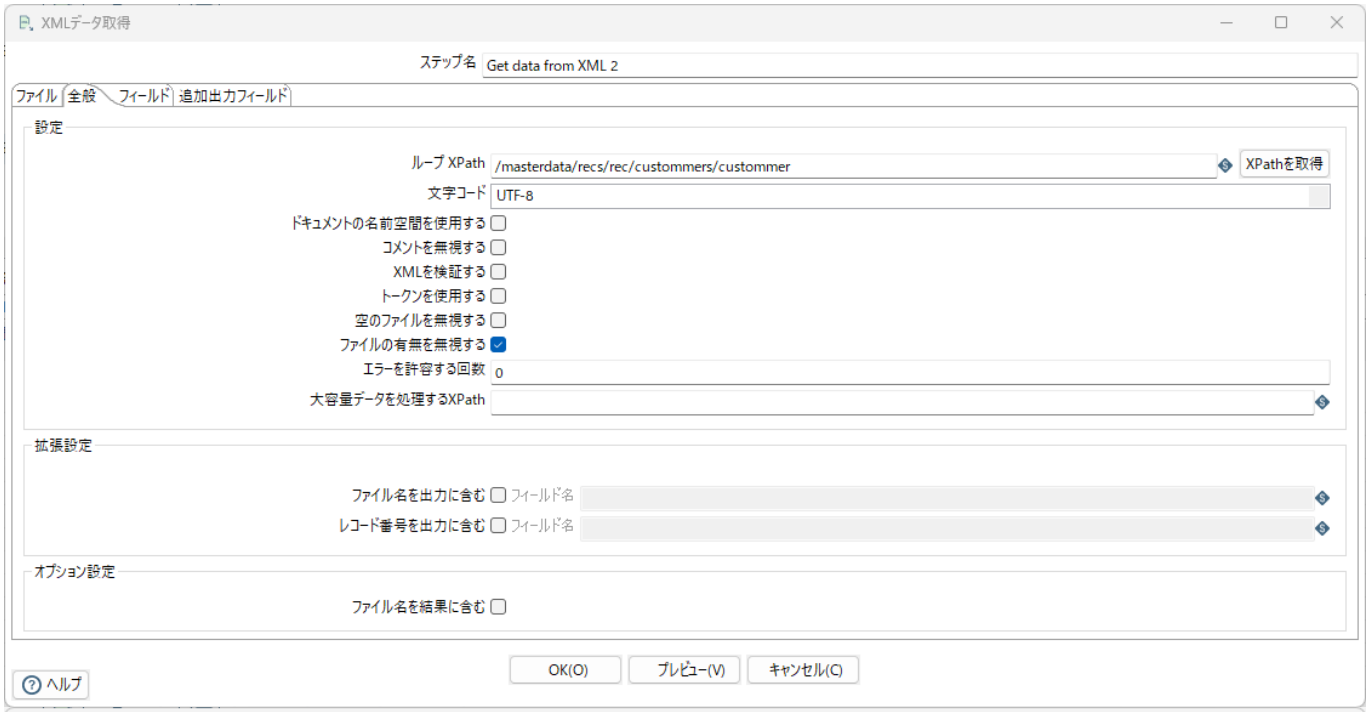
行整列1

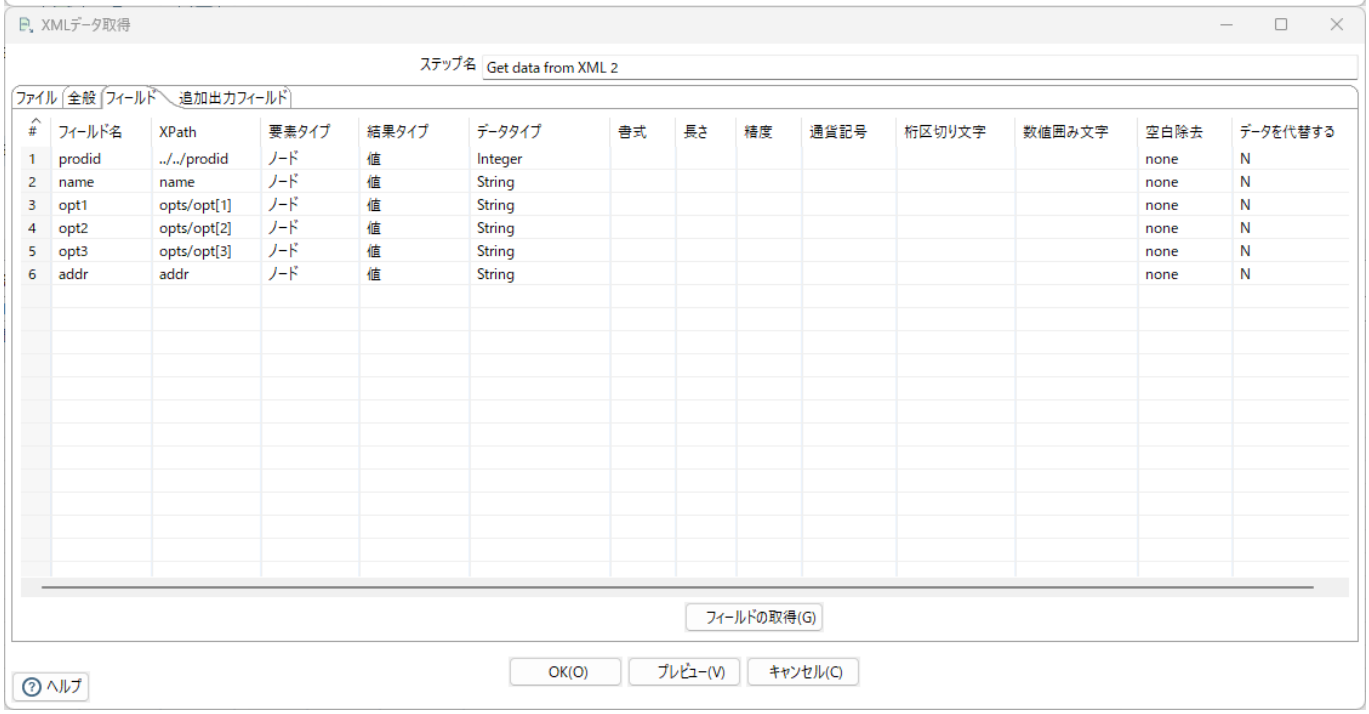
Get data from XML 2

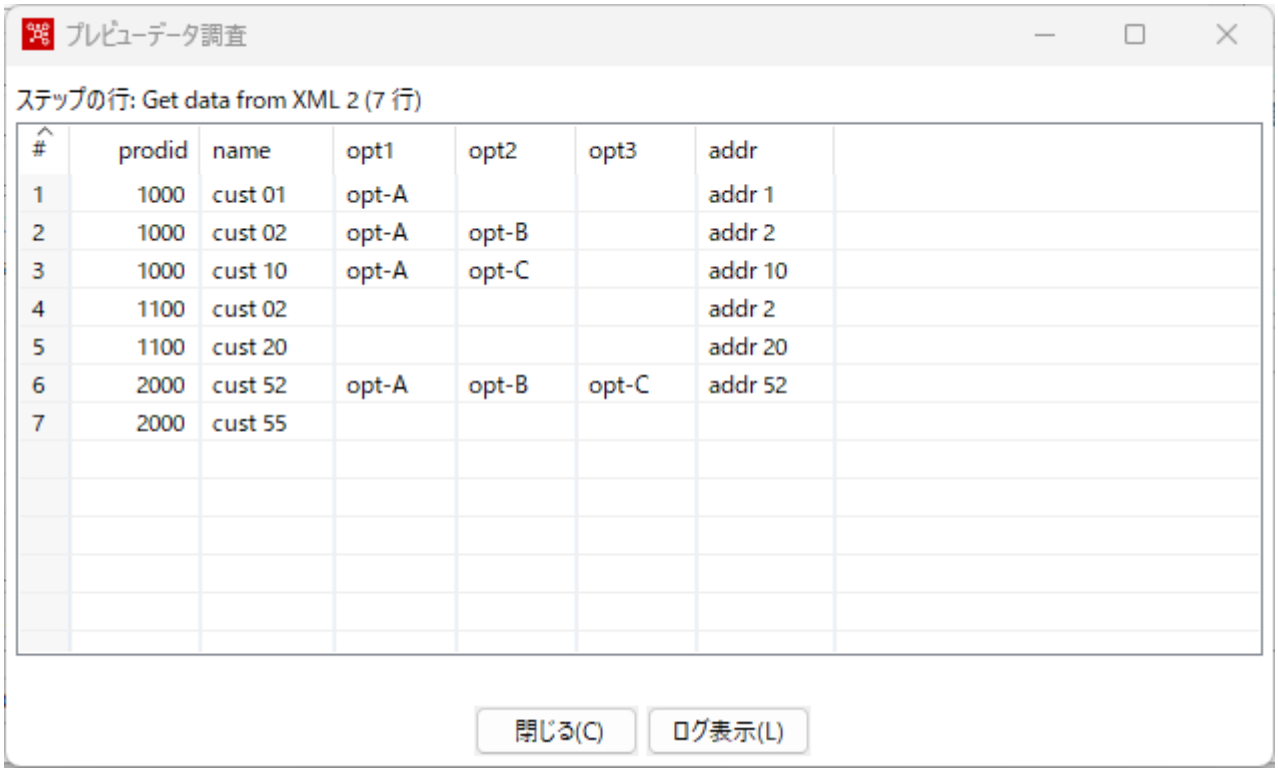
XPath指定の意のようで、相対パス指定等を利用できる。

- 要素custommerのレコードに変換されるので7レコード生成される事になる。

- 親要素を辿ってprodidを取得。
- 要素optsの中身は添え字付きで指定し独立した項目として扱う。
  ※最大3個□COBOLでいうところのOCCURS句で3回の繰り返しとしている。





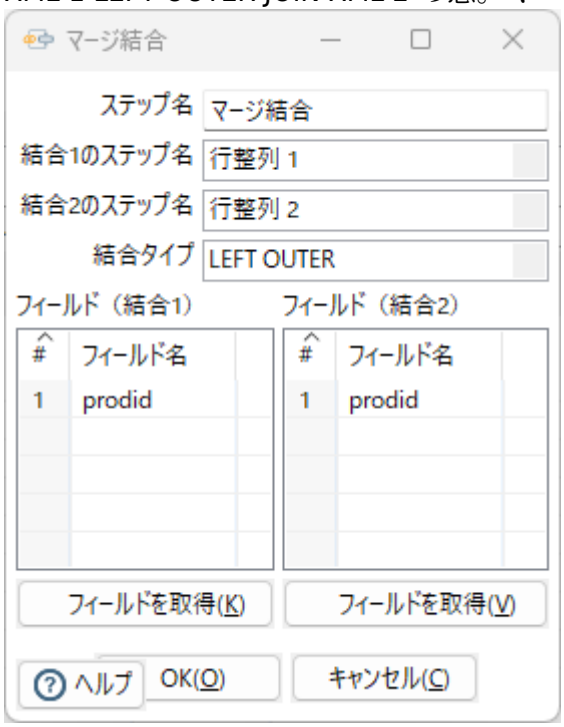


行整列 2

マージ結合

Tipsにも記述があるけど入力は適切なソートが適用されている必要がある。

XML 1 LEFT OUTER JOIN XML 2 の意。**キーはprodid**



選択/名前変更

キー項目名**が(**この例だとprodid)**が必ず**使用**される**前提**を付ける事ができるなら、マージ**結合**で**生成**される** prodid_1 **は**削除**できる。

技術資料, etl, pentaho, xml, csv, xpath