実習 8 ジオメトリの座標系変換と計測・クリッピング

これまでの実習と同じ市区町村区域データ（Esri Shapefile 形式ポリゴン）に、新たな属性として、「市区町村面積」(km²)、「農用地区域面積」(km²)、市区町村面積に占める農用地区域面積の「率」(%)を付加するためのワークスペースを作成します。

農用地区域の面積を求めるために、国土数値情報の農業地域データ（平成 27 年度版）に含まれる農用地区域データ（Esri Shapefile 形式ポリゴン）を使用します。

作成するワークスペースは、ひとつの都道府県の区域を対象とすることにします。以下、千葉県を例として説明しますが、他の都道府県について行っても構いません。
実行結果は FME Data Inspector で確認します。

はじめに、国土数値情報ダウンロードサービスサイト（JPGIS2.1）より千葉県の農業地域データ（平成 27 年度版）をダウンロード、解凍し、"a001120020160206.shp"（他の都道府県の場合はファイル名が異なります）が展開されたことを確認してください。

国土数値情報ダウンロードサービスサイト URL
http://nlftp.mlit.go.jp/ksj/

ソースデータ

mmm20151001.shp	Esri Shapefile 形式 市区町村区域ポリゴン
a001120020160206.shp	Esri Shapefile 形式 農用地区域（平成 27 年度千葉県）ポリゴン

国土数値情報農業地域データ（平成 27 年度版）には「農業地域」と「農用地区域」の 2 つの Shapefile データセットが含まれていますが、この実習では、「農用地区域」"a001120020160206.shp "のみを使用します。Shapefile 形式の国土数値情報農業地域データ（平成 27 年度版）の内容やファイル名の命名規則については、国土数値情報ダウンロードサイトで確認してください。

今回作成するワークスペースは、次の 5 つの部分で構成されます。

1) 市区町村ポリゴンを読み込み、千葉県内の市区町村を抽出する。

2) 市区町村ごとにその区域の面積を求める（「市区町村面積」属性作成）。

3) 農用地区域ポリゴンを読み込み、市区町村の境界で農用地区域を分割する（クリッピング）。

4) 市区町村ごとに農用地区域の面積を求める（「農用地区域面積」属性作成）。

5) 市区町村ポリゴンに「農用地区域面積」を結合し、市区町村面積に占める農用地区域の面積の率を求める（「率」属性作成）。

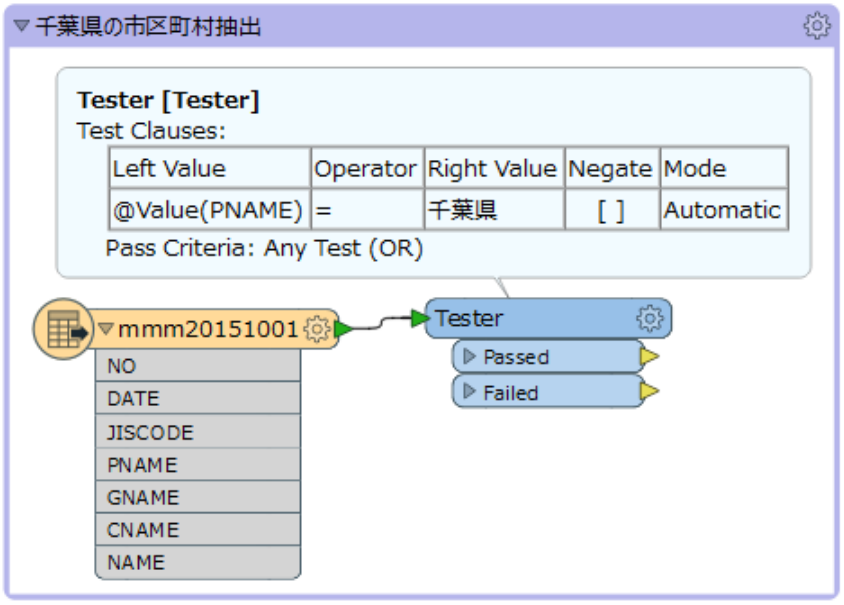
面積は、ソースデータに記録されているジオメトリに基づいて計算します。

ジオメトリに基づいて長さや面積を計測する場合、データ上の座標値の単位は m などの距離の単位でなければなりませんが、市区町村区域、農用地区域ともに、ソースデータは JGD2000 緯度経度（座標値の単位は「度」）で作成されているので、そのままでは計測できません。

このワークスペースでは、ポリゴンの面積を求めるために、ジオメトリの座標系を平面直角座標第 IX 系（単位 m）に変換することとします。

8-1. 千葉県の市区町村ポリゴン抽出

ワークスペースに市区町村ポリゴンを読み込むためのリーダーとリーダーフィーチャータイプを追加し、千葉県の市区町村を抽出するために Tester を接続します。

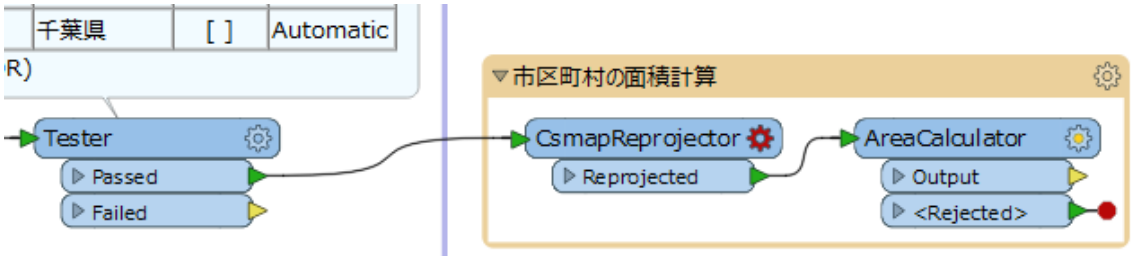


8-2. 市区町村ポリゴンの面積計算

ポリゴンの面積は AreaCalculator で求めることができますが、前述したように、ソースデータの座標系は JGD2000 緯度経度であるため、面積を求める前に、投影座標系に変換する必要があります。

データフローの途中でジオメトリの座標系を変換するには、CsmapReprojector（または Reprojector）が使用できます。

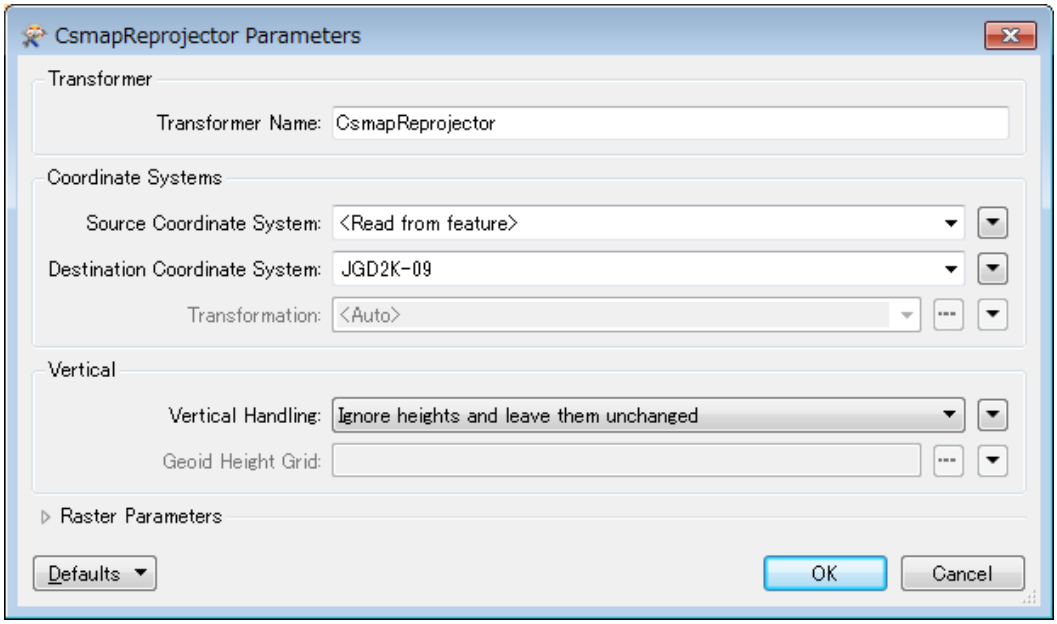
ワークスペースに CsmapReprojector と AreaCalculator を追加し、Tester の Passed ポートに次のように接続してください。



(1) 座標系変換 CsmapReprojector

CsmapReprojector（または Reprojector）により、データフローの途中でジオメトリの座標系（測地系及び投影法）を変換することができます。

Source / Destination Coordinate System パラメーターで、変換前 / 後の座標系を指定します。



Source Coordinate System: <Read from feature>（フィーチャーから読み取る）

ソースデータにおいて座標系が定義されている場合には、リーダーによって読み込まれたフィーチャーにもその定義が引き継がれます。その場合、このパラメーターに明示的に座標系を指定する必要はなく、<Read from feature>（デフォルト）とすることができます。

Esri Shapefile 形式の場合、データの座標系は"*.prj"ファイルで定義されます。市区町村ポリゴンデータには "*.prj" ファイルによって座標系（JGD2000 緯度経度）が定義されているので、ここでは<Read from feature>としました。

Destination Coordinate System: JGD2K-09

FME は世界中の標準的な座標系をサポートしており、それぞれに固有の識別子を定義しています。日本の主な座標系については次のように定義されています。

識別子	測地系・投影法
LL-JGD2011_FME	JGD2011 緯度経度
JGD2011-**_FME	JGD2011 平面直角座標第**系（**は 01～19）
LL-JGD2K	JGD2000 緯度経度
JGD2K-**	JGD2000 平面直角座標第**系（**は 01～19）
JGD2K.PlnRctCS-**	JGD2000 平面直角座標第**系（**は I～XIX） "JGD2K-**"と同じ。
Tokyo	Tokyo Datum（旧日本測地系）緯度経度
Tokyo.PlnRctCS-**	Tokyo Datum（旧日本測地系）平面直角座標第**系（**は I～XIX）

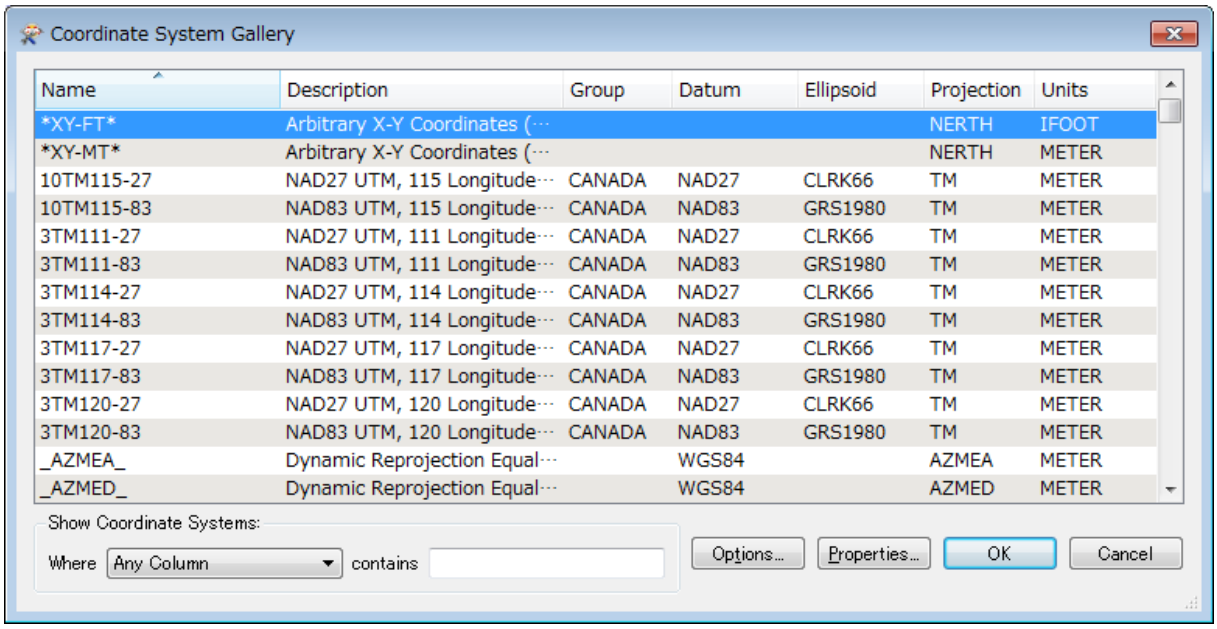
Source / Destination Coordinate System 入力フィールド右端の▼ボタンをクリックするとドロップダウンリストが開き、最近使用した座標系識別子が表示されます。その中に必要なものがあれば選択し、なければリスト最下段の More Coord Systems（その他の座標系）から Coordinate System Gallery（座標系ギャラリー）を開いて検索、選択できます（次ページ参照）。

ここでは、変換後の座標系を"JGD2K-09"（JGD2000 平面直角座標第 IX 系）としました。これによ

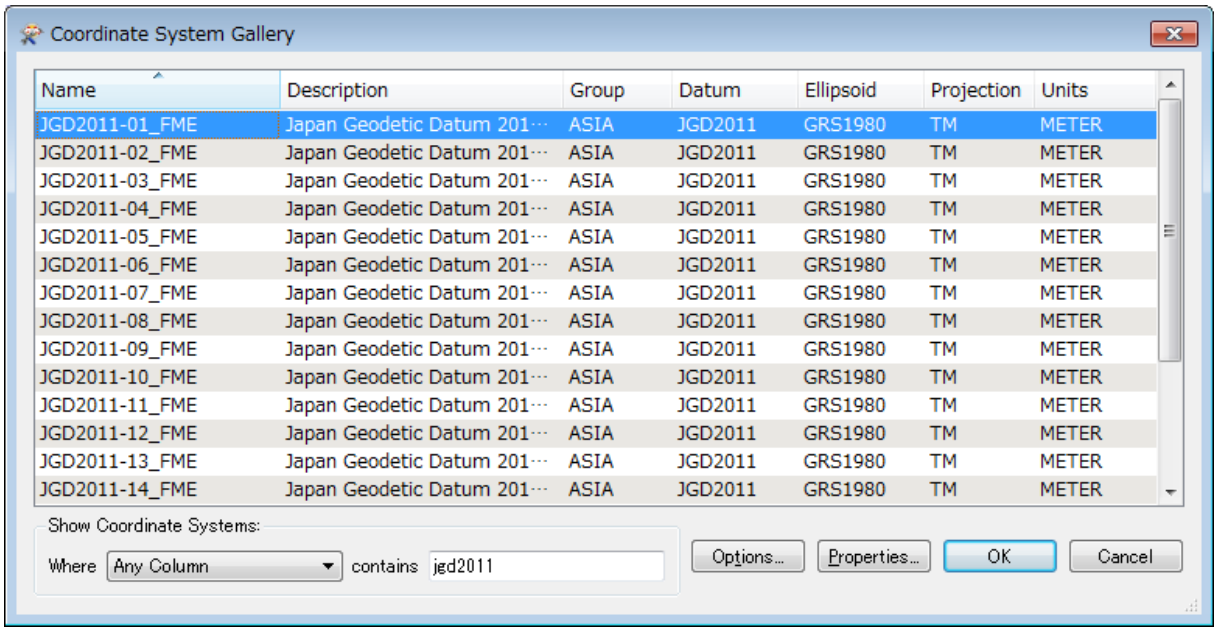
りジオメトリが投影変換されて座標値の単位が m となり、面積が計算できるようになります。

Coordinate System Gallery（座標系ギャラリー）

CsmapReprojector などの座標系に関する処理を行うトランスフォーマーやリーダー/ライターのパラメーターとして座標系を設定する場面では、パラメーター入力フィールド右端の▼ボタンをクリックして表示されるドロップダウンリストで More Coord Systems を選択して Coordinate System Gallery を開き、FME がサポートする全座標系の中から必要なものを検索、選択することができます。



画面下側の contains ボックスに文字列を入力すると、その文字列を含む座標系に絞り込まれます。例えば "jgd2011" と入力すると、JGD2011 に関するものに絞り込まれます。



使用する座標系の行を選択して OK で閉じれば、その座標系がパラメーターに設定されます。

Workbench メニュー Tools > Browse Coordinate Systems を選択すると Coordinate System Galley が開き、FME がサポートしている全ての座標系を閲覧できます。

(2) 面積計算 AreaCalculator

AreaCaluclator により、ポリゴンなど面を持つジオメトリの面積を計算し、その結果をフィーチャーに属性として付加することができます。

Type: Plane Area

ジオメトリを XY 平面に投影した形状の面積を求める場合は Plane Area、ジオメトリが Z 座標値を持っており、そのジオメトリを含む平面（XY 平面に対して傾いている場合もある）における面積を求める場合は Sloped Area を指定します。

Area Attribute: 市区町村面積

求めた面積を格納する属性名を指定します。

Multiplier: 1e-6

面積はジオメトリに設定されている座標系における座標値の距離の単位で計算されますが、このパラメーターで、計算結果を属性に格納するときに乗じる値を指定できます。

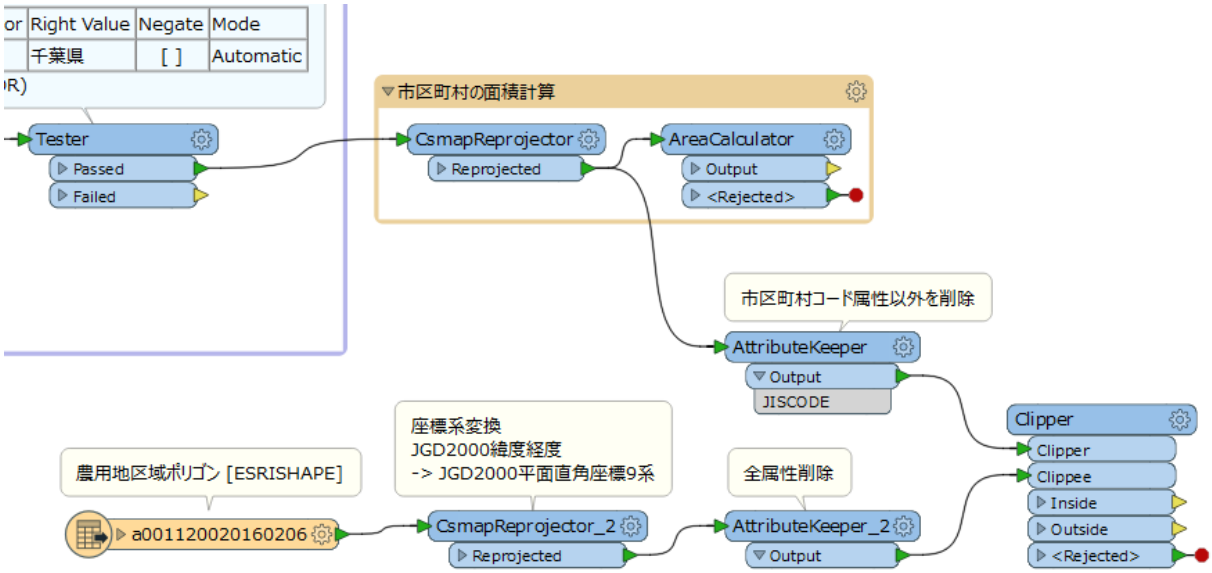
平面直角座標第 IX 系の座標値の単位は m なので、求められる面積の単位は m^2 ですが、このパラメーターに 0.000001 または 1e-6（10^{-6} の意味）を指定することにより、属性値としては km^2 単位に換算された値が格納されます。

AreaCalculator は、入力フィーチャーのジオメトリがシングルパートであるかマルチパートであるかに関わらず、フィーチャー単位で面積を求めます。
ソースデータとして利用している市区町村ポリゴンデータは、飛び地や離島を含めて市区町村の単位で集約されたマルチパートポリゴンなので、AreaCalculator で市区町村全域の面積が求められます。
飛び地などが分離されたシングルパートポリゴンである場合は、面積を求める前に市区町村ごとに集約（マルチパート化）するか、または、個々のパートの面積を求めてから市区町村ごとに合計を求める必要があります。

8-3. 市区町村ポリゴンによる農用地区域ポリゴンのクリッピング

このステップでは、市区町村ポリゴンによって農用地区域ポリゴンのクリッピング（市区町村境界による農用地区域ポリゴンの分割）を行います。

このステップまでの最終形は、次の図のようになります。上段に見えている Tester, CsmapReprojector, AreaCalculator は、前節までに追加済みのものです。



ジオメトリのクリッピングを行うには、Clipper トランスフォーマーを使用します。
Clipper には次の 2 つの入力ポートがあります。

1) Clipper ポート: クリップ「する」側のフィーチャー（市区町村ポリゴン）を入力する

上段の CsmapReprojector の Reprojected ポートに接続しているラインが、2 方向に分岐していることに注目してください。オブジェクトの出力ポート直後でデータフローを複数の方向に単純に分岐させた場合、実行時、そこで各フィーチャーが分岐の数だけコピーされ、それぞれのフローに送り出されます。
この図の場合、上段の CsmapReprojector の Reprojected ポートから出力された市区町村ポリゴンフィーチャーは 2 個ずつになり、一方は AreaCaltulator へ送られ、もう一方が Clipper の Clipper ポートへ送られます。
AttributeKeeper では、市区町村を識別できる最小限の属性（JISCODE=市区町村コード）以外を削除しました。

2) Clippee ポート: クリップ「される」側のフィーチャー（農用地区域ポリゴン）を入力する

農用地区域ポリゴン（千葉県）を読み込むためのリーダーとリーダーフィーチャータイプ（図の左下）を追加し、CsmapReprojector_2 で市区町村ポリゴンと同じ座標系（JGD2000 平面直角座標第 IX 系）に変換したうえで、Clippee ポートに送ります。

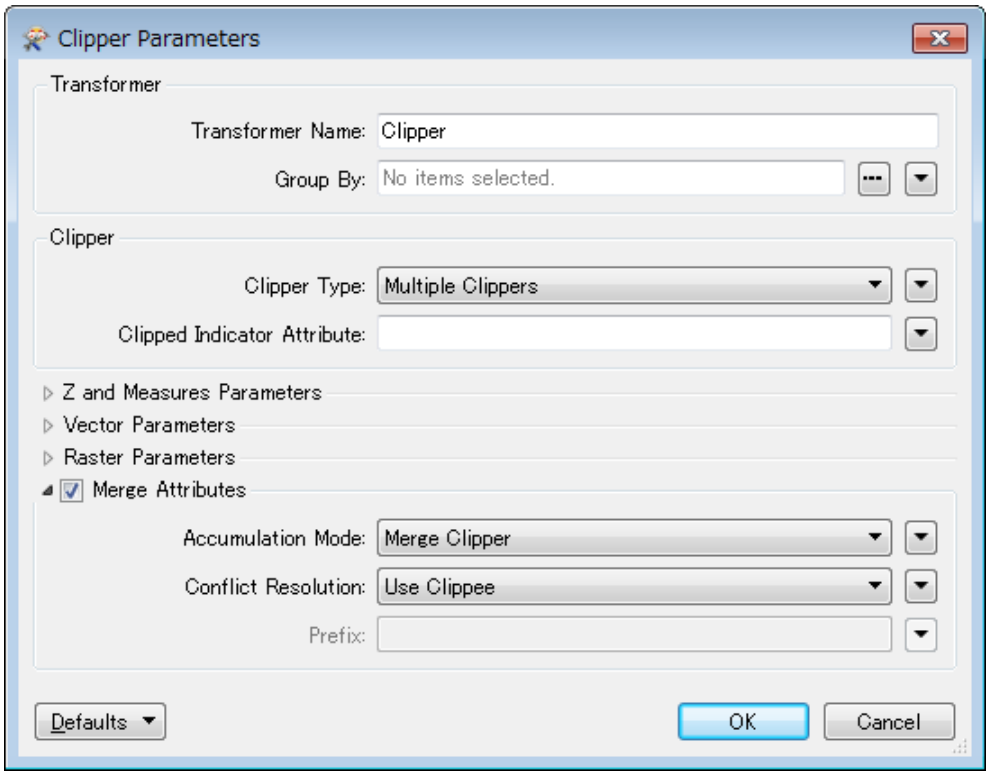
Clipper フィーチャーと Clippee フィーチャーの座標系が異なる場合は、このように、クリッピングを行う前に統一しておく必要があることに注意してください。

このワークスペースでは農用地区域のジオメトリから面積を求めるだけなので、Clippee ポートに接続する前に AttributeKeeper_2 で全ての属性を削除しました（Attributes to Keep パラメーターに何も設定しない）。

Clipper は、Clippee フィーチャー（農用地区域）のジオメトリを、Clipper フィーチャー（市区町村）のジオメトリ（面）の境界によって分割し、Clipper の内側の部分を Inside ポート、外側の部分を Outside ポートから出力します。

Clipper のパラメーターは、次のように設定します。

Clipped Indicator Attribute: (空欄)
Merge Attributes チェックボックス: オン
その他のパラメーターはデフォルトのまま



Clipped Indicator Attribute には、Inside 及び Outside ポートから出力されるフィーチャーに Clipper によって分割されたかどうかを示す属性（yes/no）を付加する必要がある場合に、その属性名を指定します。農用地区域ポリゴンが分割されたかどうかを知る必要はないので空欄としました。

Merge Attributes チェックボックスをオンにすると、Inside ポートから出力されるフィーチャーには、それと重なる Clipper フィーチャーの属性が付加されます。

千葉県の農用地区域を千葉県の市区町村の区域でクリップするのですから、理屈では Outside ポートからは何も出力されないはずです。しかし、ここまでのワークスペースを実行してみると、実際には Outside ポートからもフィーチャー=どの市区町村にも属さない農用地区域の部分が出力されます。

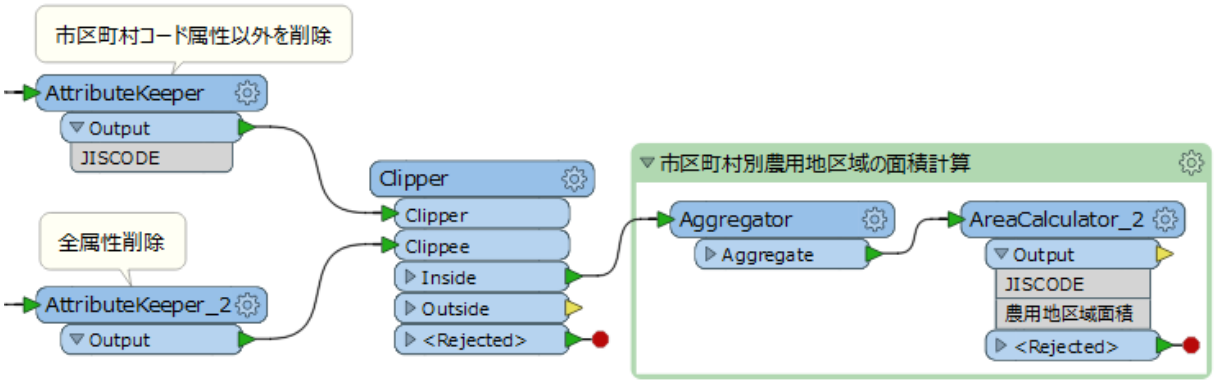
Outside ポートから出力されるフィーチャーには、次の 2 種類がありました。
ひとつは、隣接する市区町村の境界に沿った細長い形状の農用地区域です。これは、実習 6 でも述べたように、データ上は隣接する市区町村の間にごくわずかな隙間（誤差）が存在する場合があり、それによって生じたものと考えられます。
もうひとつは、市区町村境界のうち海岸線によって分割され、海上になってしまった農用地区域です。現実には海上の農用地区域はあり得ないので、ソースデータ作成に使用した元の図面や精度が、市区町村の区域と農用地区域の間で異なることによって生じたものと考えられます。

前者は、実習 6 のように CoordinateRounder を使って市区町村ポリゴンの頂点座標値の誤差を吸収することによって解消できる可能性がありますが、後者を解消することは困難です。

Clipper の Merge Attributes チェックボックスをオンにしたので、Inside ポートから出力されるフィーチャーには、そのジオメトリと重なる Clipper の属性が付加されます。つまり、Inside ポートから出力される農用地区域ポリゴンには、市区町村コード属性（JISCODE）が付加され、それによってどの市区町村に属するかを識別することができるようになります。

8-4. 市区町村別の農用地区域の面積計算

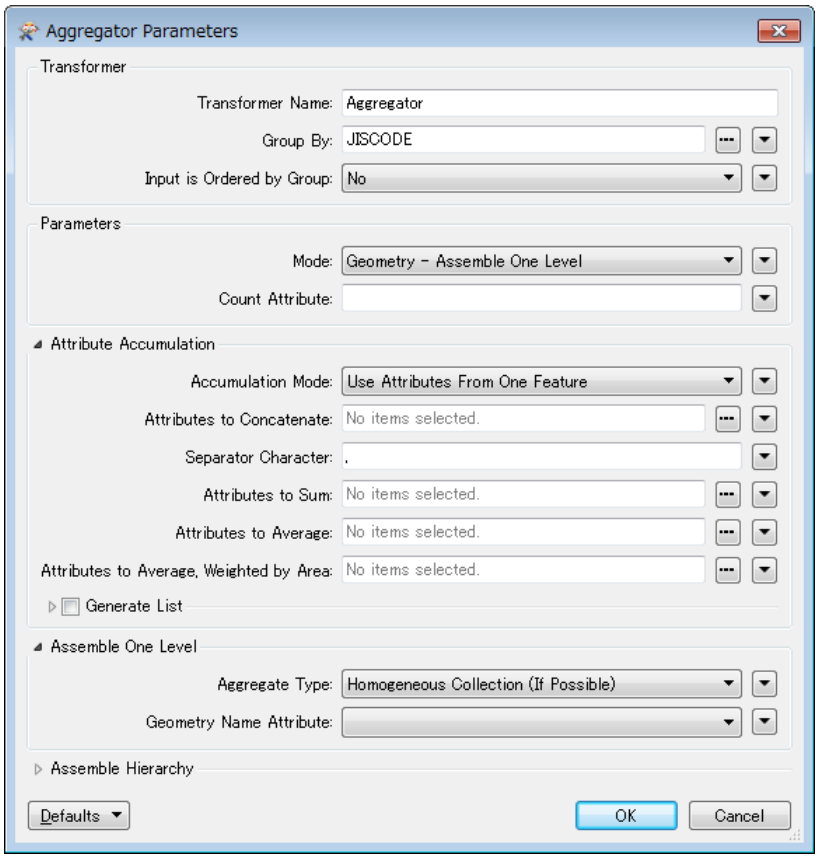
このステップの最終形は次のようになります。



(1) 市区町村ごとの農用地区域ポリゴン集約 Aggregator

ひとつの市区町村の区域に複数の農用地区域ポリゴンが含まれる（市区町村:農用地区域ポリゴン=1:N となる）場合があるので、Aggregator（Group By: JISCODE）によって、農用地区域ポリゴンを市区町村ごとに集約しておきます。

Aggregator によるフィーチャーの集約については、実習 6 を参照してください。



(2) 市区町村別の農用地区域面積計算 AreaCalculator

市区町村区域と同様に、AreaCalculator で面積を求めます。

これで、市区町村ごとに集約した農用地区域ポリゴン（マルチパート）に、農用地区域面積（km^2）が属性として付加されました。

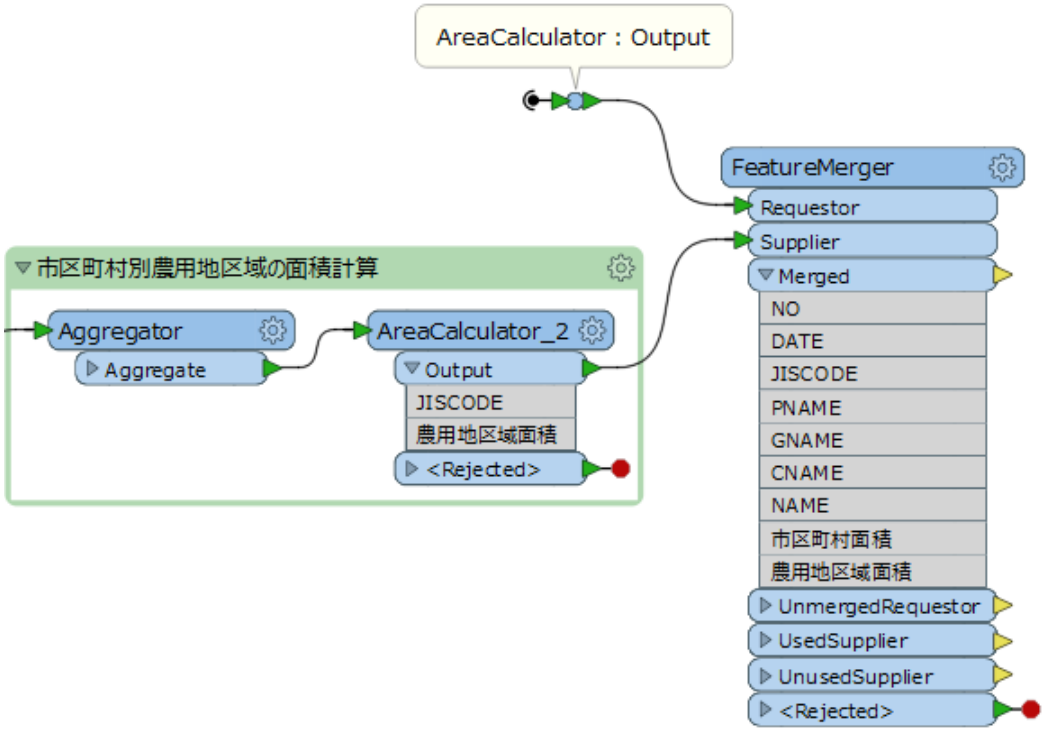
なお、区域内に農用地区域が全くない市区町村があった場合、その市区町村にかかる農用地区域ポリゴンは存在しません。

8-5. 市区町村ポリゴンへの農用地区域面積結合、率計算

最後のステップでは市区町村ポリゴンに農用地区域面積を結合したうえで、市区町村面積に占める農用地区域面積の率（%）を求めます。

(1) 属性の結合 FeatureMerger

FeatureMerger を追加し、Requestor ポートには 8-2 で追加した AreaCalculator（市区町村の面積計算）、Supplier ポートには 8-4 で追加した AreaCalculator_2（市区町村別の農用地区域面積計算）を接続し、JISCODE（市区町村コード）が一致することを条件として、Supplier の属性（農用地区域面積）を Requestor（市区町村ポリゴン）に結合します。



接続ライン（リンク）の表示/非表示と「トンネル」

キャンバス上のトランスフォーマーなどのオブジェクトの右クリックメニューで Hide Connections（接続ラインを非表示にする） を選択すると、そのオブジェクトに接続しているすべての接続ラインが非表示になり、その状態で、右クリックメニュー > Show Connections（接続ラインを表示する）を選択すると、再び表示されます。

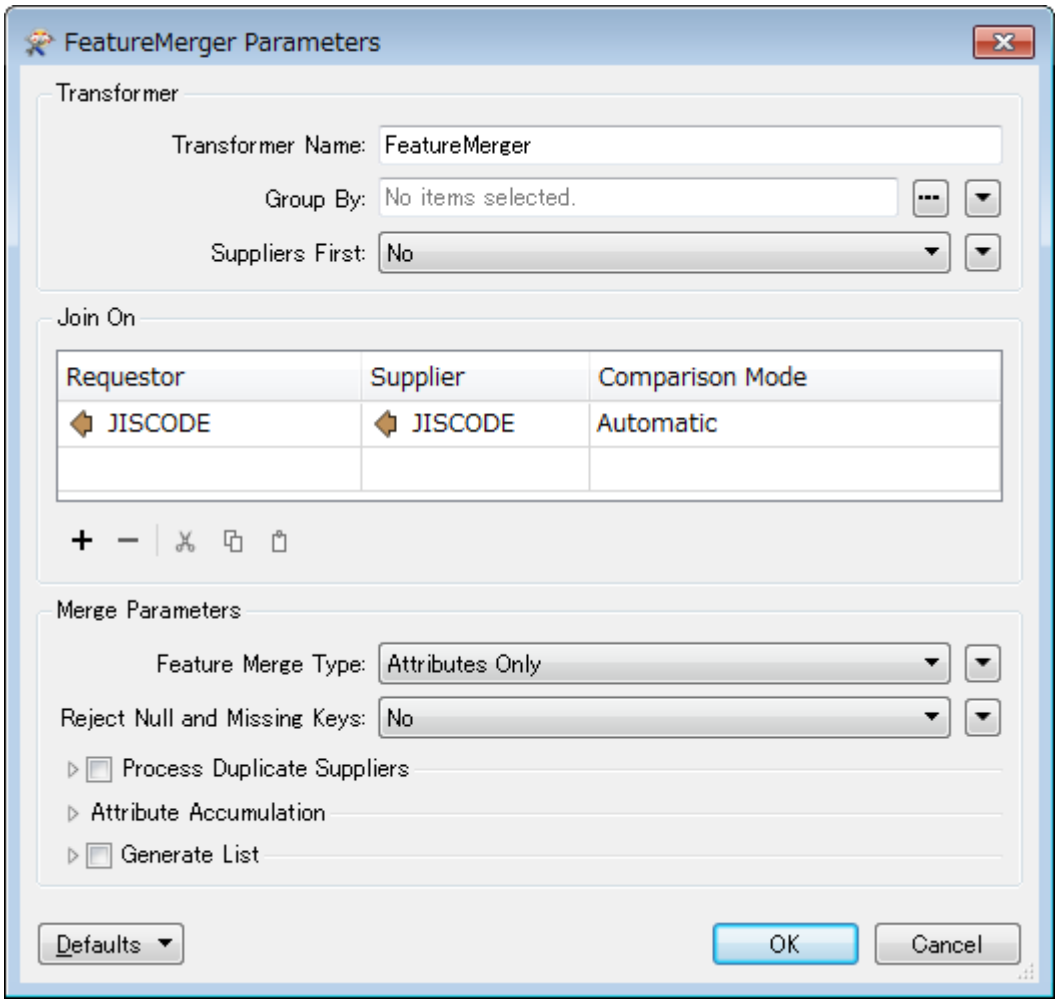
また、接続ラインの右クリックメニュー > Create Tunnel（トンネルの作成）を選択すると、その接続ラインの両端に Junction（ジャンクション）が挿入され、それらの間の接続ラインが非表示になります。このように、両端に Junction があり、その間の接続ラインが非表示になっているリンクを「トンネル」と呼んでいます。

上の図では、8-1 で追加した AreaCalculator（市区町村ポリゴンの面積計算）の Output ポートと FeatureMerger の Requestor ポートを接続した後、その接続ラインを「トンネル」にした状態を示しています。

接続ラインの表示/非表示の設定や「トンネル」は、ワークスペースの実行結果には影響を及ぼしませんが、大規模なワークスペースでは、これを活用することによって作成や保守がし易くなります。

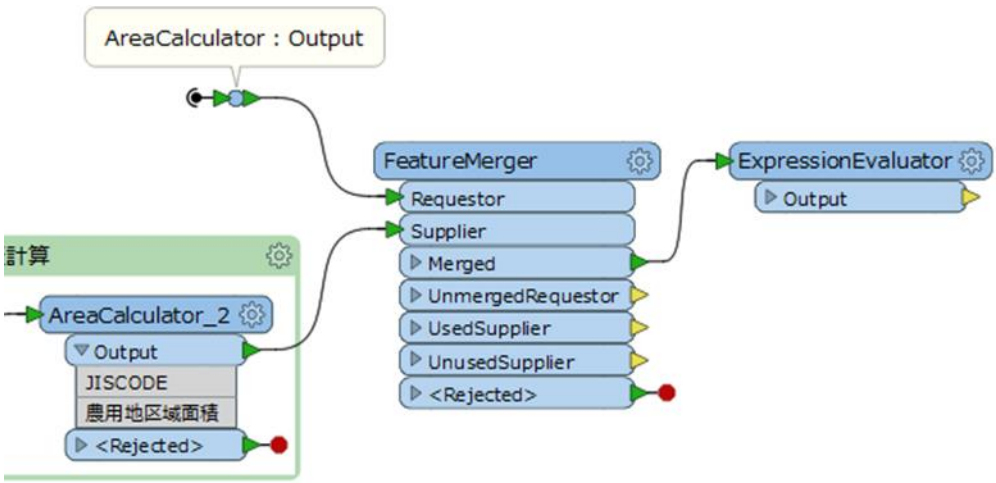
FeatureMerger のパラメーター設定画面を開き、次ページの図のように、Join On パラメーターの Requestor 側、Supplier 側双方に JISCODE（市区町村コード）を指定することで、市区町村ごとに農用地区域面積属性が市区町村ポリゴンに結合できます。

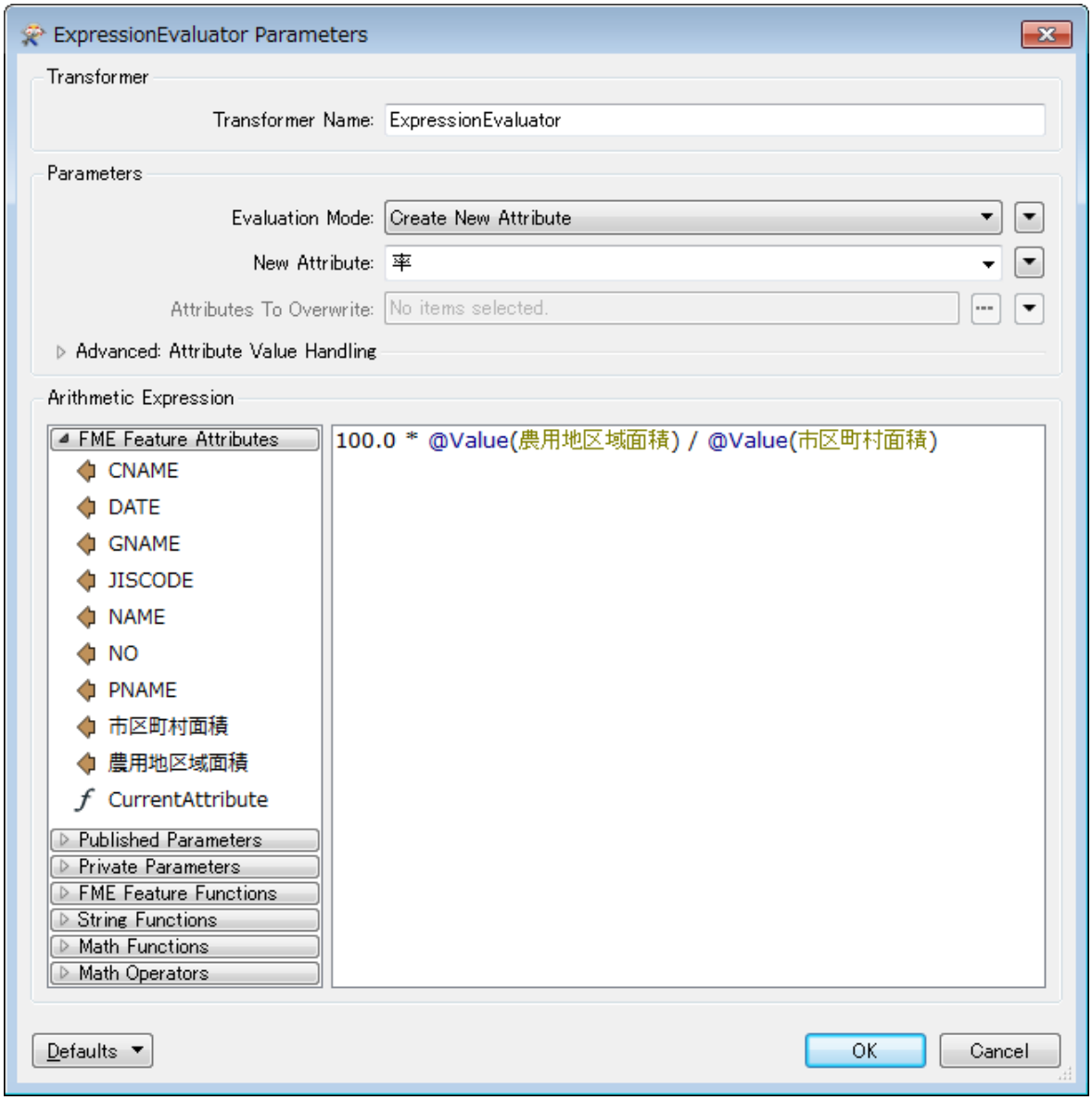
実行時には、Merged ポートから農用地区域面積が付加された市区町村ポリゴンフィーチャー、UnmergedRequetor ポートから農用地区域が全くなかった市区町村ポリゴンフィーチャーが出力されます。



(2) 市区町村面積に占める農用地区域の率の計算 ExpressionEvaluator

FeatureMerger の Merged ポートから出力されるフィーチャーは、市区町村面積と農用地区域面積を属性として持っているので、ExpressionEvaluator によってそれらの属性値に基づいて市区町村面積に占める農用地区域面積の率（%）を求めて新たな属性として付加することができます。





ExpressionEvaluator ではなく、AttributeCreator または AttributeManager を使うこともできます。

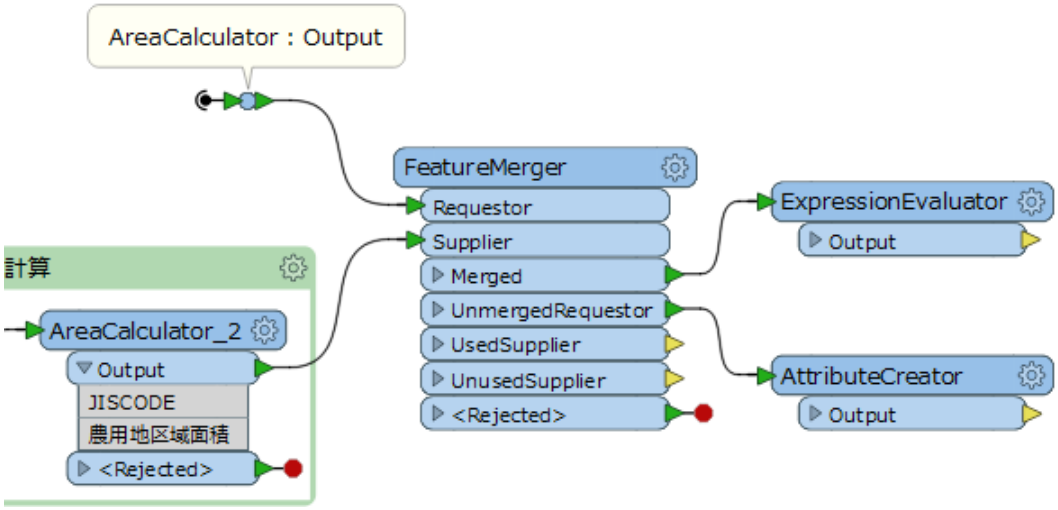
トランスフォーマーの入力フィールドで数式を設定する方法については、ヘルプや次のドキュメントを参照してください。

数式エディタ (Arithmetic Editor)
http://www.pragmatica.jp/fme/references/BasicArithmeticEditor.html

(3) 農用地区域がない市区町村の属性作成 AttributeCreator

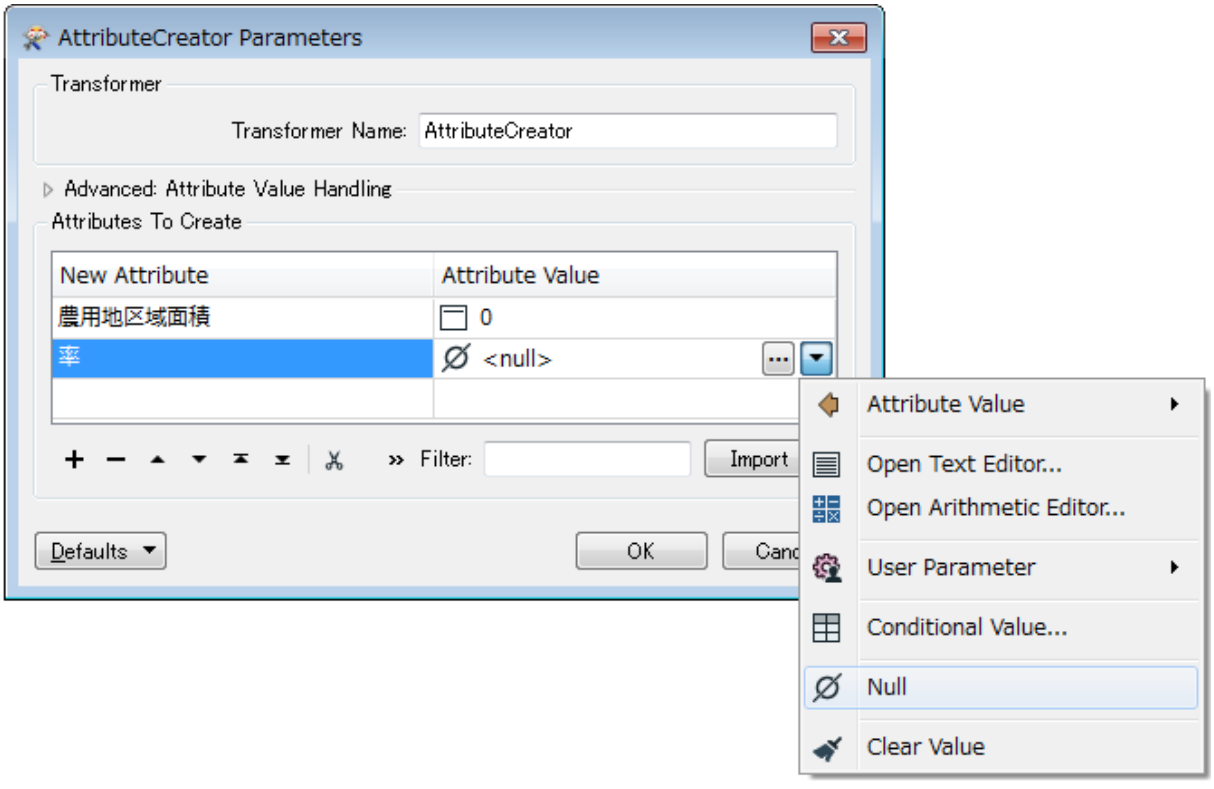
市区町村ポリゴンのうち農用地区域が全くないものについて JISCODE が一致する Supplier フィーチャー（市区町村単位で集約した農用地区域フィーチャー）が存在しないため、UnmergedRequestor ポートから出力されます。

それらには「農用地区域面積」属性が付加されず、「率」を求めることもできないので、AttributeCreator を使って「農用地区域面積」属性（値 = 0）、「率」属性（値 = <null>（ナル）：有効な値がないことを示す特別な値）を付加することにします。

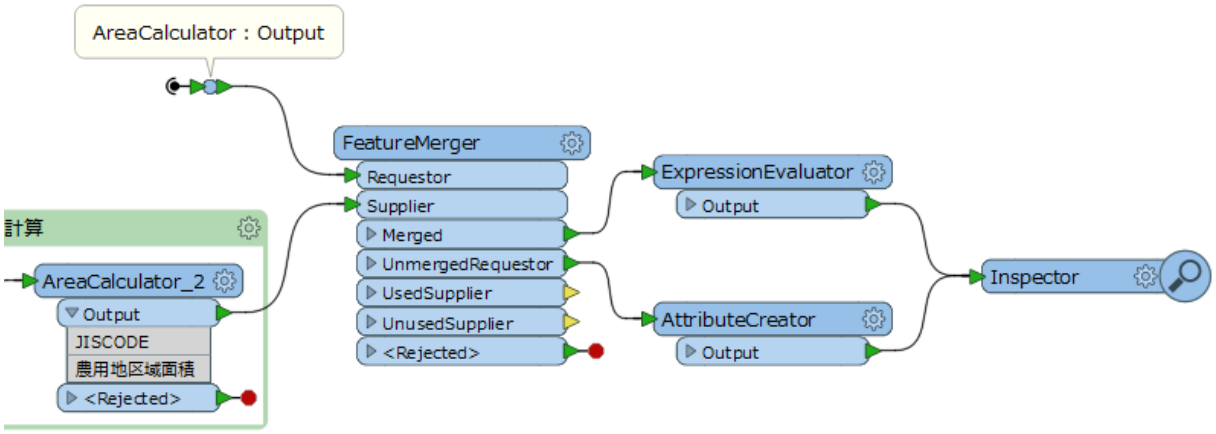


AttributeCreator のパラメーター設定画面の Attributes To Create（作成する属性）テーブルでは、作成する（あるいは値を変更する）属性の名前（New Attribute）とその値（Attribute Value）の組み合わせを設定します。

<null>（ナル）は特別な値であり、キー入力はできません。入力フィールドを選択したときに表示されるメニューボタンをクリックし、Null を選択することによって設定します。

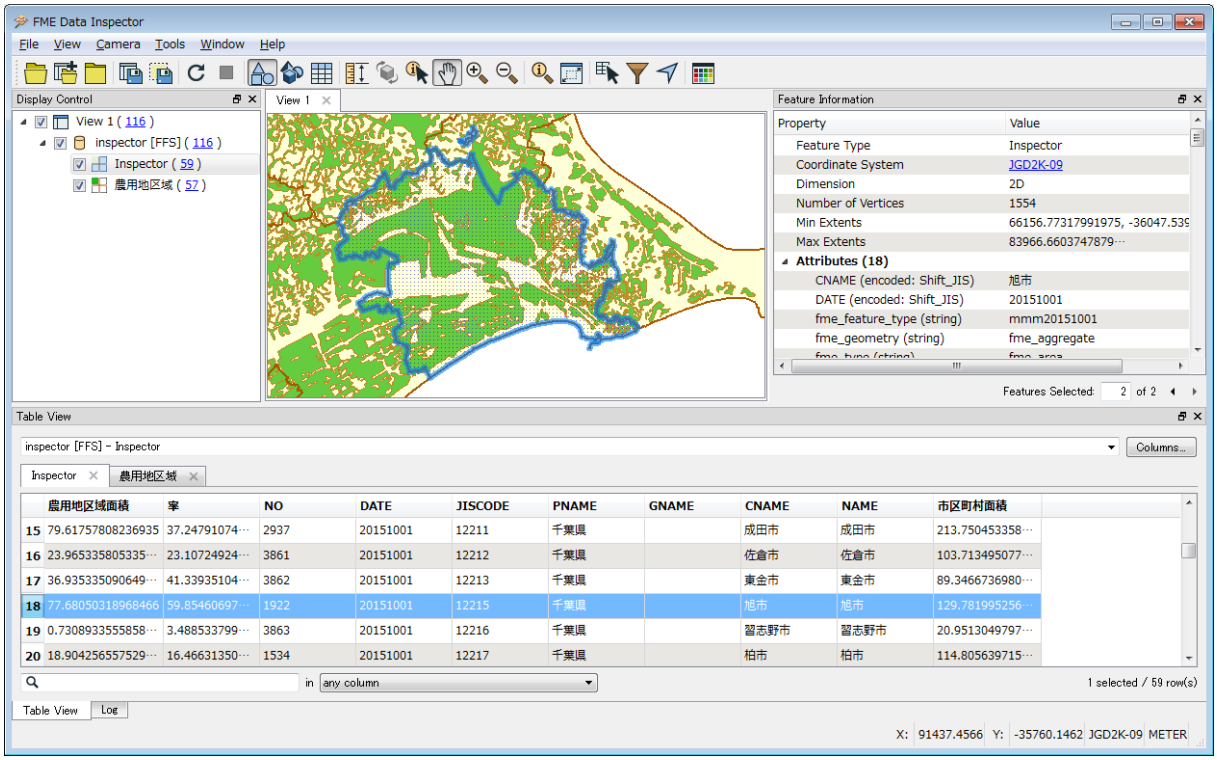


Inspector を追加し、ExpressionEvaluator、AttributeCreator に接続すれば完成です。



次の図は、クリッピング後の農用地区域ポリゴンの形状も確認するために、もうひとつの Inspector を AreaCalculator_2（農用地区域の面積計算）の Output ポートに接続して実行した結果です。

フィーチャーを選択したときに Feature Information ウィンドウで Coordinate System として JGD2K-09（変換後の座標系）が表示されることも確認してください。



実習 8 はここまでです。主に次の事項を学びました。

・座標系変換 CsmapReprojector, Coordinate System Gallery（座標系ギャラリー）
・面積計算 AreaCalculator
・クリッピング Clipper
・属性値の計算 ExpressionEvaluator

座標系変換に関する補足 1: フィーチャーへの座標系設定

演習で使用したソースデータ（Shapefile 形式の市区町村ポリゴン、農用地区域ポリゴン）には、データの内容として座標系（JGD2000 緯度経度）が定義されていたので、FME はそれを自動的に読み取り、フィーチャーに与えることができました。

しかし、ソースデータに座標系の定義が含まれていない場合や、データフォーマット自体が座標系の記述をサポートしない場合もあります。

そのようなソースデータの座標系の変換を行いたいときは、次のどちらかによって、入力フィーチャーに元の座標系を明示的に設定しておく必要があります。

・リーダーの Coordinate System（座標系）パラメーターによって座標系を設定する。
・CoordinateSystemSetter トランスフォーマーによって座標系を設定する。

座標系変換に関する補足 2: 座標系の復元

市区町村ポリゴンデータの元の座標系は JGD2000 緯度経度でしたが、データフローの途中で面積を求めるために座標系を変換したため、最終的に出力される市区町村ポリゴンの座標系は変換後の座標系（平面直角座標系第 IX 系）となります。

実務では、データフローの途中でジオメトリに基づいて面積等の計測を行うために座標系を変換する必要はあるが、出力先のデータの座標系はソースデータと同じでなくてはならないというケースもあります。

出力する前にもうひとつ CsmapReprojector を挿入する、あるいは、ライターの Coordinate System パラメーターを設定することによって座標系を再変換することはできますが、座標系変換には必ず何らかの補間処理が伴うため、そのように座標系の変換を繰り返したときに、ソースデータと厳密に同じジオメトリが復元できるとは限りません。

データフローの途中で座標系を変換した後、正確にソースデータと同じジオメトリを復元したいという場合、次のようにいくつかのトランスフォーマーを使用することでそれが可能になります。

・座標系変換の前に GeometryExtractor によってジオメトリを属性として格納しておき、
・座標系変換、計測等の後で GeometryReplacer によってその属性からジオメトリを復元し、
・CoodinateSystemSetter によってソースデータと同じ座標系を設定する。

GeometryExtractor の Geometry Encoding パラメーターにはジオメトリを属性値に変換する方法としていくつかの選択肢がありますが、ジオメトリを後で復元するのが目的である場合は"FME Binary"とするのが最も効率良く、かつ、正確な復元ができます。

CoordinateSystemSetter は、フィーチャーが特殊な属性として持っている座標系識別子を設定、変更するものです（ジオメトリには変更を加えません）。

GeometryReplacer でジオメトリを復元しても、フィーチャーが持っている座標系識別子は変わらない（実習で作成したワークスペースの場合は、CsmapReprojector で変換後の平面直角座標 IX 系のままである）ので、CoordinateSystemSetter によって明示的に正しい座標系識別子を設定し直す必要があります。

あるいは、ソースデータ読込直後のデータフローを 2 つに分け、一方では何も処理は行わず、もう一方で座標系変換、計測等を行った後、FeatureMerger 等を使って計測等によって作成した属性を何も手を加えていないフィーチャーに結合するという方法も可能です。