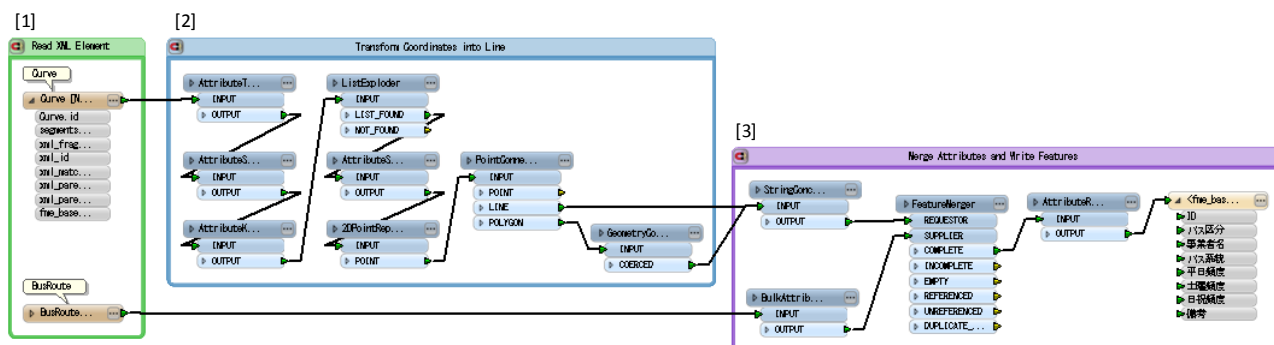


FME ワークスペースサンプル <フォーマット変換>
 国土数値情報 GML 形式「バスルート」の変換



概要

- 国土数値情報 GML 形式「バスルート」を読み込み、GIS で利用可能なデータ形式に変換して出力する。

入力

- 国土数値情報 GML 形式「バスルート」

出力

- バスルート（ライン）を ESRI Shape 形式で出力する。

国土数値情報のデータは GML 形式とあわせて ESRI Shape 形式でも提供されているので実用的な例ではありませんが、製品仕様書が公開されておりデータ構造を確認し易いため、敢えてとりあげました。

国土数値情報 GML 形式「バスルート」ドキュメント抜粋（埼玉県の例）
 （空間属性記述部分） (主題属性記述部分)

```
<gml:Curve gml:id="cv1">
  <gml:segments>
    <gml:LineStringSegment>
      <gml:posList>
        35.79235344 139.75762312
        35.79190273 139.75714944
        35.79190247 139.75714916
        35.79190222 139.75714889
        35.79137998 139.75657675
        35.79137995 139.75657672
```

～ 以下略 ～

```
<ksj:BusRoute gml:id="br1">
  <ksj:brt xlink:href="#cv1"/>
  <ksj:bsc>1</ksj:bsc>
  <ksj:boc>秋北バス（株）</ksj:boc>
  <ksj:bln>ジュピター号</ksj:bln>
  <ksj:rpd>1.0</ksj:rpd>
  <ksj:rps>1.0</ksj:rps>
  <ksj:rph>1.0</ksj:rph>
  <ksj:rmk></ksj:rmk>
</ksj:BusRoute>
```

～ 以下略 ～

キーワード

FME ワークベンチ	FME Workbench	FME の中核であるソフトウェアの名称
ワークスペース	Workspace	FME ワークベンチで作成されたデータ処理フロー
リーダー	Reader	データ読み込みを担うワークスペース構成要素
トランスフォーマー	Transformer	データ変換を担うワークスペース構成要素
ライター	Writer	データ出力を担うワークスペース構成要素
フィーチャー	Feature	地物を表す幾何図形とその属性、テーブルの 1 レコードなど、処理対象とする事物を表すひとかたまりのデータ

[1] XML データの読み込み

FME で XML データを読み込む方法はいくつかありますが、データ構造が比較的単純な場合には、必要なデータが記述されている XML 要素名をリーダーのパラメータとして指定し、それをフィーチャータイプとする方法 (Feature Paths) が推奨されています。ここでもその方法を採用し、バスルートのラインを構成する頂点座標リストが記述されている <Curve> 要素と、バスルートの主題属性を記述している <BusRoute> 要素をフィーチャータイプとしました。

指定した XML 要素の属性と下位の XML 要素 (とその属性) はフィーチャの属性として扱われ、フィーチャの属性名は、XML 要素名 / 属性名が XML ドキュメントの構造に従ったパスの順で「.」区切りで連結された文字列となります。

Curve フィーチャの主な属性

Curve.id (ラインの id)

segments.LineStringSegment.posList (ラインの頂点座標リスト)

BusRoute フィーチャの主な属性

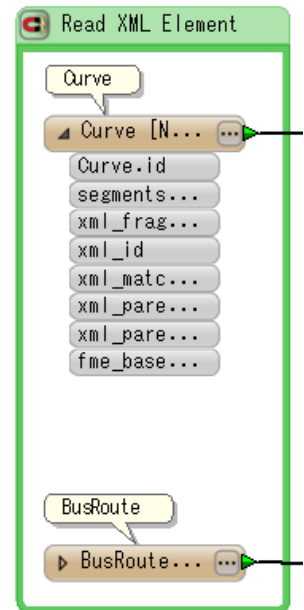
brt.href (参照先ラインの id。Curve.id の先頭に # を付加)

bsc (バス区分)

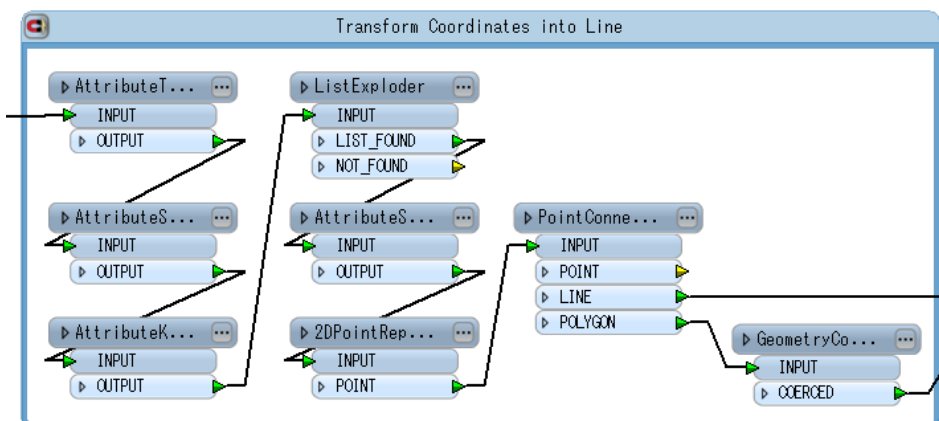
boc (事業者名)

bln (バス系統)

...



[2] 頂点座標リストのラインへの変換

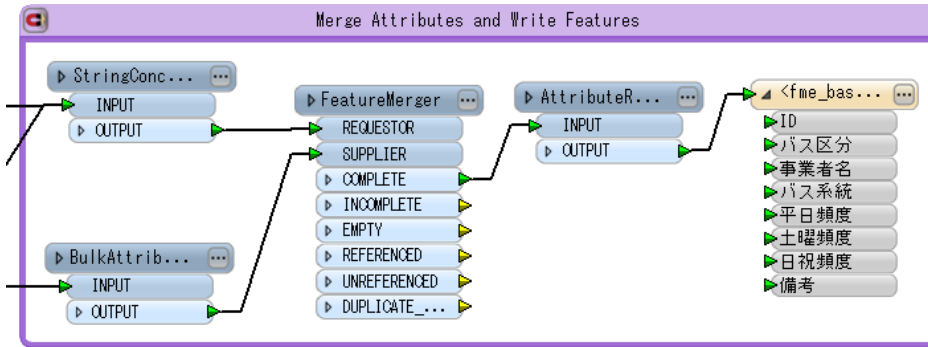


Curve フィーチャの segments.LineStringSegment.posList 属性には、<gml:posList> 要素の値、すなわち、バスルートのラインを構成する頂点の緯度、経度のペア (スペース区切り) のリスト (改行区切り) が格納されています。これをジオメトリ (幾何図形データ) に変換するために次のトランスフォーマーを使います。

"AttributeTrimmer"	頂点座標リスト全体の前後の空白類文字を取り除く
"AttributeSplitter"	改行で分割して緯度、経度ペアを要素とするリスト属性に変換する
"AttributeKeeper"	以後の処理に必要な属性のみを残す
"ListExploder"	緯度、経度ペアひとつずつを属性として持つフィーチャに分解する
"AttributeSplitter"	緯度、経度ペアを空白で分割して緯度と経度を要素とするリスト属性に変換する
"2DPointReplacer"	緯度と経度に基づいてポイントジオメトリを作成する
"PointConnector"	Curve.id ごとにポイントを連結したジオメトリを作成する

"PointConnector" は、連結するポイントの終点と始点の位置が異なるときはラインジオメトリ、同じとき (閉じられているとき) にはポリゴンジオメトリを作成します。バスルートが循環路線である場合はポリゴンが作成されるので、それは "GeometryCoercer" によってラインに変換します。

[3] 主題属性の結合とデータ出力

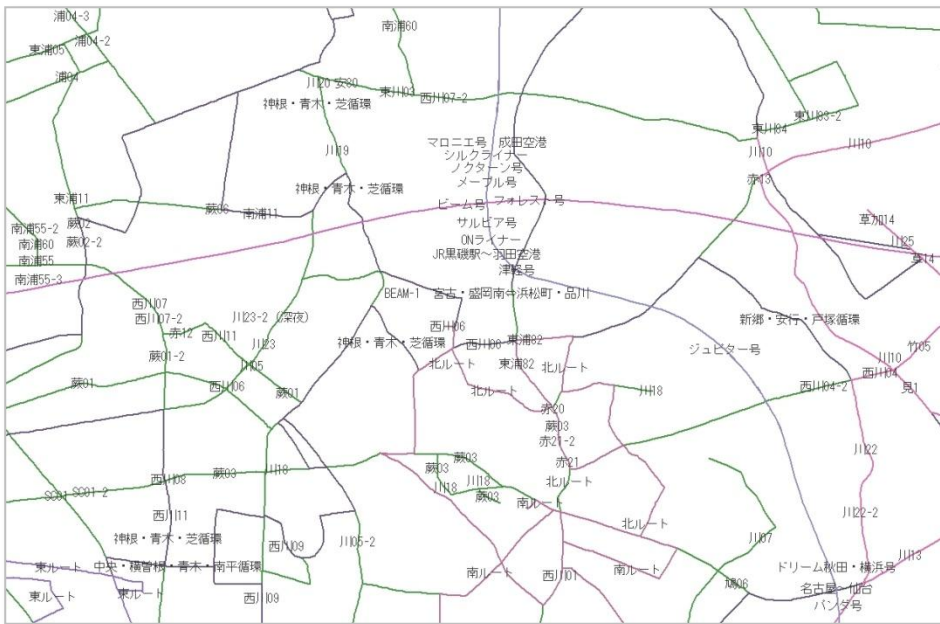


GML 形式の国土数値情報では、主題属性を記述している XML 要素から空間属性を記述している XML 要素を参照する際の id は先頭に「#」をつけた書式となっています。そのため、"StringConcatenator"によって[2]で作成したラインジオメトリの Curve.id 属性値の先頭に「#」をつけておきます。

また、<BusRoute>フィーチャーからは"BulkAttributeRemover"によって不要な属性を削除しておきます。ここで不要な属性を削除するのは必須ではありませんが、処理の効率化のためには望ましいこともあります。属性の削除は"AttributeRemover"でもできますが、属性名に共通のプレフィクス(この場合は xml_)がついている属性をまとめて削除する場合には"BulkAttributeRemover"が便利です。

これらの準備をしてから、"FeatureMerger"によって<BusRoute>フィーチャーの属性をラインジオメトリに結合します。

"AttributeRenamer"によって属性名を分かり易い名前に変更して Shape 形式ファイル出力用のライターを接続すれば、ワークスペースは完成です。



処理結果の表示例（埼玉県の一部。ESRI ArcMap により作成）
事業者別にラインを色分けし、バス系統をラベルで表示しました。

