FME® Desktop Basic Training Course

FME Desktop 基础培训教程

Table of Contents

简介 1.1
关于本文档 1.2
课程大纲 1.2.1
课程资源 1.2.2
数据转换基础知识 1.3
什么是FME？ 1.3.1
练习：探索FME 1.3.1.1
FME Desktop组件 1.3.1.2
FME Workbench简介 1.3.2
Workbench中的窗口控件 1.3.2.1
创建转换 1.3.2.2
新工作空间 1.3.2.3
运行工作空间 1.3.2.4
练习：基本工作空间创建 1.3.2.5
读模块参数 1.3.3
写模块参数 1.3.4
数据检查 1.3.5
FME Data Inspector简介 1.3.5.1
在FME Data Inspector中查看数据 1.3.5.2
查询FMEData Inspector中的数据 1.3.5.3
练习：基本数据检查 1.3.5.4
Data Inspector显示控件 1.3.5.5
Data Inspector中的背景地图 1.3.5.6
练习：FME Data Inspector 1.3.6
模块复习 1.3.7
Q + A答案 1.3.7.1
练习：旅游局项目 1.3.7.2
数据转换 1.4
什么是数据转换 1.4.1
结构转换 1.4.1.1
模式编辑 1.4.1.2
模式映射 1.4.1.3
练习：地面维护项目 - 模式编辑 1.4.1.4
用转换器进行转换 1.4.2
练习：地面维护项目 - 结构转型 1.4.2.1
内容转换 1.4.3
转换器系列 1.4.3.1
练习：地面维护项目 - 计算统计数据 1.4.3.2
要素计数显示 1.4.3.3

转换器并行 1.4.3.4
练习：地面维护项目 - 标签功能 1.4.3.5
分组处理 1.4.4
练习：地面维护项目 - 街区平均值 1.4.4.1
Workbench的数据检查 1.4.5
坐标系转换 1.4.6
练习：地面维护项目 - 数据重投影 1.4.6.1
模块复习 1.4.7
Q + A答案 1.4.7.1
练习：投票分析项目 1.4.7.2
工作空间设计 1.5
工作空间原型 1.5.1
增量发展 1.5.1.1
被拒绝的端口 1.5.1.2
练习：住宅垃圾收集区 1.5.1.3
读取和写入工作流程 1.5.2
主要组件 1.5.2.1
多个读模块和写模块 1.5.2.2
FeatureReader和FeatureWriter 1.5.2.3
使用集成转换器进行读取/写入 1.5.2.4
练习：住宅垃圾收集区 1.5.2.5
工作空间测试技术 1.5.3
综合检查 1.5.3.1
部分运行 1.5.3.2
练习：住宅垃圾收集区 1.5.3.3
模块复习 1.5.4
Q + A答案 1.5.5
实用转换器使用 1.6
找到转换器 1.6.1
转换器库 1.6.1.1
转换器搜索 1.6.1.2
最有价值的转换器 1.6.1.3
练习：转换器选择练习 1.6.1.4
管理属性 1.6.2
创建和设置属性 1.6.2.1
构造属性 1.6.2.2
构建转换器参数 1.6.2.3
重命名和复制属性 1.6.2.4
批量属性重命名 1.6.2.5
删除属性 1.6.2.6
练习：地址开放数据项目 1.6.2.7
条件过滤 1.6.3

Tester和TestFilter 1.6.3.1
其他主要过滤转换器 1.6.3.2
练习：噪音控制法律项目 1.6.3.3
数据连接 1.6.4
基于键的连接转换器 1.6.4.1
基于空间的连接转换器 1.6.4.2
练习：犯罪映射数据请求 1.6.4.3
模块复习 1.6.5
练习1答案 1.6.5.1
Q + A答案 1.6.5.2
最佳实践 1.7
样式 1.7.1
注释工作空间 1.7.1.1
书签 1.7.1.2
书签用于设计 1.7.1.2.1
书签用于访问 1.7.1.2.2
书签用于编辑 1.7.1.2.3
书签用于性能 1.7.1.2.4
对象布局 1.7.1.3
连接样式 1.7.1.4
练习：FME风格指南 1.7.1.5
方法伦 1.7.2
维护方法 1.7.2.1
性能方法 1.7.2.2
练习：设计模式 1.7.2.3
调试 1.7.3
记录 1.7.3.1
检查输出 1.7.3.2
要素计数 1.7.3.3
检查关键阶段 1.7.3.4
练习：调试工作空间 1.7.3.5
要素调试 1.7.3.6
模块复习 1.7.4
Q + A答案 1.7.4.1
练习：最短路线黑客马拉松 1.7.4.2
课程总结 1.8
产品信息和资源 1.8.1
社区信息和资源 1.8.2
反馈和证书 1.8.3
谢谢 1.8.4

简介

这是Safe Software FME Desktop应用程序入门培训课程的手册。

培训将介绍基本概念和术语，帮助学生成为FME的有效用户，并指导您使用资源来帮助您将产品应用于您自己的需求。

课程结构

全部课程由五个主要部分组成。这些部分是：

- 数据转换基础知识
- 数据转换
- 工作空间设计
- 实用转换器使用
- 最好的练习

当前状态

本手册的当前状态为：完整：本手册可用于培训。

本手册适用于**FME2018.0**

每章的状态是：

- 第0章：完整内容。没有练习
- 第1章：完整内容和练习
- 第2章：完整内容和练习
- 第3章：完整内容和练习
- 第4章：完整内容和练习
- 第5章：完整内容和练习
- 第6章：完整内容。没有练习
- 幻灯片：完整

- FME数据：完整
- 课程大纲：完整

注意： 即使是已完整的内容，*Safe Software Inc.*也不对本文档中的任何错误或其后果承担任何责任，并保留对本文档进行改进和更改的权利，恕不另行通知。有关详细信息，请参阅完整许可协议。

关于本文档

本手册是FME Desktop的入门级培训课程。

留意Interopolis市的居民，他们将不时出现，为您提供建议并分发与FME相关的智慧。事实上，现在就有一些人：

E.Dict先生（**FME**法律部律师）说**.....**
代表Interopolis市，欢迎您参加此培训课程。以下是有关此培训文档的标准法律信息以及课程中使用的数据集。 请务必阅读，特别是如果您正在考虑重新使用或修改此内容。

许可和保修

特此授予使用，修改和分发FME教程以及相关数据和文档（统称为“教程”）的权限，但受以下限制：

1. 不得歪曲教程的起源和任何相关的FME®软件。
2. 原始或修改形式的再分发必须包括安全软件的版权声明和任何适用的数据源通知。
3. 您不能建议Safe Software公司认可或批准任何修改版本的教程。
4. 原始或修改形式的再分发必须包括类似于以下内容的免责声明：（a）声明教程是“按原样”提供的;（b）不承担任何保证;（c）放弃任何责任索赔。

Safe Software Inc.不做任何明示或暗示的保证，包括但不限于适销性，非侵权性或针对特定用途的适用于本教程的任何默示保证，并且使这样的教程仅在“原样”的基础上可用。在任何情况下，Safe Software Inc.均不对任何人因使用，修改或分发本教程而导致的直接，间接，特殊，抵押，偶然或后果性损害承担责任。

本手册介绍了软件在发布时的功能和用途。此处描述的软件和描述本身如有更改，恕不另行通知。

数据源

温哥华市

除非另有说明，否则此处使用的数据源自不列颠哥伦比亚省温哥华市提供的开放数据。它包含根据开放政府许可证 - 温哥华许可的信息。

其他

转发分拣区域：加拿大统计局，2011年人口普查数字边界文件，2013年。经加拿大统计局许可，以“原样”转载和分发。©此数据包含经加拿大邮政公司许可复制的信息。

数字高程模型：GeoBase®

消防站数据：一些属性数据改编自维基百科的内容©2013 ，在Creative Commons Attribution-ShareAlike许可下使用

Stanley公园GPS追踪：获得VancouverTrails.com的许可。

OpenStreetMap数据集：贡献者©OpenStreetMap。

包含根据开放政府许可证 - 不列颠哥伦比亚省许可的信息

版权

©2005-2018 Safe Software Inc.保留所有权利。

修订

我们已尽最大努力确保本文件的准确性。Safe Software Inc.对可能发生的任何错误和遗漏表示遗憾，并希望了解发现的任何错误。Safe Software Inc.将尽可能纠正后续版本中的任何此类错误和遗漏。请联系我们：

Safe Software Inc.
电话： 604-501-9985
电子邮件： train@safe.com
网址： safe.com
GitHub： github.com/antu/FMETraining

Safe Software Inc.对本文档中的任何错误或其后果不承担任何责任，并保留对本文档进行改进和更改的权利，恕不另行通知。

商标

FME®是Safe Software Inc.的注册商标。所有品牌或产品名称均为其各自公司或组织的商标或注册商标。

文件信息

文档名称：FME Desktop Basic培训手册2018.0

新内容是什么？

可以在GitHub上找到本手册及其附带数据集的更改列表。该文件包含与上一年的材料相比的一般修订清单。它旨在帮助培训师快速掌握新内容，并帮助学生确定当前版本中哪些FME功能是新功能。

课程大纲

该培训课程为FME的基本理解提供了框架。我们发现用户是来掌握一项功能的，但却带着许多新的FME用途回家。

培训将介绍基本概念和术语，帮助学生成为FME的有效用户，并指导您使用资源来帮助您将产品应用于您自己的需求。

课程结构

全部课程由五个部分组成。这些部分是：

- 数据转换基础知识
- 数据转换
- 工作空间设计
- 实用转换器使用
- 最佳练习

课程长度

教师可以选择在允许的时间内尽可能多地覆盖这些部分。他们还可能以不同的顺序覆盖课程内容，并会跳过或添加新内容以根据您的需要自定义课程。

因此，课程的长度和内容可能会有所不同，特别是在网上授课时。

Safe Software提供各种长度的培训，但通常为两天或五天。这两门课程都涵盖相同的内容。为期两天的课程包括更少但更长的每日课程，并且更加紧张。这个为期五天的课程包含更多但更短的每日课程，并允许更多时间从初学者的角度来涵盖内容。

关于手册

FME Desktop培训手册不仅构成了FME桌面培训的基础 - 面对面或在线 - 而且也是您可能与FME进行的未来工作的有用参考资料。它针对FME的每个主要版本进行了更新。

这些材料中的所有屏幕截图都是在Windows Server 2016上使用FME拍摄的。使用的字体（特别是在日志窗口的屏幕截图中）可能会调整大小或以其他方式更改，以提高可读性。

课程资源

本课程将使用许多样本数据集和工作空间。

在你的训练计算机上

本培训课程中使用的数据基于加拿大温哥华市的公开数据。

大多数练习要求您扮演虚构城市Interopolis的城市规划师的角色，并使用这些数据解决特定问题。

无论是本地计算机还是托管在云中的虚拟计算机，您都可以在以下位置找到手册中的示例和练习的资源：

位置	资源
C:\FMEData2018\Data	Interopolis市使用的数据集
C:\FMEData2018\Resources\DesktopBasic	培训中使用的其他资源
C:\FMEData2018\Workspaces\DesktopBasic	学生练习中使用的工作空间
C:\FMEData2018\Output	写入练习输出的位置
<documents> \ FME \ Workspaces	保存FME工作空间的默认位置

您还应该预先安装FME，以及本手册的数字副本。

如果您的设置中缺少任何项目，请提醒您的教师。

您可以在 **Safe Software**网站上找到适用于Windows，Mac和Linux的最新版本的FME Desktop和FME Server以及最新的Beta版本。

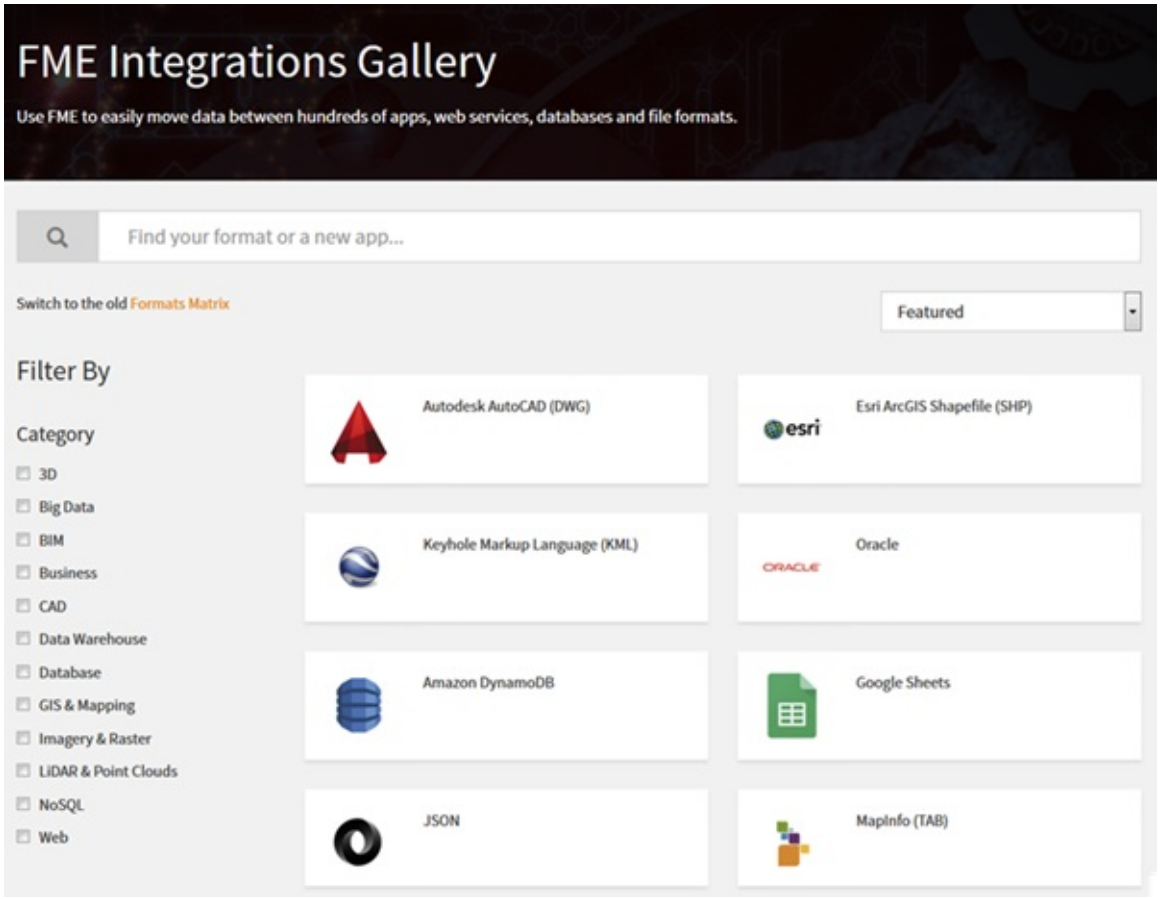
课程礼仪

对于在线课程，请考虑其他学生并在课程开始前测试您的虚拟机连接。教师在课程期间无法帮助调试连接问题！

对于现场课程，请在使用手机或查收电子邮件时将噪音降至最低，以满足其他学生的需求。

数据转换基础知识

从本质上讲，FME是一种数据转换工具，这通常是用户希望了解的第一个方面。

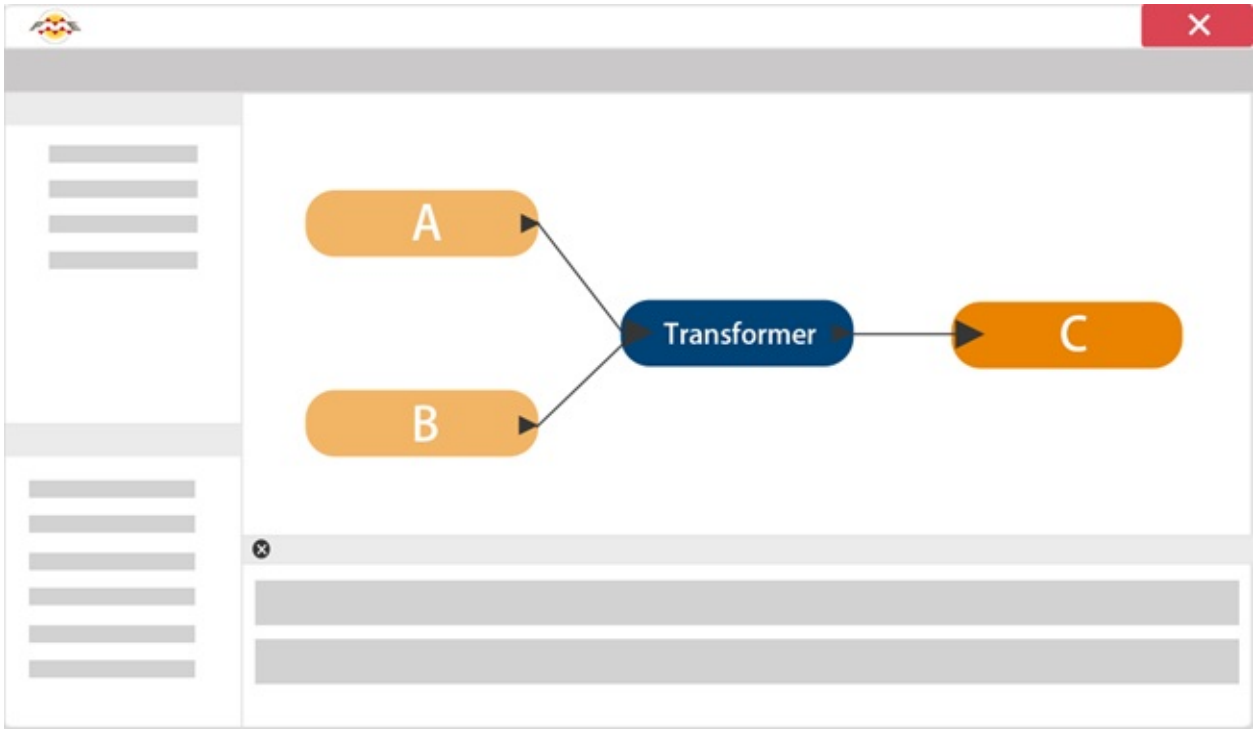


什么是**FME**？

FME（要素操作引擎）是一种数据转换和变换工具，用于解决数据互操作性问题，无需编码。

提取，转换和加载

FME有时被归类为**ETL**应用程序。ETL代表Extract，Transform和Load。它是一个数据仓库工具，可以从多个源（此处为A和B）提取数据，对其进行转换以满足用户的需求并将其加载到目标（C）中：

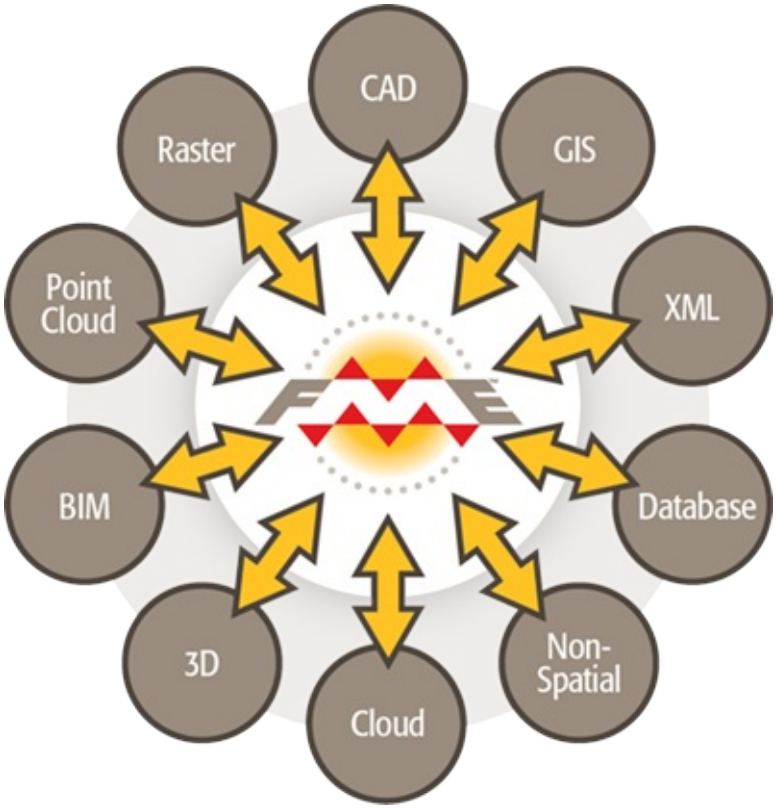


虽然大多数ETL工具仅处理表格数据，但FME还具有处理空间数据所需的处理能力，因此称为空间**ETL**。

FME如何工作

FME的核心是一个支持一系列空间和表格数据类型和格式的引擎; GIS，CAD，BIM，Point Cloud，XML，Raster，数据库等等。

通过处理所有可能的属性和几何类型的丰富数据模型，可以支持如此多的数据类型。



最重要的是，数据转换过程对用户是无缝的; FME根据需要自动在数据类型之间进行转换，并自动将一种属性或几何类型替换为目标格式不支持的另一种属性或几何类型。

Vector小姐说.....

学生们注意了！我是Vector，FME教师。我来这里测试你关于FME。我希望你不要弄错这些问题！ Q）ETL是.....的首字母缩写？ 1.地外生命形态 2.提取，转换，加载 3.快速收费车道 4.吃，变换，爱 Q）FME可以在如此多的格式之间无缝转换，因为它有..... 1.一个有知觉的数据字典 2.一个复古增压器 3.丰富的数据模型 4.独角兽头发的核心

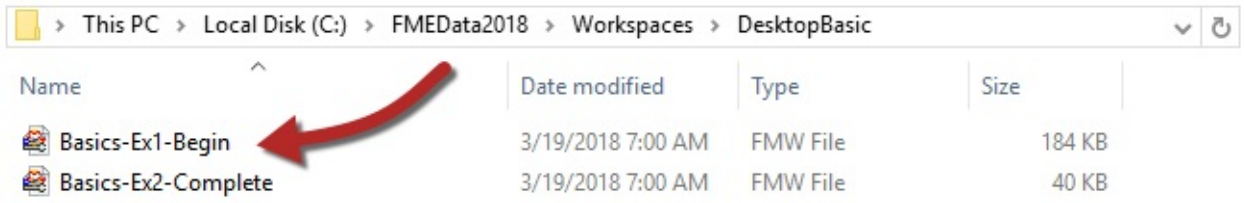
练习：探索FME

练习**1**	打开和运行工作空间
数据	库（Esri Geodatabase） 道路（AutoCAD DWG）
总体目标	打开并运行FME工作空间以探索它可以对数据执行的操作
演示	打开并运行工作空间
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Basics-Ex1-Begin.fmw
结束工作空间	N / A

不要试图解释FME是什么和做什么，而是让我们自己尝试一下！

1）查找工作空间文件
在FME中定义转换和变换时，可以将它们保存在.fmw文件中。

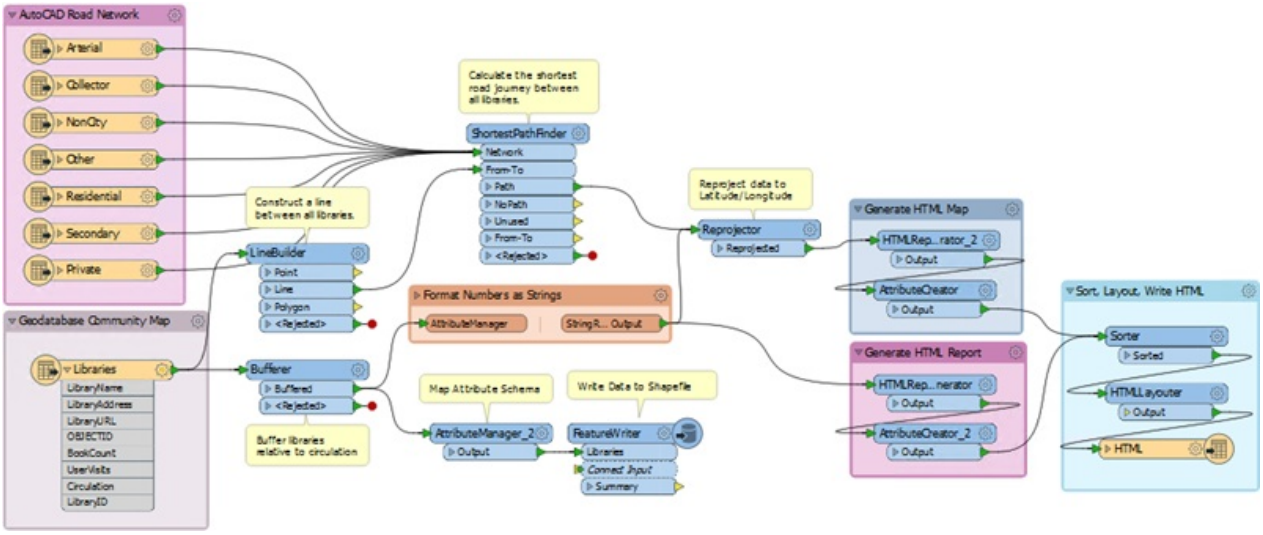
使用文件资源管理器，浏览到上面列出的文件：



双击该文件。它将打开一个名为FME Workbench的应用程序。

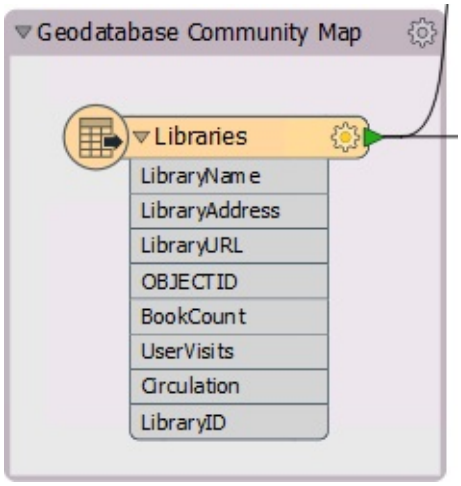
2）探索**FME**工作空间
在FME Workbench中，通过单击“不再显示我”来关闭可能打开的任何“开始”对话框。

该应用程序的主要部分将如下所示：



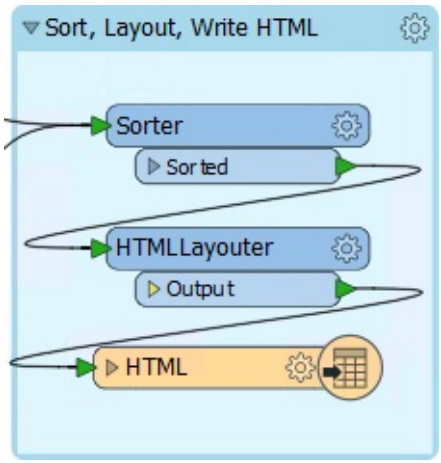
这部分我们称之为画布。它是以图形方式定义数据的转换和变换的地方。虽然它可能看起来很复杂，但使用FME创建此类型的工作流程并不需要太多练习。

检查画布的左侧：



这个区域是我们读取数据的地方，在本例中是Esri地理数据库中的一个库表。它是ETL的“E”（Extract）部分。

现在看右边：



这个区域是我们编写数据的地方，在本例中，是HTML格式的库报告。它是ETL的“L”（加载）部分。

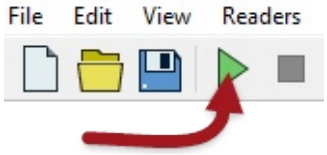
读模块和写模块之间是转换数据的对象。它们代表ETL的“T”（变换）部分。

标签和其他注释向我们展示了工作空间的功能。它：

- 读取道路（AutoCAD DWG）和库（Esri Geodatabase）
- 计算所有图书馆的最短道路路线
- 创建直径相对于图书馆图书流通的圆圈
- 创建HTML报告和库的HTML映射
- 将数据写入HTML以及Esri Shapefile

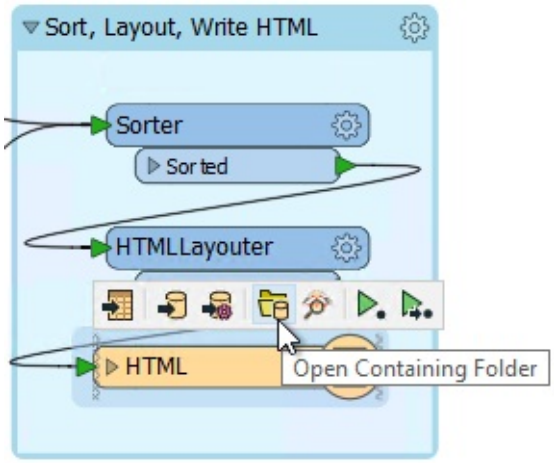
警察局长**Webb-Mapp**说.....
我是警察局长，负责追查针对FME的犯罪行为。 所以，让我们确保您获得正确的术语。 应用程序本身称为FME“Workbench”，但在画布窗口中定义的过程称为“工作空间”。条款非常相似，很容易混淆，但请不要。否则，我将不得不派我的语法小队逮捕你！ 虽然虐待FME术语是一种轻微的冒犯，但被抓住的耻辱是持久的！

3）运行**FME**工作空间
让我们运行此工作空间。要执行此操作，请单击Workbench工具栏上的绿色运行按钮：

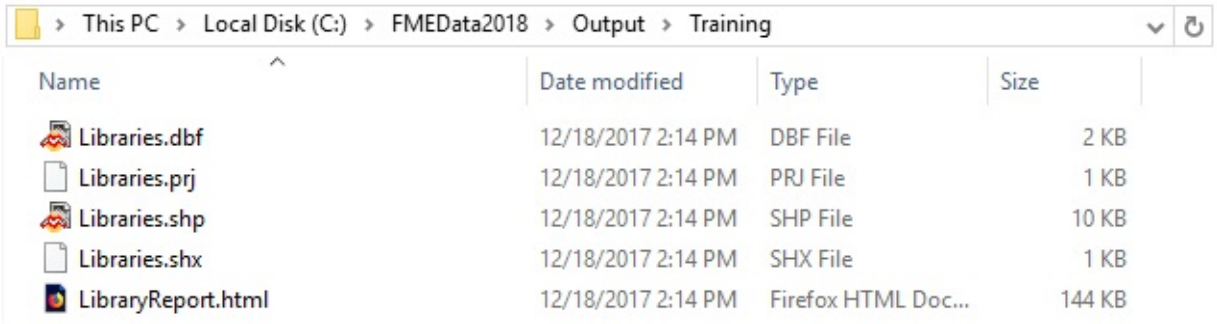


工作空间现在将运行。当它这样做时，您将在日志窗口中看到消息传递。您还可能会在画布连接上看到数字，并在每个对象上看到绿色带注释的图标。我们将在以后了解这些内容！

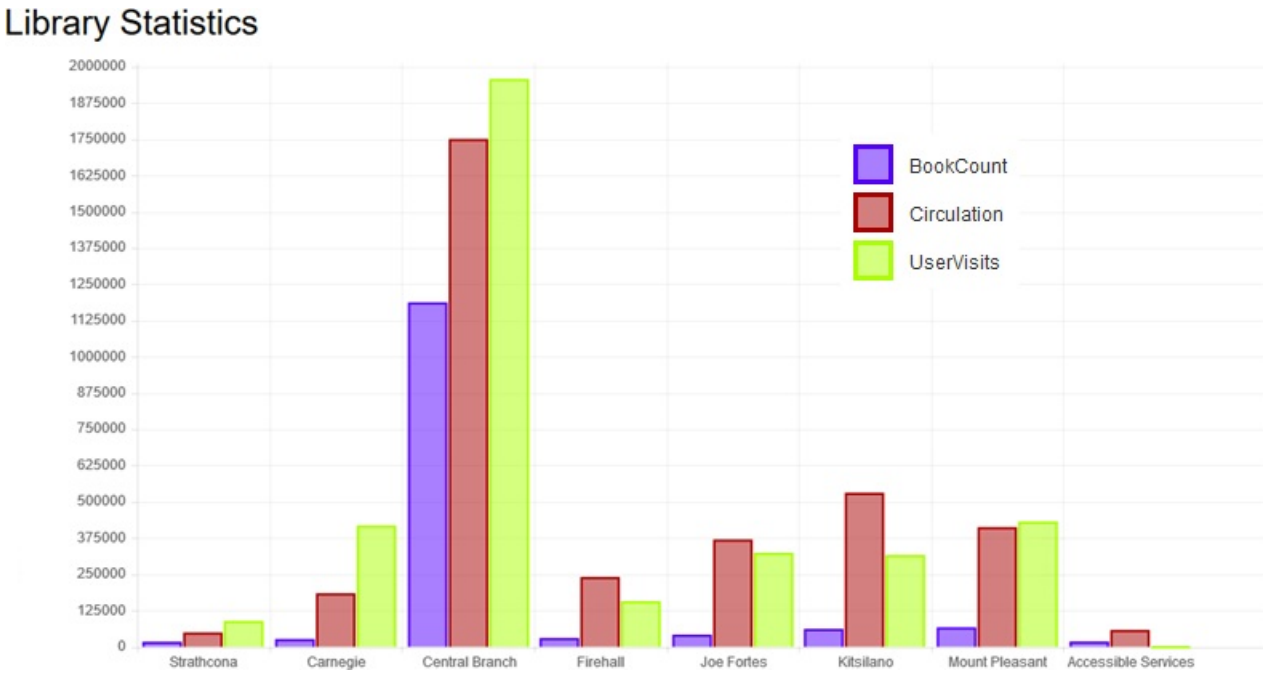
4）找到并检查输出
转换完成后，单击画布上的HTML写模块对象。选择打开包含文件夹的选项：



在打开的资源管理器对话框中，您将找到HTML输出和Shapefile数据集：



打开Firefox或Chrome等Web浏览器。打开由FME创建的输出文件（通常是Ctrl + O或文件>打开是最简单的方法）。您将看到一个库表，一个库统计图表，以及一个显示库所在位置的交互式地图。所有这些都是由FME从传入的地理数据库点和属性生成的：



Vector小姐说.....

这个小型演示展示了FME的强大功能。此工作空间从多个空间数据集中读取数据，并以空间和“表”格式将其写入数据集。在它之间进行了一系列的转换和空间分析，缓冲和重新投影数据，并创造附加价值和信息。

恭喜

通过完成本练习，您已学会如何：

- 打开FME工作空间
- 运行FME工作空间
- 找到FME工作空间的输出

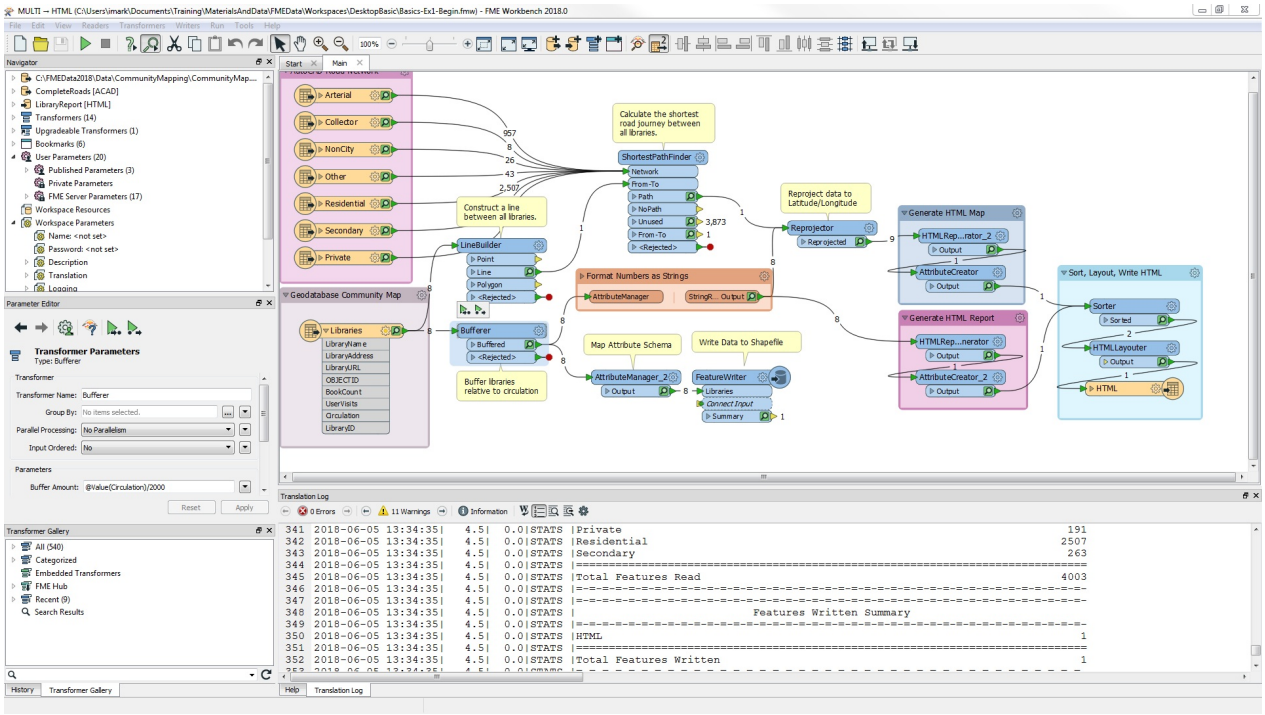
FME Desktop组件

本课程是关于FME Desktop的。FME Desktop用于桌面级别的数据转换和变换（与FME Server相反，后者是基于Web的企业级产品）。

FME Desktop包含许多不同的工具和应用程序。两个关键应用程序是**FME Workbench**和**FME Data Inspector**。

FME Workbench

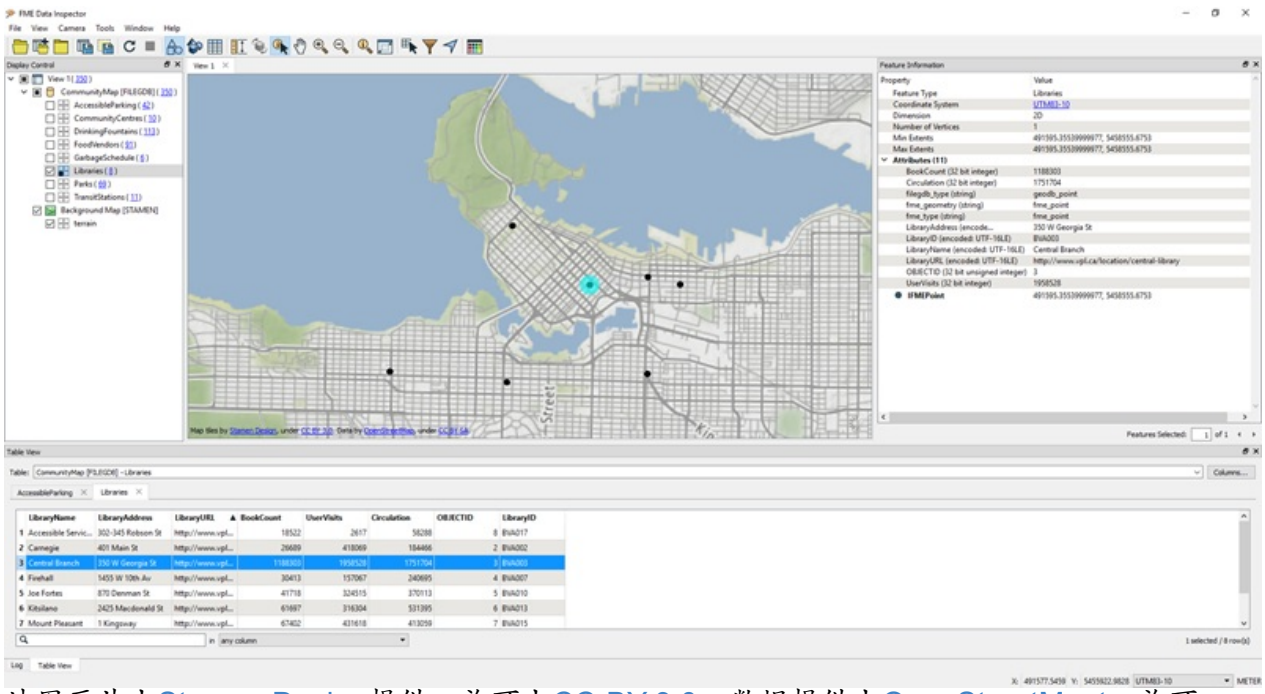
FME Workbench是定义数据转换和数据变换的主要工具。它具有直观的点击式图形界面，可以将转换图形化地描述为数据流。



Workbench不是一个独立的工具。它完全集成，可与其他FME桌面应用程序（如FME Data Inspector）进行交互。

FME Data Inspector

FME Data Inspector是一种用于查看任何FME支持格式的数据的工具。它主要用于在转换之前预览数据或在转换后查看数据。



地图瓦片由Stamen Design提供，许可由CC-BY-3.0。数据提供由OpenStreetMapt，许可由CC-BY-SA。

FME工具

除了Workbench和Data Inspector之外，还有其他几个FME实用程序。

- **FME**帮助
  - 用于浏览FME的各种帮助文档的工具。
- **FME**快速转换器
  - FME Workbench的前身，仅用于无需数据变换的快速转换。
- **FME**集成控制台
  - 用于将FME功能嵌入到其他GIS和CAD应用程序（如ArcGIS，AutoCAD，Geomedia和MapInfo）中的工具。
- **FME**许可助理
  - 管理FME许可的申请。

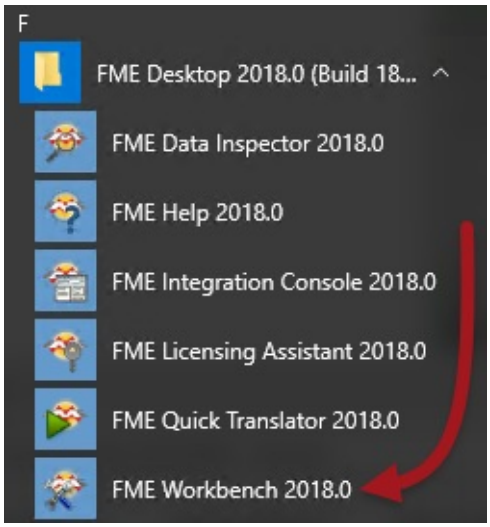
其他**FME Desktop**组件

其他组件也包含在FME Desktop（专业版或更高版本）中。

- **FME**命令行引擎
  - FME命令行引擎允许在命令行级别启动转换。
- **FME Plug-In SDK**
  - FME Plug-In SDK允许开发人员向FME核心添加格式和功能。

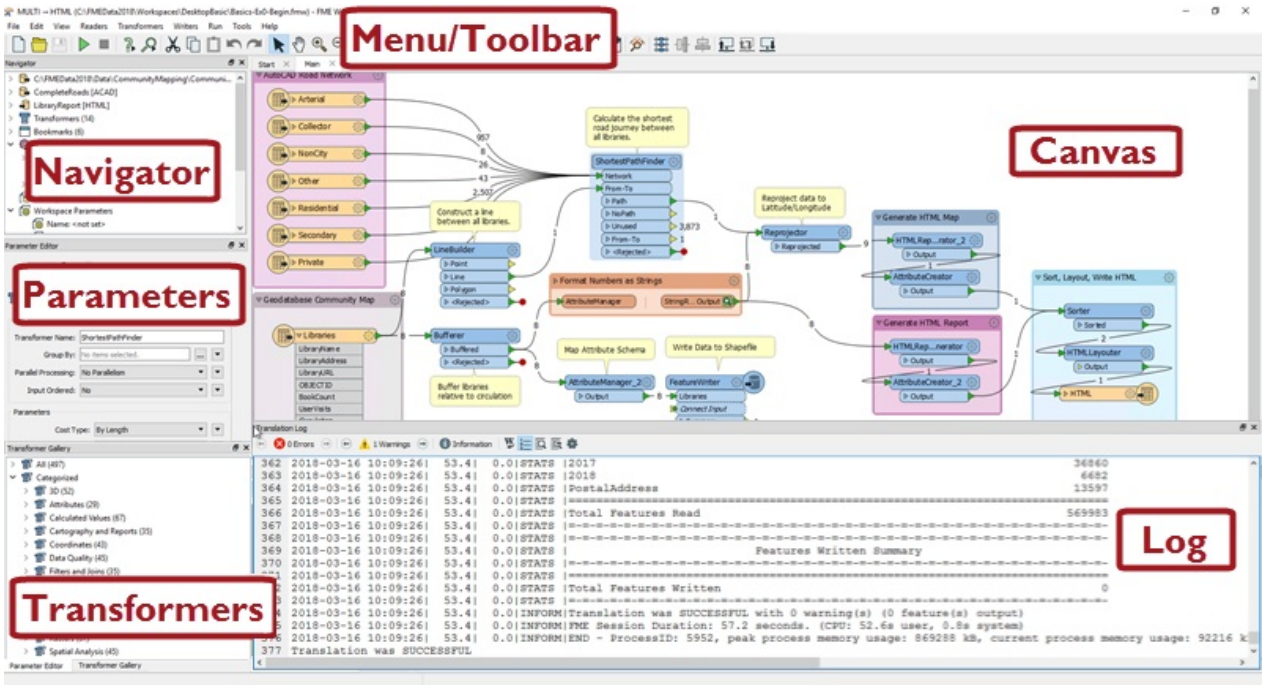
FME Workbench简介

让我们仔细看看FME Workbench，首先是如何启动它。要启动Workbench - 除了双击fmw文件外 - 在Windows开始菜单中找到Workbench：



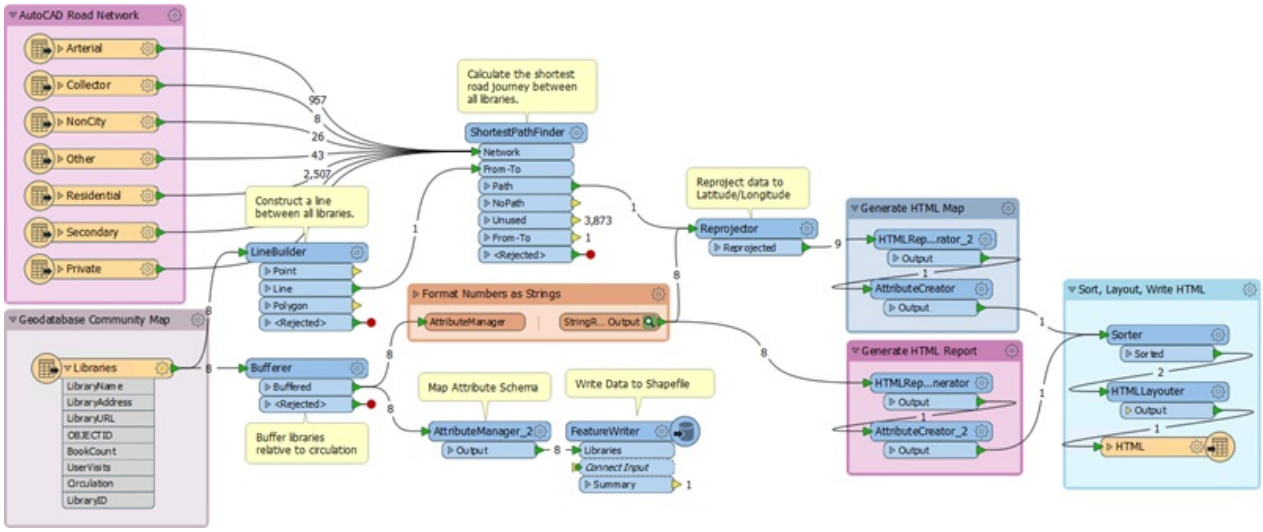
FME Workbench的主要组件

FME Workbench用户界面有许多主要组件：



画布

正如我们所见，FME Workbench画布是定义转换的地方。它是Workbench中的主要窗口：



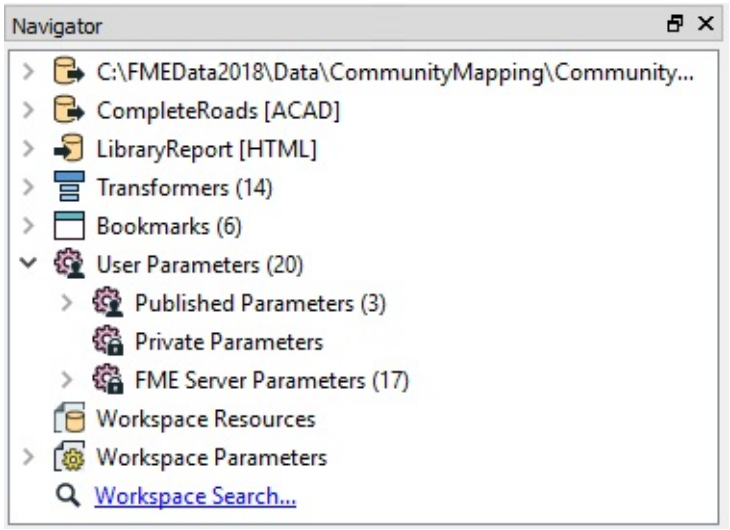
默认情况下，工作空间从左到右读取; 左侧的数据源，中间的转换工具和右侧的数据目标。每个项目之间的连接表示数据流，可以分支在不同的方向，合并在一起，或两者兼而有之。

菜单/工具栏

菜单栏和工具栏包含许多工具：例如，用于在Workbench画布周围导航，控制管理任务以及添加或删除读模块/写模块的工具：

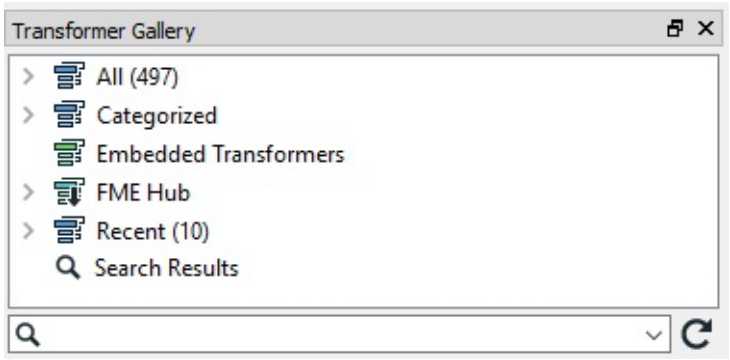
导航器

“导航”窗口是一个结构化的参数列表，用于表示和控制转换的所有组件：



转换器库

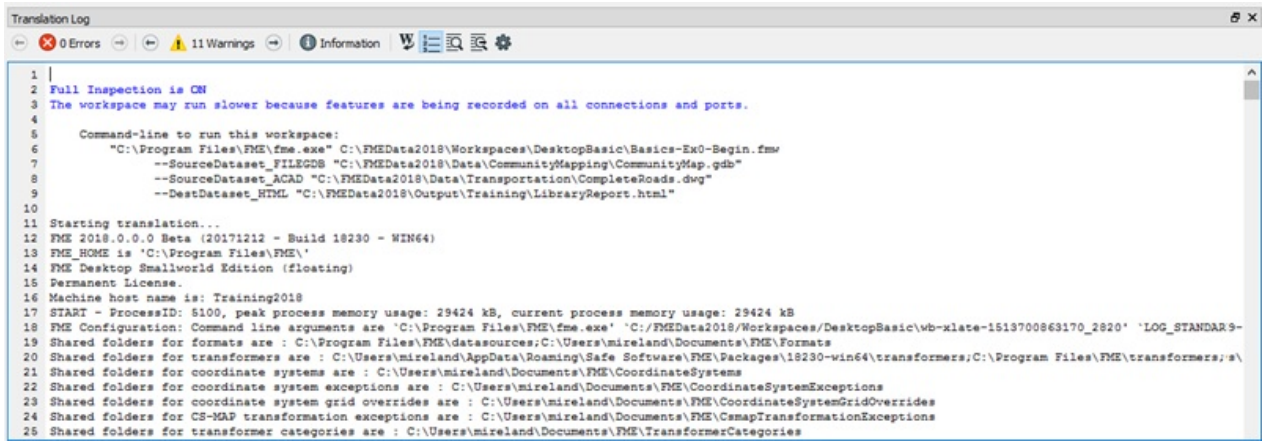
转换器库是用于定位和选择FME转换工具的工具。



转换器的数量（以上，484）将根据FME的版本和安装的任何可选定制转换器而有所不同：

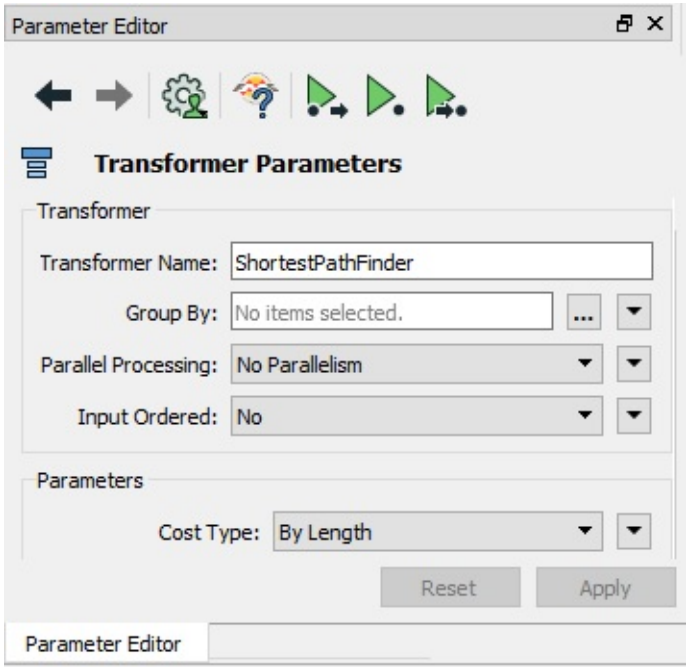
转换日志

转换日志报告转换和其他操作。信息包括任何警告或错误消息，转换状态，转换长度以及处理的要素数量：



参数编辑器窗口

参数编辑器窗口用于编辑画布窗口上对象的参数：



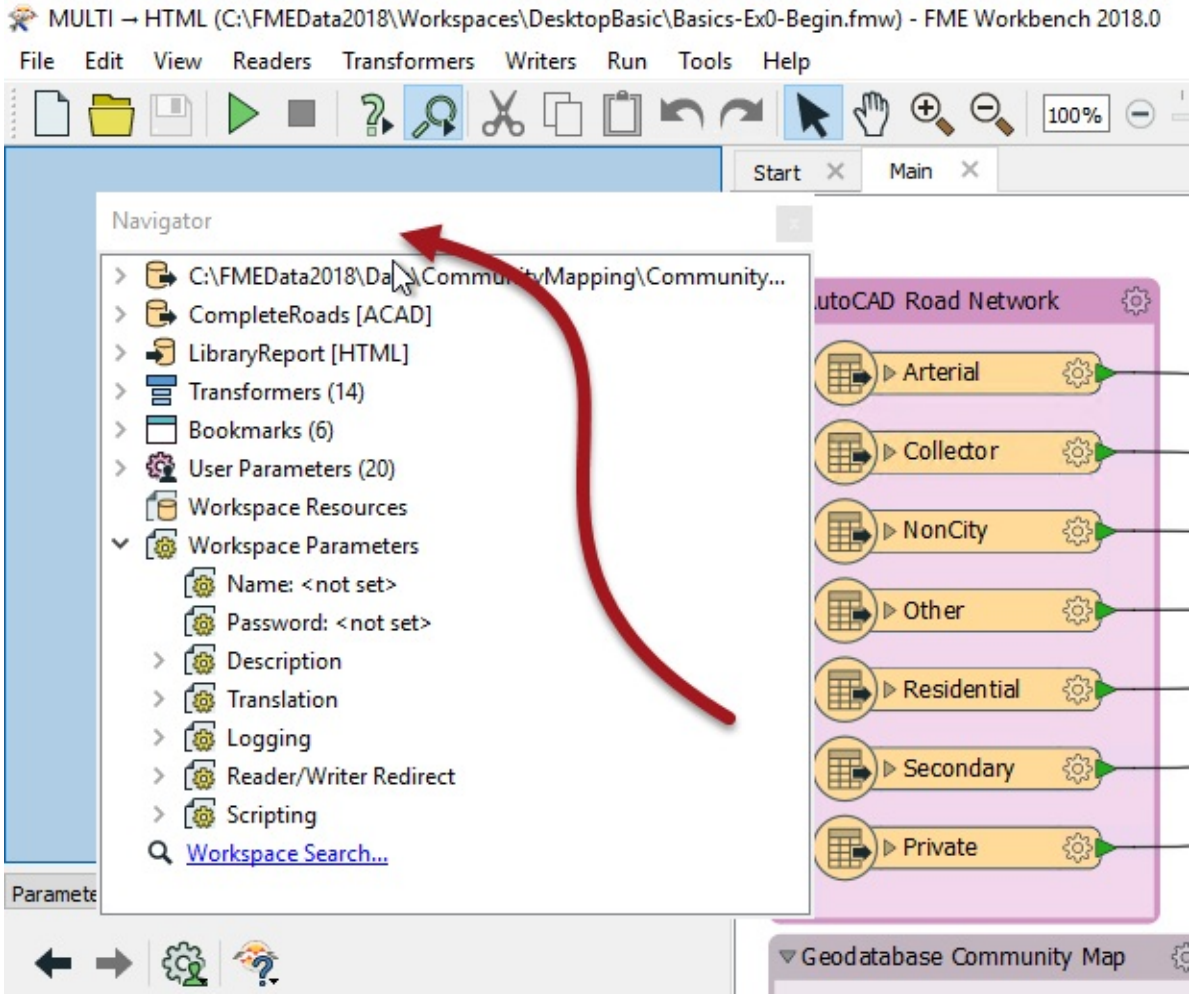
如下一节所示，您可以随意打开或关闭这些窗口，或者将它们重新排列为您喜欢的任何配置。

Vector小姐说.....

随着课程的深入，问题会变得越来越难。不过，这些非常简单： 以下哪些应用程序不属于FME Desktop？ 1. FME Workbench 2. FME集成控制台 3. FME Server控制台 4. FME Data Inspector FME Workbench允许您以哪种方式定义数据流... 1.图形化地 2.心灵感应地 3.有问题地 4.通过编写大量代码用C ++或Java

Workbench中的窗口控件

Workbench中的所有窗口都可以从其默认位置分离并存放在自定义位置。要执行此操作，只需单击窗口框架并将其拖动到新位置即可。



如果窗口放在现有窗口的顶部，那么这两个窗口将成为选项卡式的。

如果窗口放在现有窗口旁边（或两个现有窗口之间），则它们将堆叠在一起。

在上面的屏幕截图中，用户将导航器放在Workbench左侧的位置，堆叠在参数编辑器的顶部。

消防员**Mapp**说.....
你好。我是你友好的街区消防队员，给你一个热门的技巧。不要因缺乏画布空间而感到沮丧。按F11将较少使用的窗口发送到一侧，并将画布窗口展开到惊人的大小！

Vector小姐说.....

尝试使用Workbench窗口（在View菜单下查看）来回答以下问题： 这些中的哪个是一个FME Workbench窗口？

1.数学窗口
2.地理窗口
3.英国文学窗口
4.历史窗口

哪些不是FME Workbench中的窗口排列？

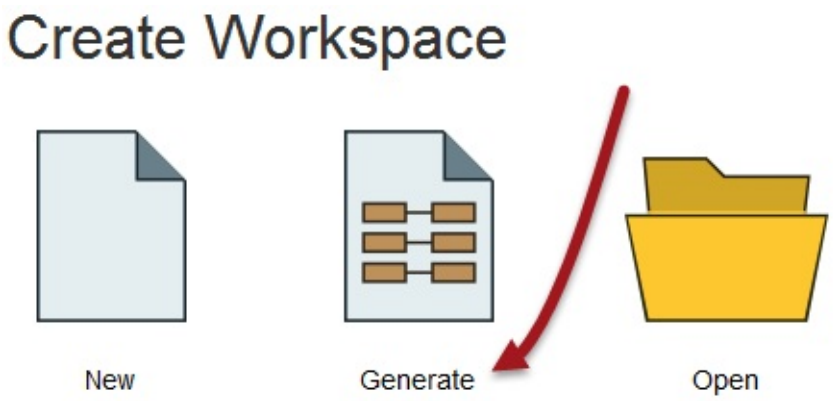
1.堆叠
2.浮动
3.双层玻璃
4.选项卡式的

创建转换

Workbench的直观界面使您可以轻松设置和运行简单的格式到格式（“快速”）转换。

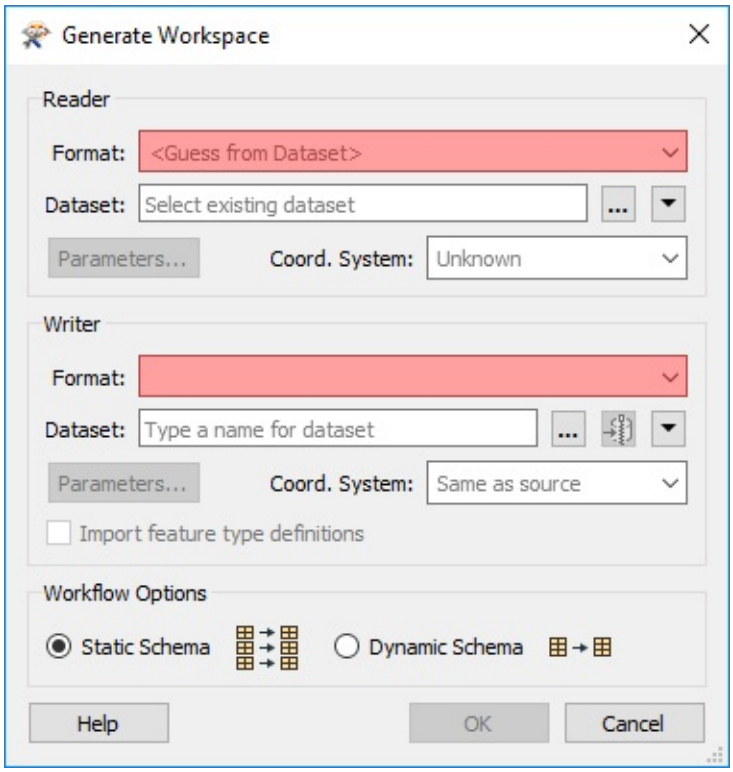
“开始”选项卡

FME Workbench中的“开始”选项卡包含创建或打开工作空间的不同方法。最简单的方法是Generate Workspace：



生成工作空间对话框

“生成工作空间”对话框将所有选项压缩为单个对话框。它具有用于定义要读取的数据和要写入的数据的格式和位置的字段。



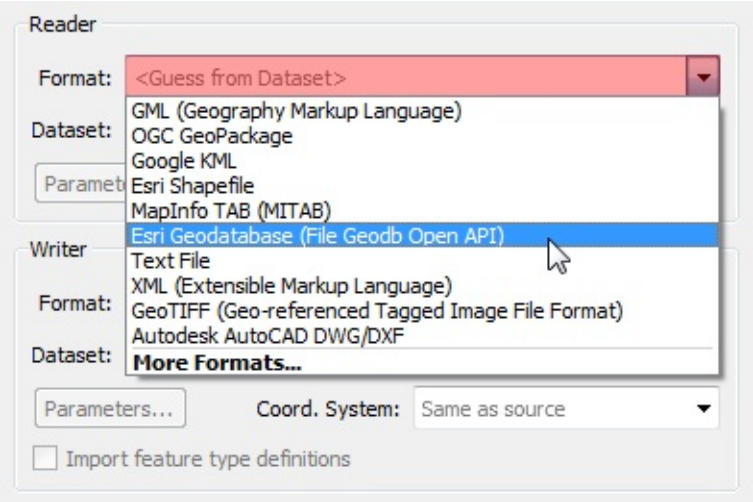
FME对话框中的红色着色表示必填字段。用户必须在这些字段中输入数据才能继续。在大多数对话框中，“确定”按钮将被取消激活，直到必填字段完成。

格式和数据集选择

关键要求是源数据的格式。FME中的所有格式选择字段都是下拉菜单和文本输入字段。

文本输入字段允许您直接键入格式名称。它具有“智能完成”功能，可在您键入时选择紧密匹配。

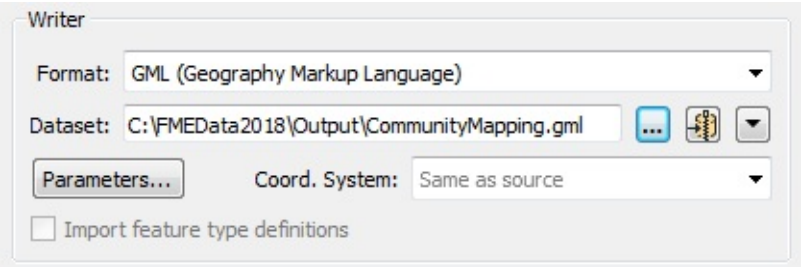
下拉列表显示了一些最常用的格式，因此可以立即获得许多喜爱的格式：



单击“更多格式”，将打开一个表格，显示FME支持的所有格式。

源数据集是另一个关键要求。数据集选择字段是文本输入字段，但具有浏览按钮以打开类似于资源管理器的文件选择对话框。

同样，写模块格式和数据集在此对话框中定义：



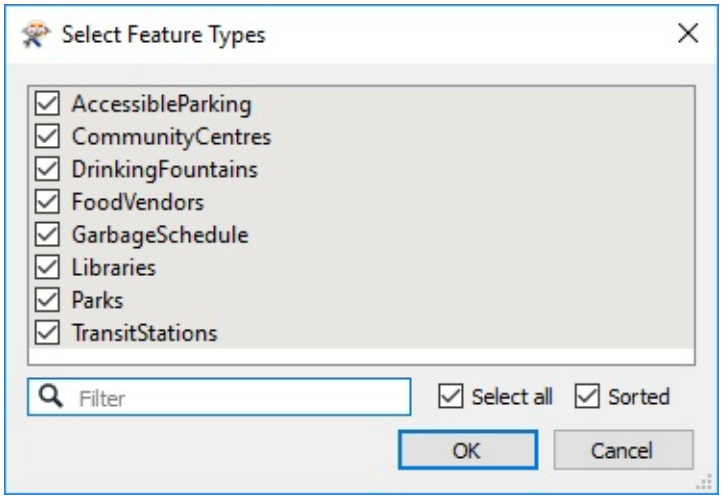
Vector小姐说.....

这是一个你无法用'a'，'b'，'c'或'd'回答的问题！在“生成工作空间”对话框中，为什么在浏览源数据之前设置数据格式有用呢？ 在设置格式类型之前尝试浏览数据集，看看是否可以检测到差异。

要素类对话框

单击“生成工作空间”对话框上的“确定”会导致FME生成已定义的工作空间。但是，只要源数据集包含多个图层，就会首先提示用户选择要转换的图层。

这是通过“选择要素类”对话框实现的。在FME中，要素类是图层的另一个术语。只有选定的图层显示在工作空间中。

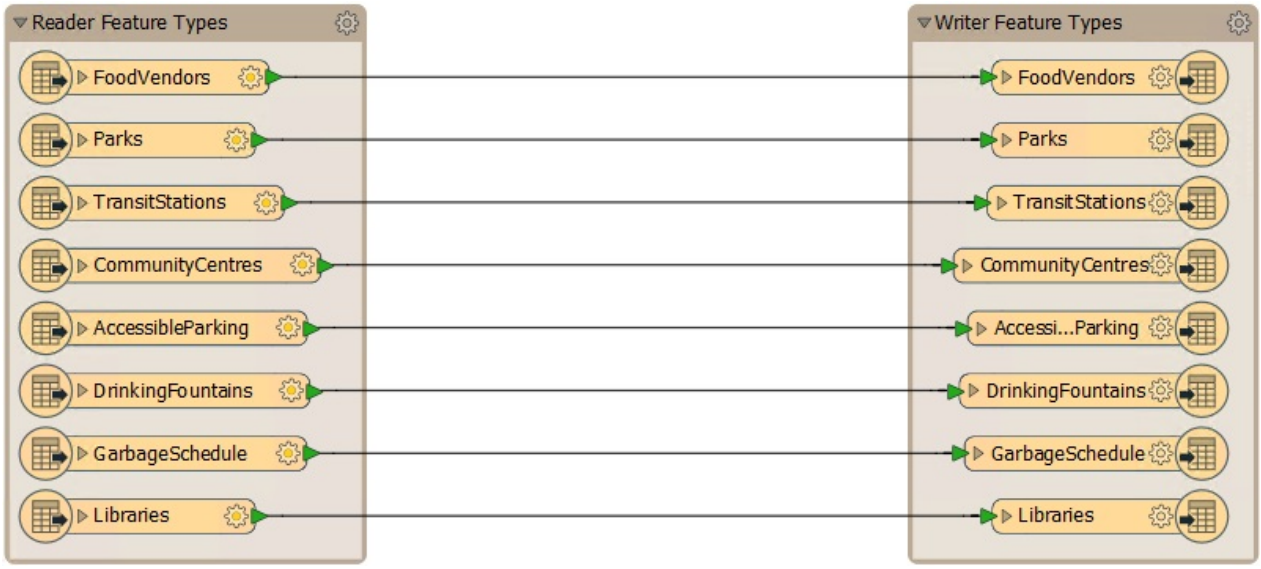


例如，这是一个“选择要素类”对话框，其中用户已选择在工作空间中包含所有可用图层。

新工作空间

新工作空间

新工作空间从左到右，从源（读模块）图层到目标（写模块）图层进行读取。箭头表示数据流的方向：



在上面的屏幕截图中，从一种格式读取八个图层并将其写入另一个格式。

TIP

在大多数情况下，FME使用术语“读模块”和“写模块”而不是“来源”和“目的地”。因此，读模块读取数据集，写模块写入数据集，类似于源/目标和输入/输出。

保存工作空间

工作空间可以保存到文件中，以便以后可以重复使用。工具栏上的保存按钮是执行此操作的一种方法：

还有一些菜单选项可以执行相同的操作，在本例中，文件>保存（快捷键= Ctrl + S）或文件>另存为。默认文件扩展名为.fmw。

Intuitive修女说...

大家好。我是永恒翻译秩序的Intuitive修女. 我在这里展示了FME Workbench界面的一些直观但不太明显的功能。 例如，在菜单栏上选择“文件”>“打开最近”以显示以前使用的工作空间列表。此列表最多可显示15个条目。

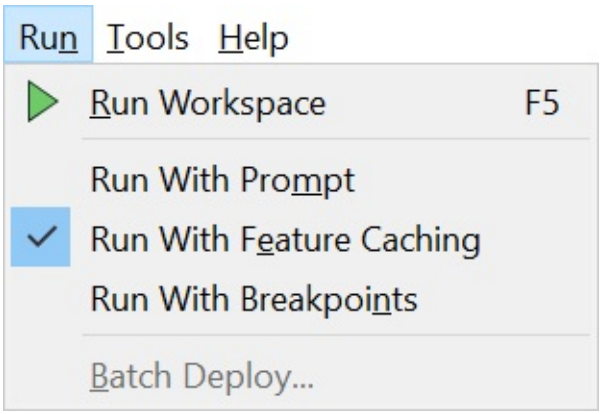
运行工作空间

运行工作空间

Workbench工具栏上的绿色箭头（或“播放”按钮）启动转换：

或者，在菜单栏上的运行下查看：

菜单栏和工具栏上都会显示相同的工具栏选项。请注意，可以使用快捷方式选项F5：



技巧
“运行”按钮的操作可以通过一系列选项进行修改，包括提示用户输入的功能（使用提示符运行），缓存中间数据的功能（使用完全检查运行）以及在调试模式下运行的功能（断点运行）。这些是可以随意打开和关闭的切换选项。

工作空间结果

运行工作空间后，可在转换日志中找到相关信息和统计信息，该日志显示在Workbench日志窗口中。

转换日志显示转换是成功还是失败，从源读取多少要素并写入目标，以及执行转换所需的时间。

```
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
                            Features Read Summary
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
AccessibleParking                                                           42
CommunityCentres                                                            10
DrinkingFountains                                                          113
FoodVendors                                                                 91
GarbageSchedule                                                              6
Libraries                                                                    8
Parks                                                                       69
TransitStations                                                             11
==============================================================================
Total Features Read                                                        350
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
                           Features Written Summary
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
AccessibleParking                                                           42
CommunityCentres                                                            10
DrinkingFountains                                                          113
FoodVendors                                                                 91
GarbageSchedule                                                              6
Libraries                                                                    8
Parks                                                                       69
TransitStations                                                             11
==============================================================================
Total Features Written                                                     350
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
Translation was SUCCESSFUL with 0 warning(s) (8 feature(s) output)
FME Session Duration: 2.1 seconds. (CPU: 1.1s user, 0.5s system)
END - ProcessID: 4084, peak process memory usage: 131024 kB
Translation was SUCCESSFUL
```

在此示例中，日志文件显示已读取350个要素（来自Esri地理数据库）并已写出（到GML数据集）。

整个过程取得了成功，没有任何警告。转换的经过时间是2.1秒。

Vector小姐说……

我打赌你到目前为止所有的问题都是正确的！做得好。现在看看你是否可以得到这些：

其中哪一个不是设置转换格式的方法？
1.键入格式名称
2.从下拉列表中选择格式
3.浏览格式库中的格式
4.选择具有已知文件扩展名的数据集
5.以上都不是（它们都是设置格式的有效方式）

哪个键是运行工作空间的快捷方式？
1. F4
2. F5
3. F5.6
4. F#

练习：基本工作空间创建

练习**2**	基本工作空间创建
数据	分区数据（MapInfo TAB）
总体目标	创建工作空间以将MapInfo TAB格式的分区数据转换为GeoJSON（地理JavaScript对象表示法）
演示	使用FME Workbench创建基本工作空间
启动工作空间	无
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Basics-Ex2-Complete.fmw

恭喜！您刚刚在当地城市的GIS部门担任技术分析师。你的老教师，Vector小姐，给了你一个参考，所以不要让她失望！

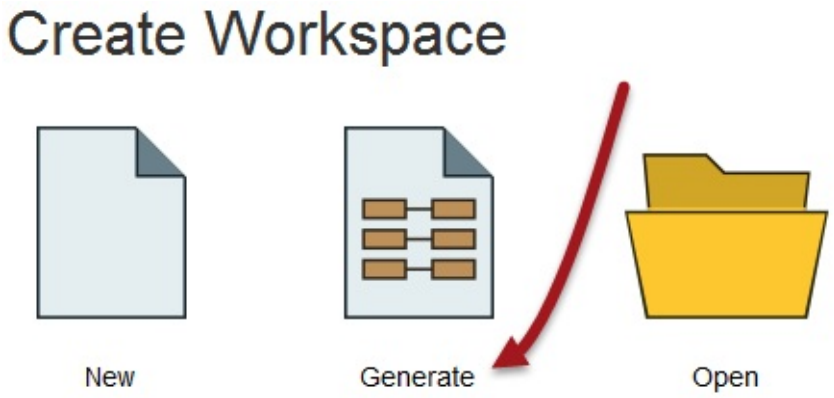
在第一天，您被要求进行简单的文件格式转换。

我们已经概述了您需要采取的所有操作，尽管FME的界面非常直观，您应该能够在不需要这些分步说明的情况下进行练习。

1）启动**FME Workbench**
如果它尚未打开，请从Windows开始菜单中选择它来启动FME Workbench。您可以在开始> FME Desktop 2018.0> FME Workbench 2018.0下找到它。

如果Workbench已打开，请单击主画布上方的“开始”选项卡。

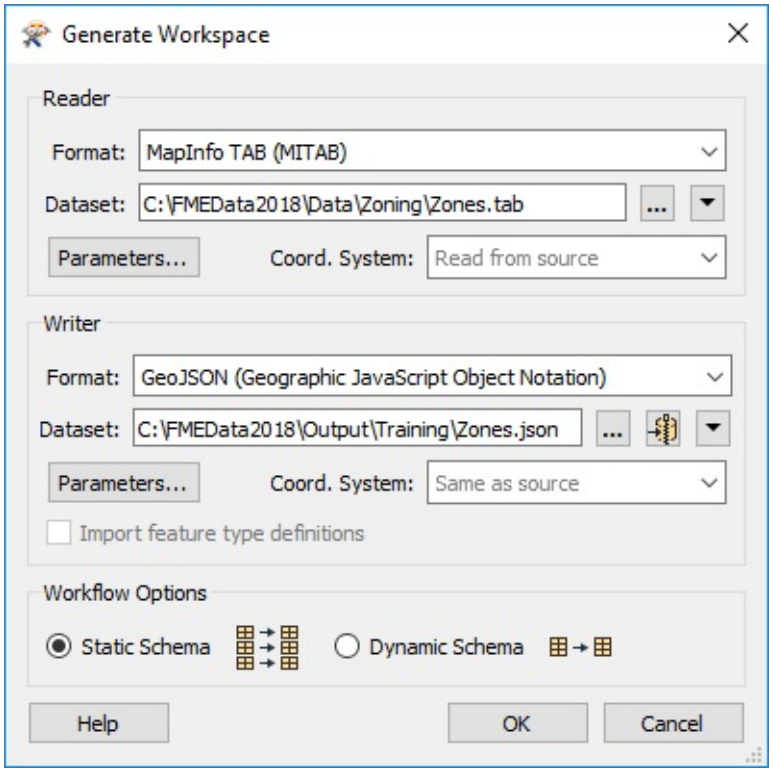
2）选择生成工作空间
在开始页面的生成工作空间部分中选择生成（工作空间）选项。或者，您可以使用快捷键Ctrl + G.



3）定义转换
生成工作空间工具打开一个对话框，用于定义要执行的转换。填写此对话框中的字段，如下所示：

读模块格式	**MapInfo TAB（MITAB）**
读模块数据集	C:\FMEData2018\Data\Zoning\Zones.tab
作家格式	GeoJSON（地理JavaScript对象表示法）
作家数据集	C:\FMEData2018\Output\Training\Zones.json

对话框如下所示：

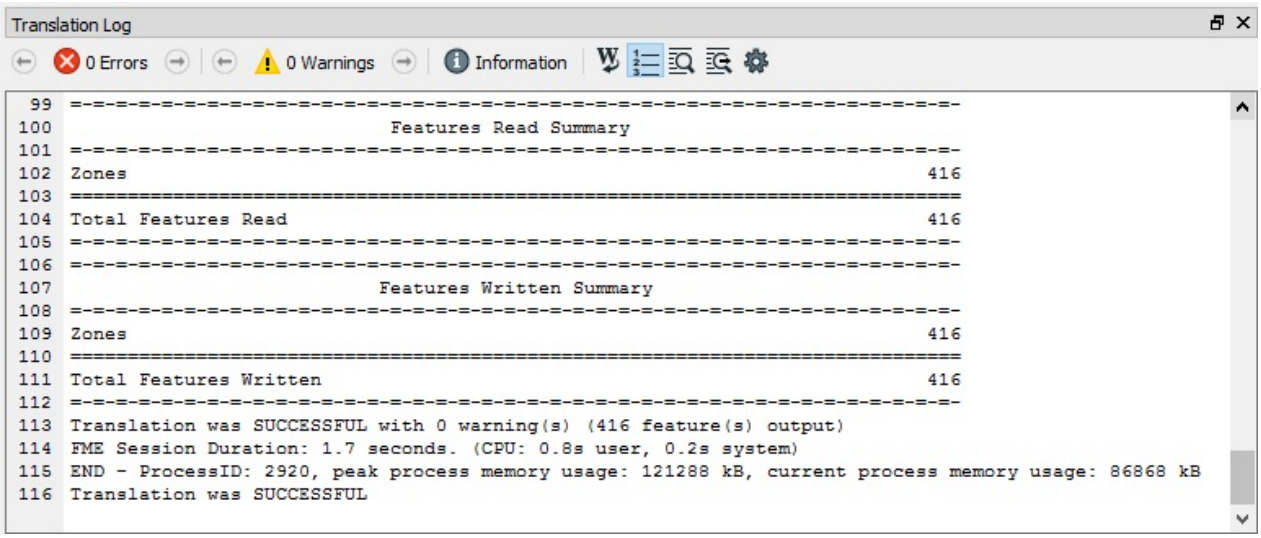


请记住，您可以通过键入名称，从下拉列表中选择格式，或选择“更多格式”并从完整格式表中选择格式来设置格式。现在，忽略工作流选项并保留默认的“静态模式”。

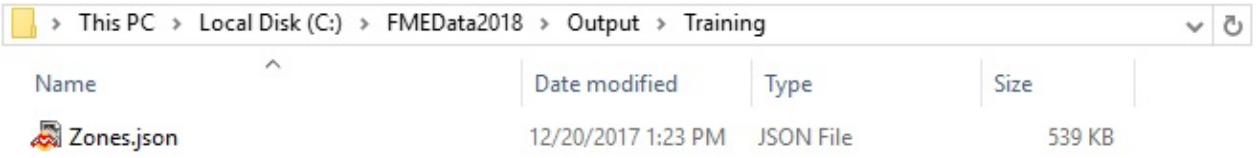
4）生成并检查工作空间
单击“确定”关闭“生成工作空间”对话框。将在FME Workbench画布中生成一个新工作空间：

单击每个对象上的箭头图标即可显示属性列表。

5）运行工作空间
通过单击工具栏上的运行按钮或使用菜单栏上的“运行”>“运行转换”来运行工作空间。工作空间运行，日志文件报告成功转换：



6）定位输出
在Windows资源管理器中找到目标数据，以证明它已按预期写入（不要忘记练习1中的“打开包含文件夹”按钮）。在下一节中，我们将介绍如何直观地检查数据集以确保它是正确的。



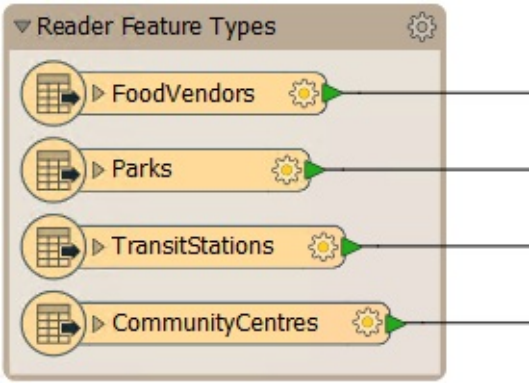
7）保存工作空间
保存工作空间。我们将在以后的练习中使用它。请记住，有一个工具栏保存按钮，在菜单上有文件>另存为。

技巧
如果在没有调整的情况下立即运行转换，则称为“快速转换”。因为FME是一个“语义”转换器，具有增强的数据模型，考虑到目标格式的功能，快速转换的输出尽可能接近结构和意义上的源数据。

恭喜
通过完成本练习，您已学会如何： • 创建FME工作空间 • 运行FME工作空间

读模块参数

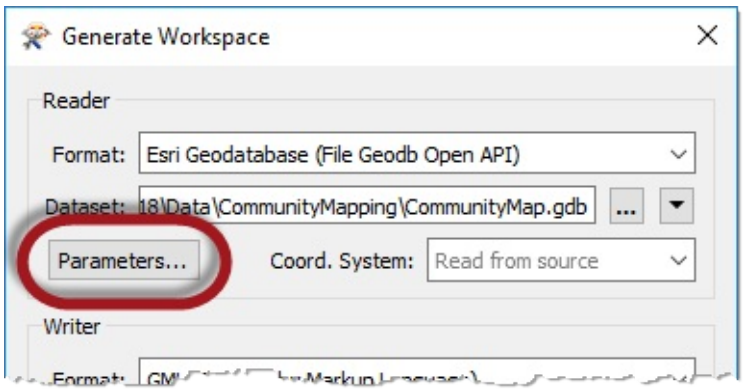
我们知道，工作空间包含一个读取数据集的读模块，该数据集中的每个图层都显示在工作空间画布中：



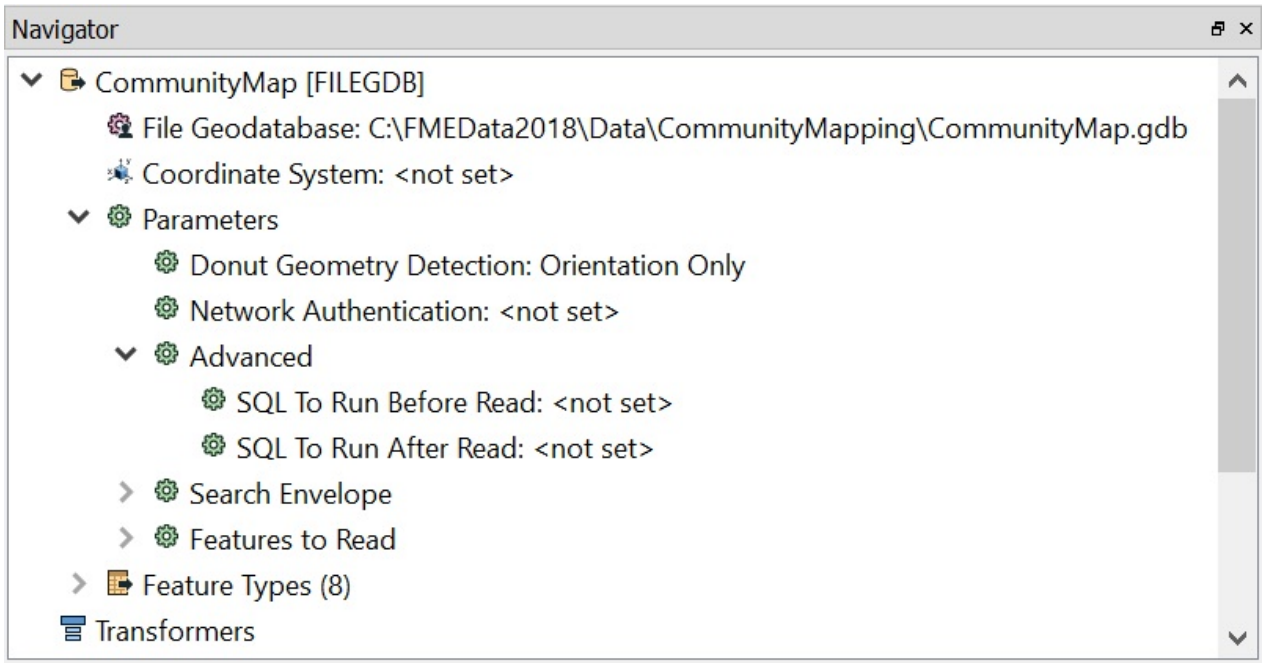
要控制读模块的操作方式，需要使用读模块参数。

查找读模块参数

通过在生成新工作空间时单击“参数”，可以定位和设置读模块参数：



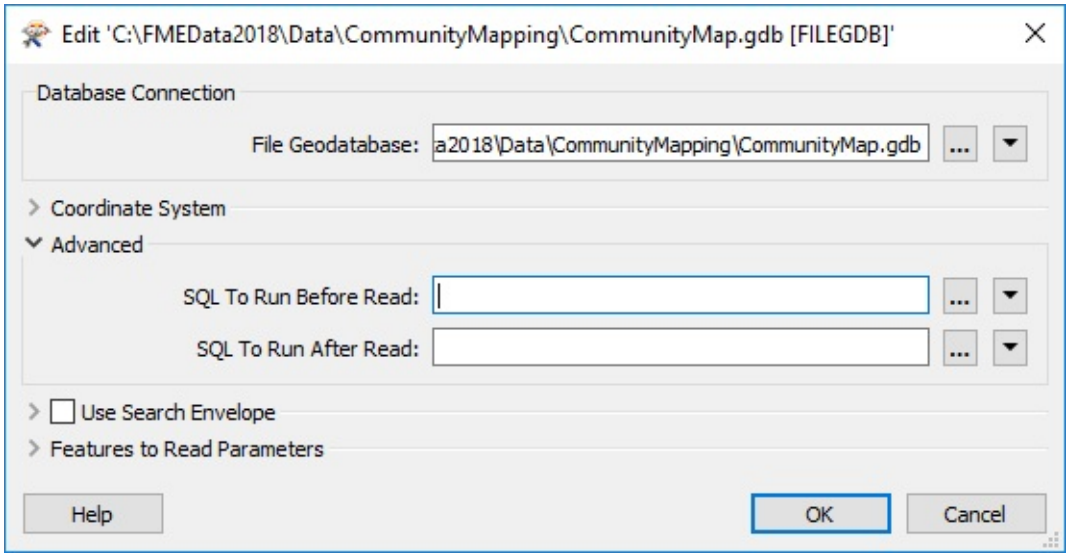
它们也可以在Workbench的导航r窗口中找到：



因为参数是指相关格式的特定组件和特征，所以不同格式的读模块具有不同的控制参数集。

设置读模块参数

要在“导航”窗口中编辑参数，请双击任何参数。这样做会打开一个对话框，其中可以设置参数的值：

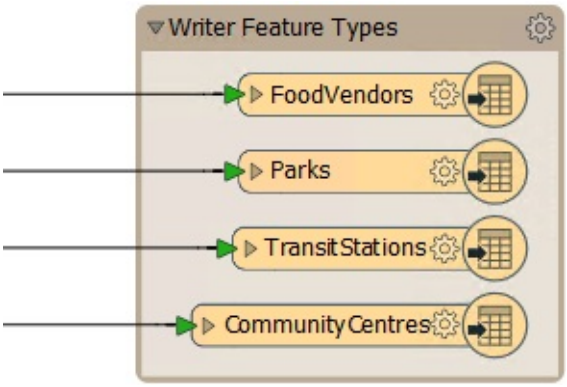


Workbench博士说....

读模块参数控制数据集中的所有要素类型。把它想象成酿造一壶咖啡。咖啡机的强度控制会影响所有倒入的杯子。

写模块参数

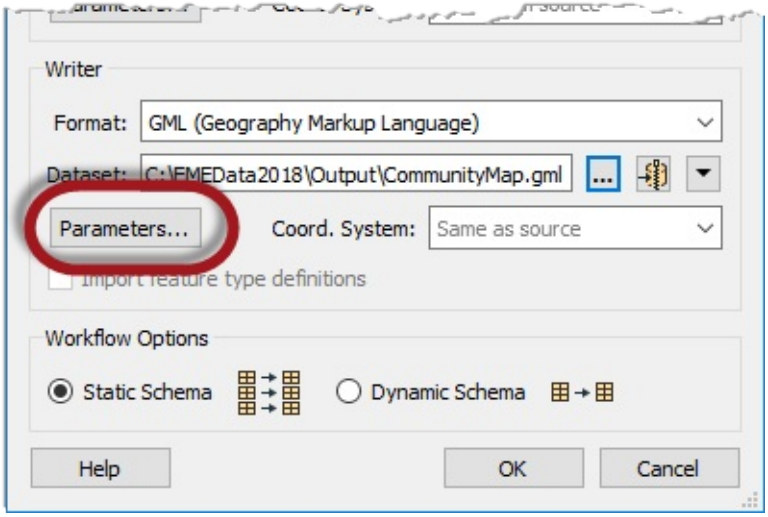
与读模块一样，我们知道工作空间包含编写数据集的写模块，每个要编写的层都显示在工作空间画布中：



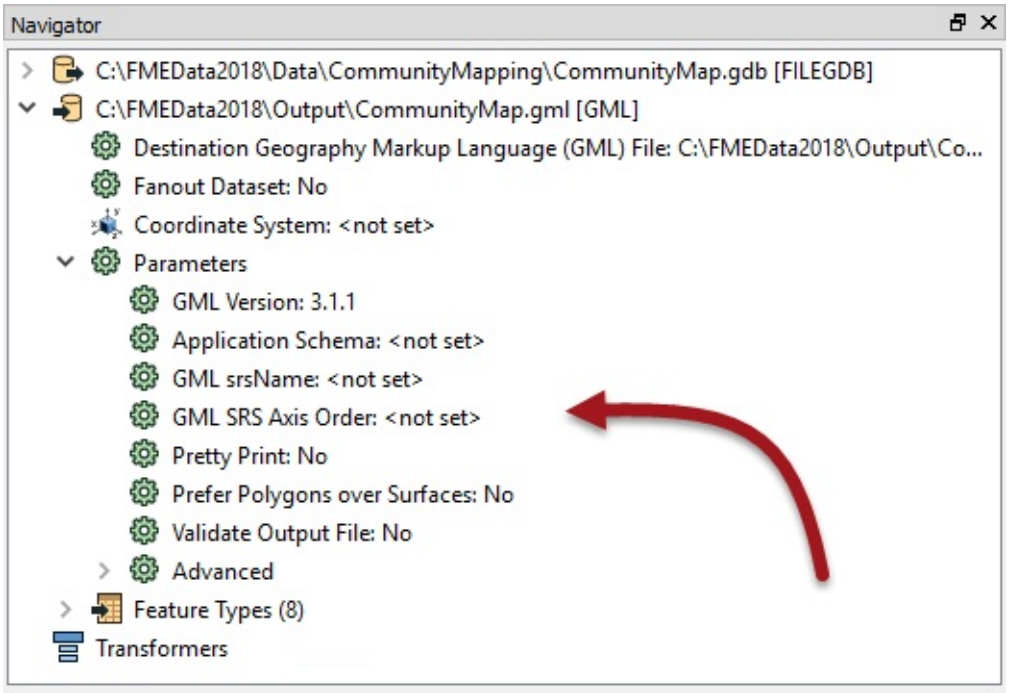
要控制写模块的操作方式，需要使用 写模块参数.

查找写模块参数

通过在生成新工作空间时单击“参数”，可以定位和设置写模块参数：



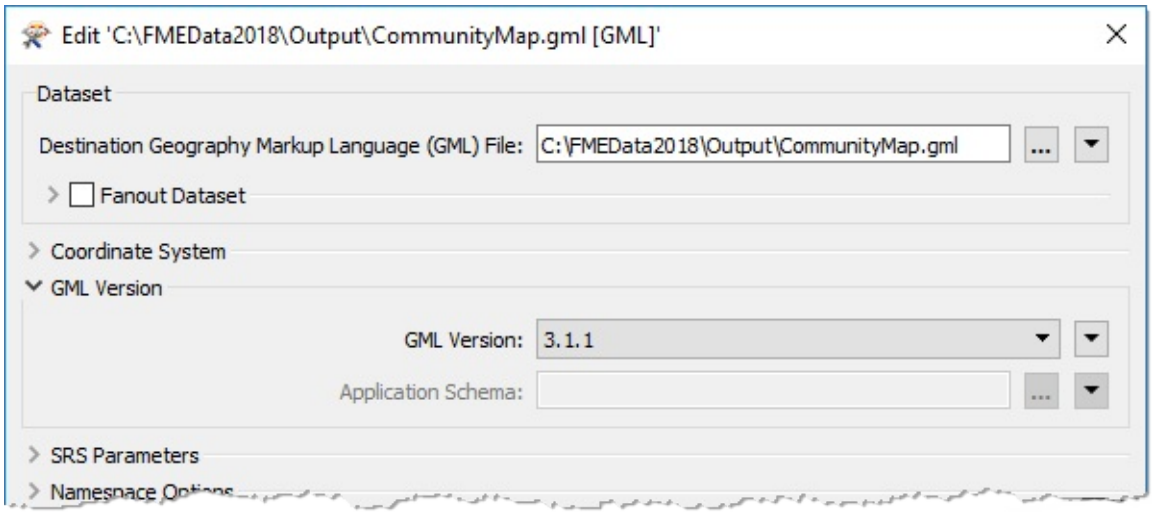
它们也可以在Workbench的导航器窗口中找到：



因为参数指的是相关格式的特定组件和特征，所以不同格式的写模块具有不同的控制参数集。

设置写模块参数

要在"导航"窗口中编辑参数，请双击任何参数。这样做会打开一个对话框，其中可以设置参数的值：



Workbench博士说...

与读模块一样，写模块参数控制数据集中的所有要素类型。例如，在上面的屏幕截图中，所有要素类型都是3.1.1版。 但每种读模块要素类型都有自己的设置（就像每杯咖啡可以用奶油和糖调节一样）。我们将在下一章中找到相关内容。

数据检查

为了确保您处理正确的信息，您需要在转换过程的每个阶段都清楚地查看数据。

数据检查满足了这一需求：它是在转换之前，期间或之后查看数据以进行验证和调试的行为。

什么可以检查？

可以检查数据的许多不同方面，包括以下内容：

- 格式: 数据是否为预期格式？
- 模式: 数据是否细分为正确的层、类别或类?
- 几何对象: 几何对象是否在正确的空间位置？几何类型是否正确？
- 符号: 每个要素的颜色，大小和样式是否正确？
- 属性: 是否存在所有必需属性？是否遵循所有完整性规则？
- 数量: 数据是否包含正确数量的要素？
- 输出: 转换流程是否按预期重组了数据？

主厨**Bimm**说...
你好。我是Bimm主厨，我来这里帮你做一些美味的数据转换。 我有一个很好的方法可以将CAD文件加载到建筑信息模型中。检查成分...我的意思是数据.....在我使用它们之前，让我在它们影响转换之前检测问题。 错误的源层中的要素可能需要重复整个过程。数据检查为我节省了麻烦。现在我知道如果“酱”很好，最后的结果也是如此！

FME Data Inspector简介

在FME中开始检查数据的最佳位置是在一个称为FME Data Inspector的补充应用程序中。

什么是FME Data Inspector?

FME Data Inspector是一个实用程序，允许以任何FME支持的格式查看数据。它主要用于在转换之前预览数据或在转换后进行验证。

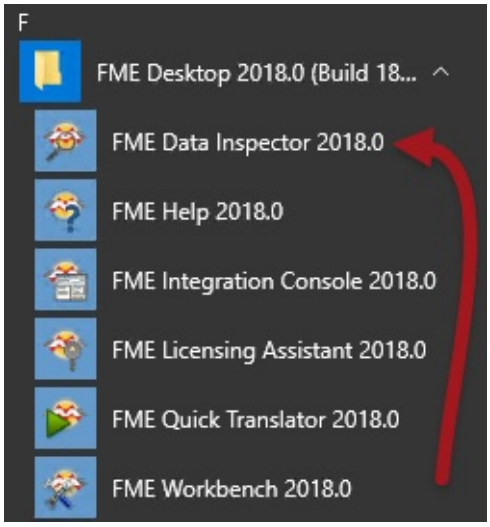
FME Data Inspector与FME Workbench密切相关，因此Workbench可以将数据直接发送给Inspector。它也嵌入在FME Workbench中，通过在转换期间检查数据来帮助设置和调试工作空间。

FME Data Inspector不是什么!

FME Data Inspector不是一种GIS或地图应用程序。它没有分析功能，用于符号系统修改或打印的工具旨在用于用于数据验证，而不是生成地图输出。

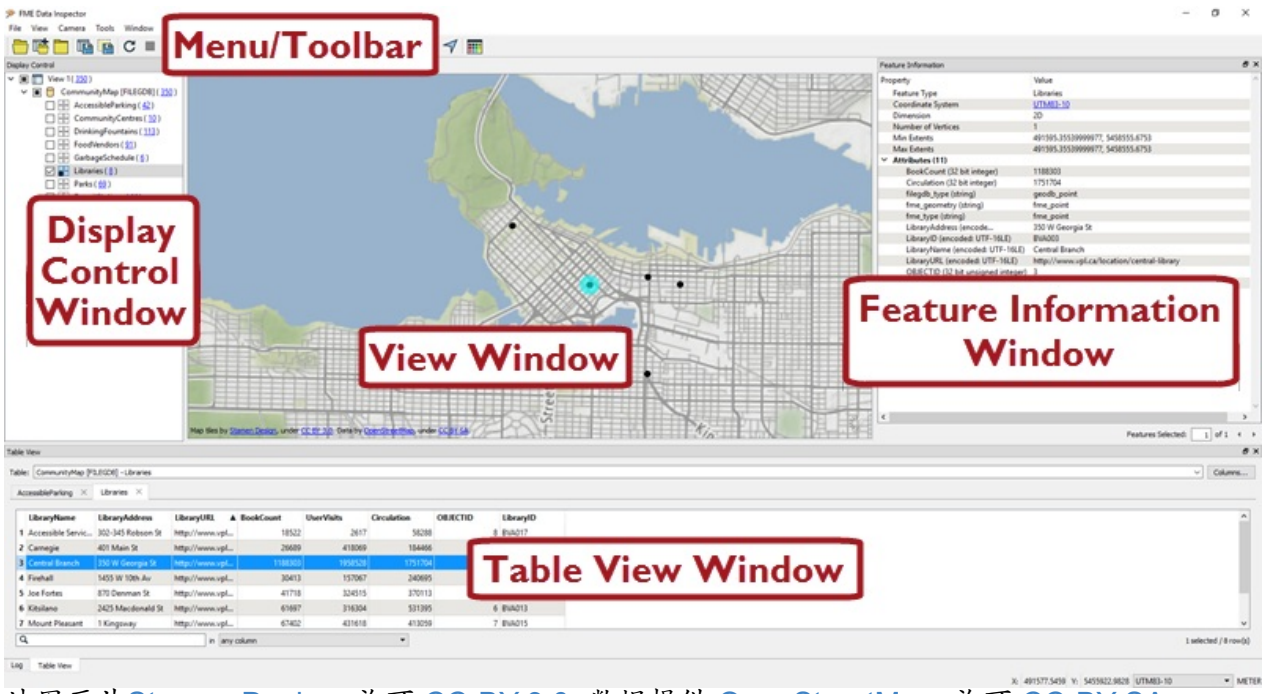
启动FME Data Inspector

要启动Data Inspector，请在Windows开始菜单中找到它：



FME Data Inspector的主要组成部分

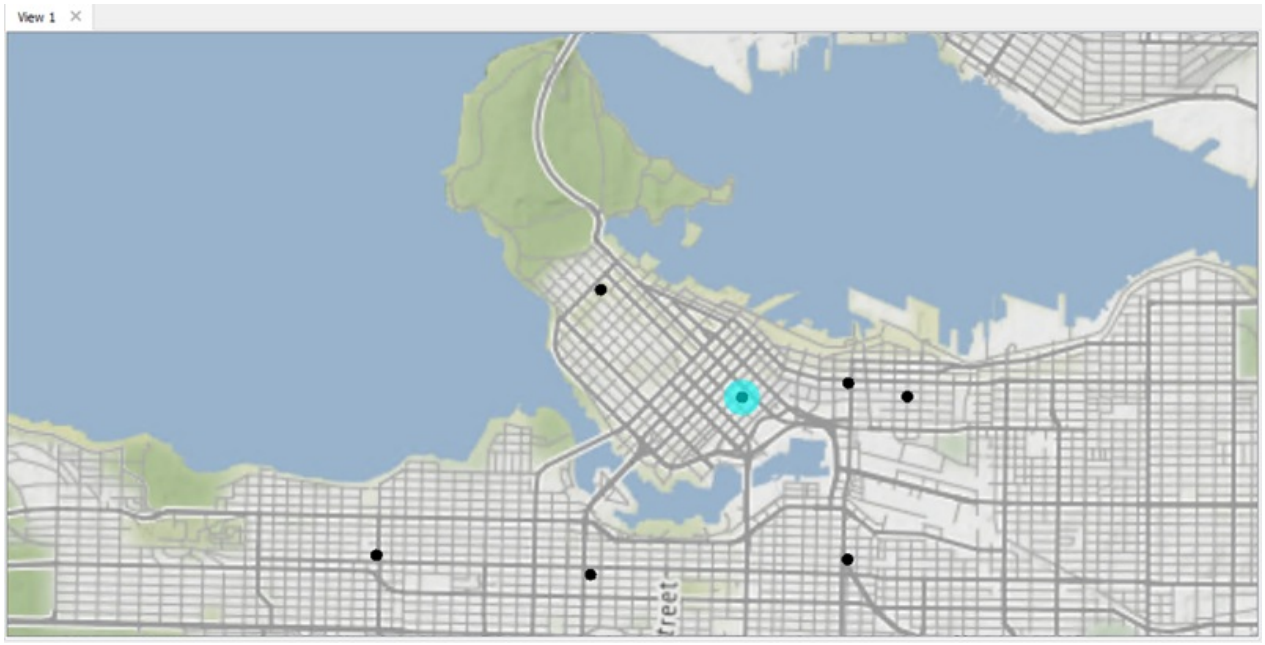
启动FME Data Inspector并打开数据集时，它看起来像这样：



地图瓦片Stamen Design, 许可 CC-BY-3.0. 数据提供 OpenStreetMap, 许可 CC-BY-SA.

视图窗口

“视图”窗口是FME Data Inspector的空间显示区域。可以在任何时间打开不同数据集的多个视图。



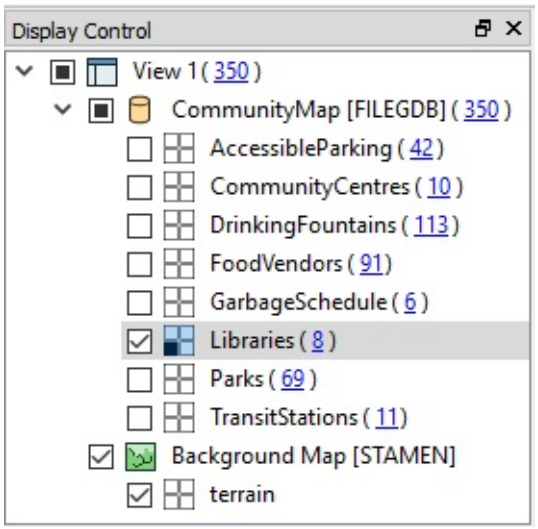
地图瓦片Stamen Design, 许可 CC-BY-3.0.数据提供OpenStreetMap, 许可CC-BY-SA.

菜单栏和工具栏

菜单栏和工具栏包含许多工具。一些用于在视图窗口中导航，一些用于控制管理任务（例如打开或保存数据集），而另一些用于特殊功能，例如选择性过滤数据或创建动态属性。

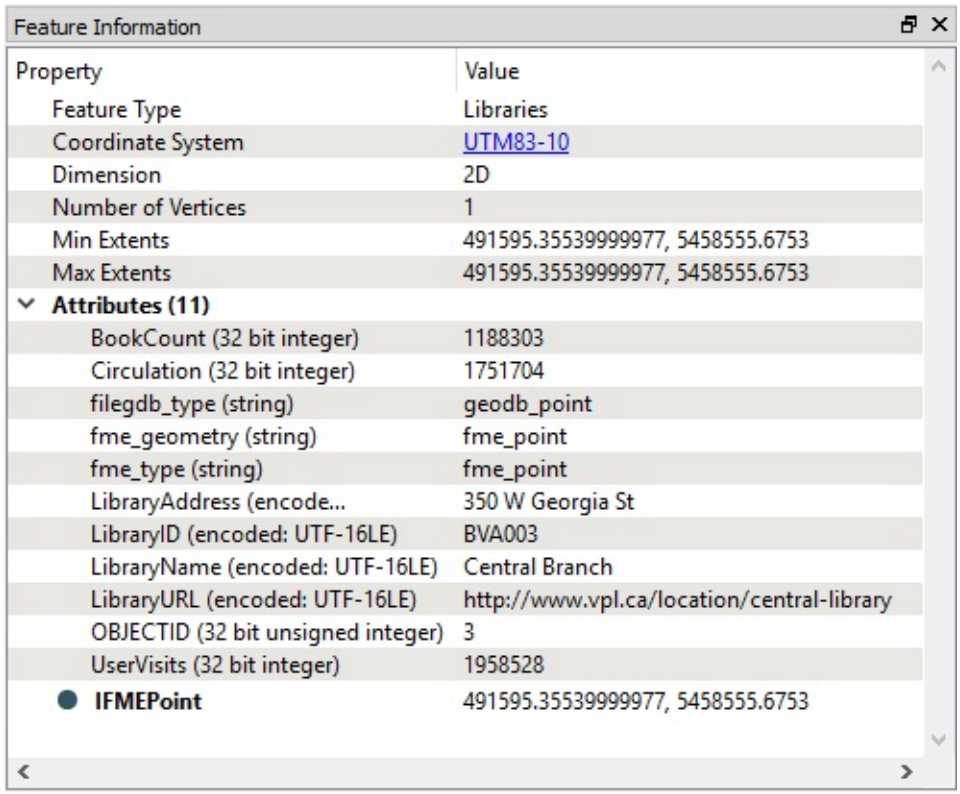
显示控制窗口

“显示控制”窗口显示打开的数据集及其要素类型的列表。此处的工具允许用户在显示中打开或关闭这些工具，更改其符号系统并调整显示顺序。



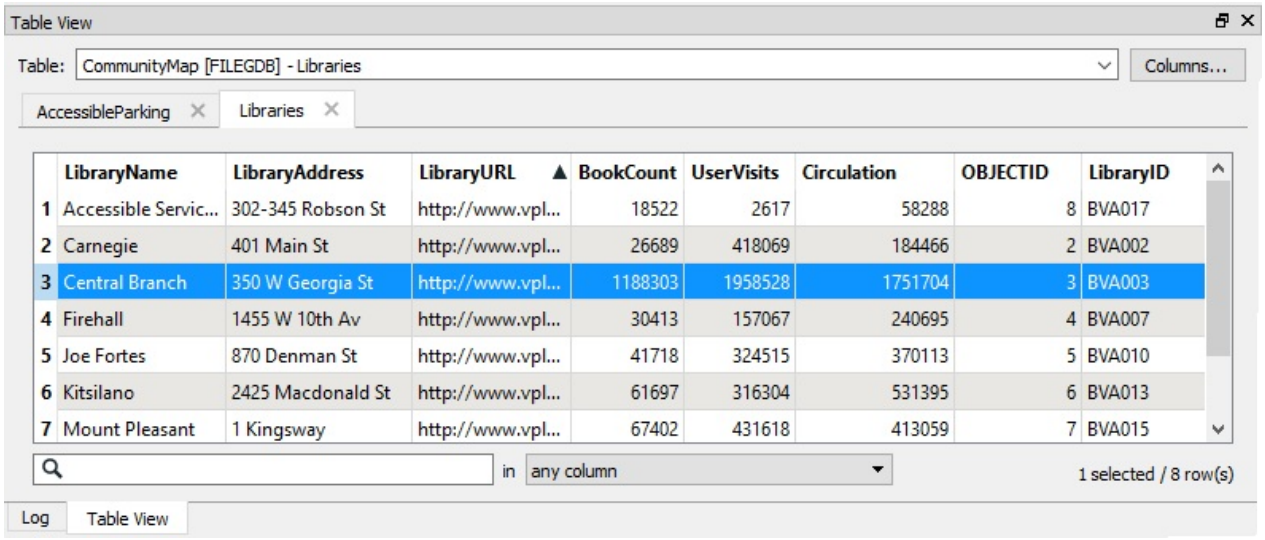
要素信息窗口

当用户在“视图”窗口中查询要素时，“信息”窗口中会显示有关该要素的信息。此信息包括要素的要素类型，属性（用户和格式属性），坐标系以及有关其几何对象的详细信息。



表视图窗口

“表视图”窗口是数据集的类似电子表格的视图，包括所有要素和所有属性，每个要素类型（图层）都有单独的选项卡。



Vector小姐说...

FME非常易于使用，很难想出一些棘手的问题！但我会试试。

当您检查模式时, 您要验证什么？
1. 要素的颜色和线型
2. 要素的数量
3. 要素类型（层，类，表）及其属性
4. 最近的咖啡店是哪里

在FME Data Inspector中查看数据

使用FME Data Inspector，可以轻松打开和查看任意数量的数据集并查询其中的要素。

查看数据

FME Data Inspector提供了两种查看数据的方法：打开或添加。

打开 一个数据集将打开一个新的视图窗口，供其显示。 添加 一个数据集将在现有视图窗口中显示数据; 这样，可以同时查看多个数据集。

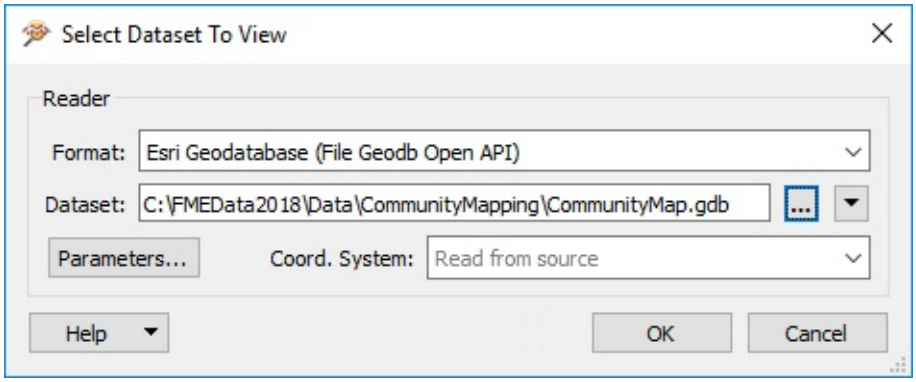
打开一个数据集

可以通过多种方式在FME Data Inspector中打开数据集。

- 从菜单栏中选择“文件”>“打开数据集”
- 选择工具栏按钮打开数据集。
- 将文件拖放到任何窗口（“视图”窗口除外）
- 从Workbench中打开

只需右键单击画布要素类型（源或目标）并选择“检查”选项，即可在FME Workbench中打开数据。

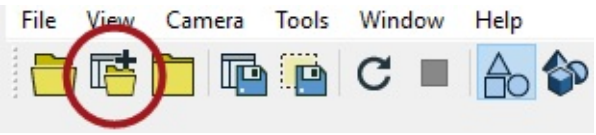
所有这些方法都会导致在FME Data Inspector中打开一个对话框，用于定义要查看的数据集。



添加数据集

打开数据集会导致创建新的“视图”选项卡并显示数据。要在现有视图选项卡中打开数据集，需要使用工具添加数据集。

- 从菜单栏中选择“文件”>“添加数据集”
- 选择工具栏按钮添加数据集
- 将文件拖放到视图窗口中



窗口工具

一旦在FME Data Inspector中打开数据后，有许多工具可用于更改视图。

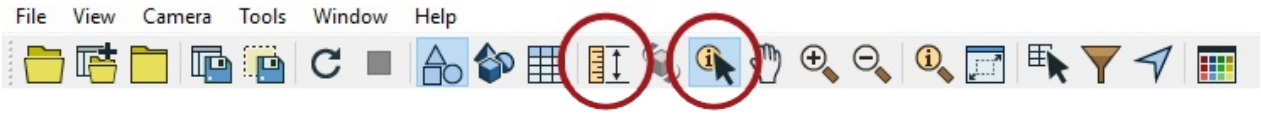
- 拖动
- 放大
- 缩小
- 缩放到所选要素
- 缩放到数据的完整范围

技巧
按键盘上的Shift键，它将激活Inspector中的放大工具。按Ctrl键，将激活缩小工具。释放键以恢复到以前的工具。 此功能允许用户在按键时快速在查询和导航模式之间切换，因此无需在菜单栏或工具栏上的查询和导航工具之间进行单击。

查询FMEData Inspector中的数据

FME Data Inspector包含多个查询工具，但其中两个特别重要：

- 查询个别要素
- 在视图窗口中测量距离

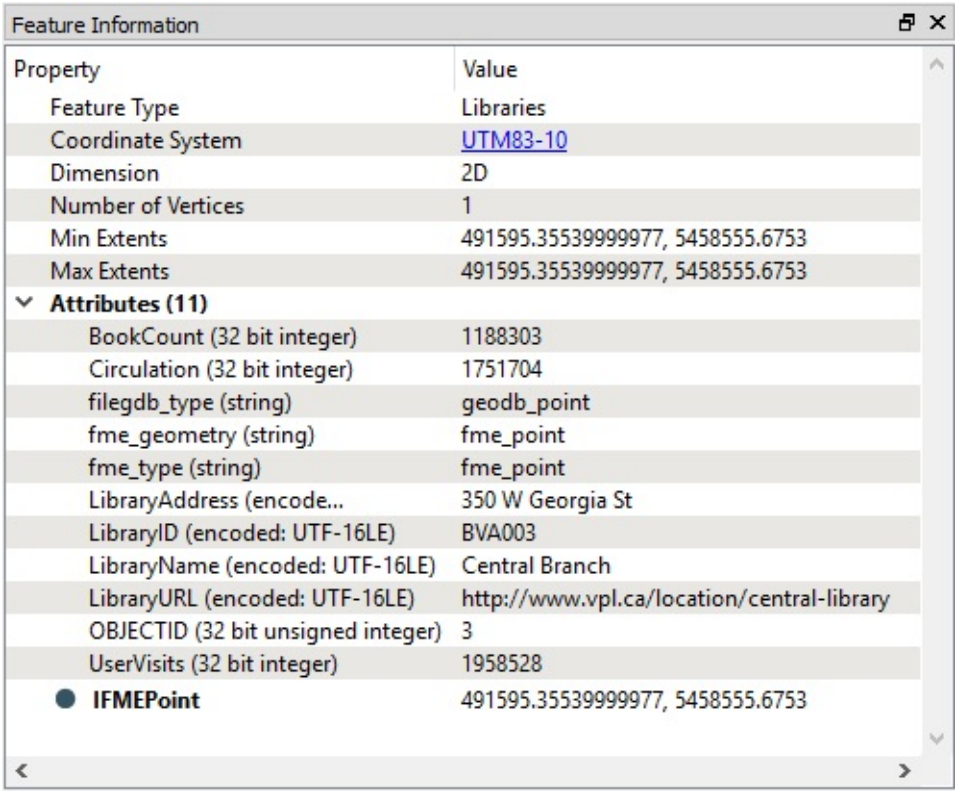


查询工具按钮就像一个切换键。默认情况下，它在您启动FME Data Inspector时处于活动状态; 如果再次单击它 - 或选择窗口工具 - 则关闭查询工具。

要素查询的结果显示在“要素信息”窗口中。

要素信息窗口

此窗口的上半部分报告有关该要素的一般信息; 它所属的要素类型（图层/表格），它所处的坐标系，是二维还是三维，以及它拥有多少个顶点。

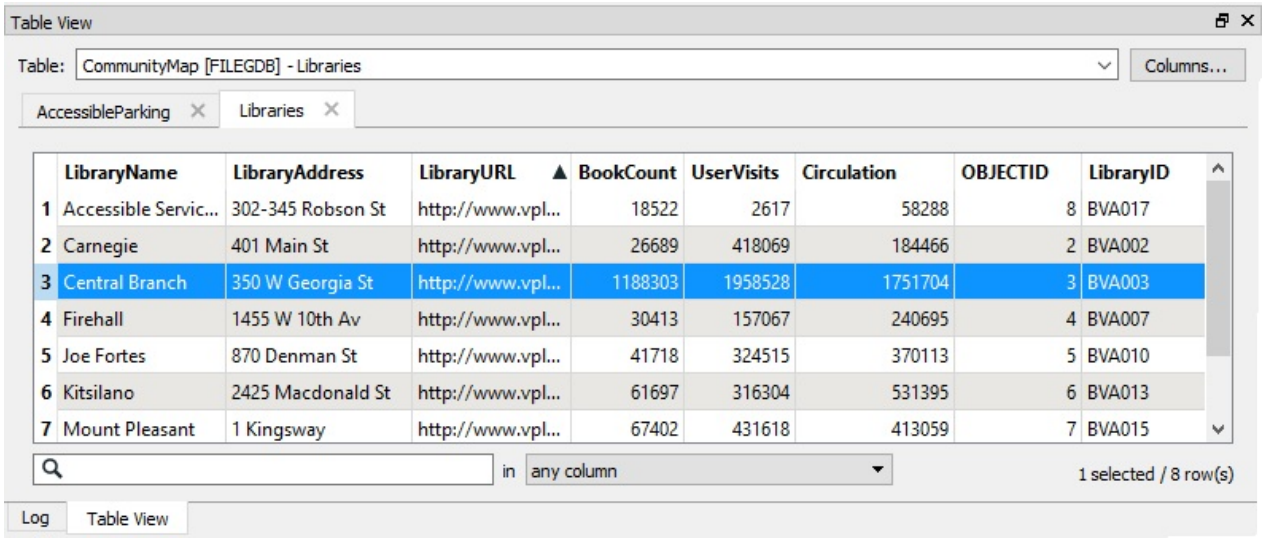


中间部分报告与该要素关联的属性，包括用户属性和格式属性（例如*fme_type*）。

下半部分报告要素的几何形状。它包括几何类型和构成要素的坐标列表。

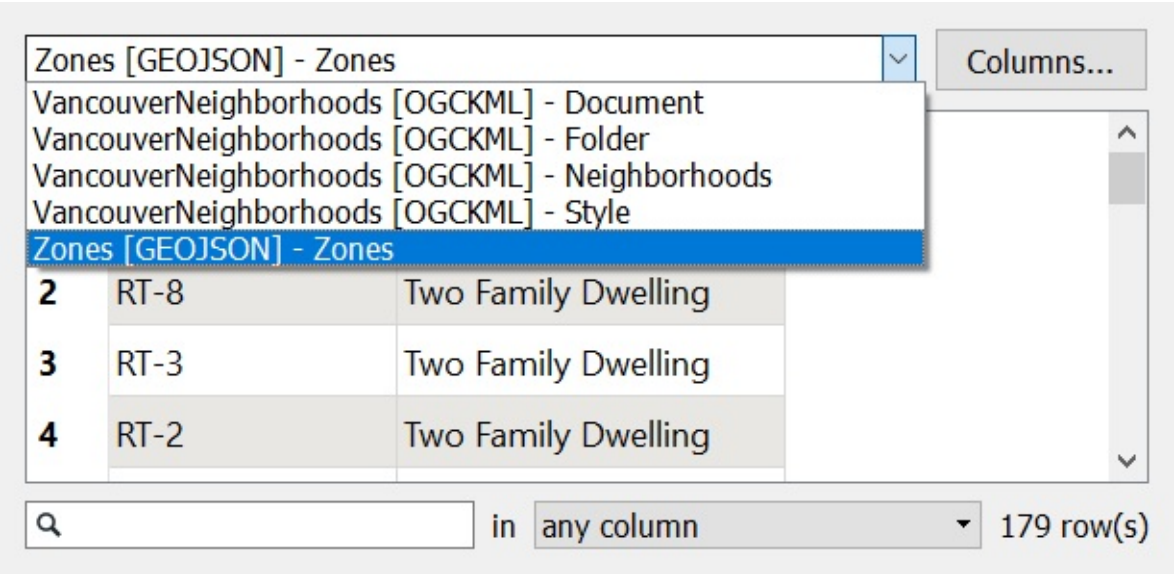
表视图窗口

还有一个名为表视图的窗口。



表视图是一种以表格式电子表格式布局检查数据的方法。虽然它没有信息窗口所示的相同信息深度，但表格视图对于同时检查多个要素的属性值特别有用。

您可以通过单击表视图顶部的下拉菜单在表视图中的要素类型之间来回切换。



技巧

为了提高性能，只有从下拉列表中选择或在当前视图窗口中查询时，才会自动显示表格。

Vector小姐说.....

启动FME Data Inspector并打开数据集。在“表视图”窗口中，右键单击记录和列标题以查看上下文菜单。以下哪项不是可用的菜单选项：

1.排序（按字母顺序或数字顺序）
2.检查值
3.剪切/复制/粘贴
4.将所选数据另存为

练习：基本数据检查

练习**3**	基本数据检查
数据	分区数据（GeoJSON） 街区（Google KML）
总体目标	检查先前转换的输出
演示	使用FME Data Inspector进行基本数据检查
启动工作空间	无
结束工作空间	无

在上一个练习中，您被要求在格式之间转换一些数据。在发送转换后的数据之前，您应该检查它以确保它是正确的。让我们通过检查快速转换的输出来了解FME Data Inspector接口的工作原理。

1）启动**FME Data Inspector**
从Windows开始菜单中选择FME Data Inspector，启动它。您可以在“开始”>“FME Desktop 2018.0”>“FME Data Inspector 2018.0”下找到它。

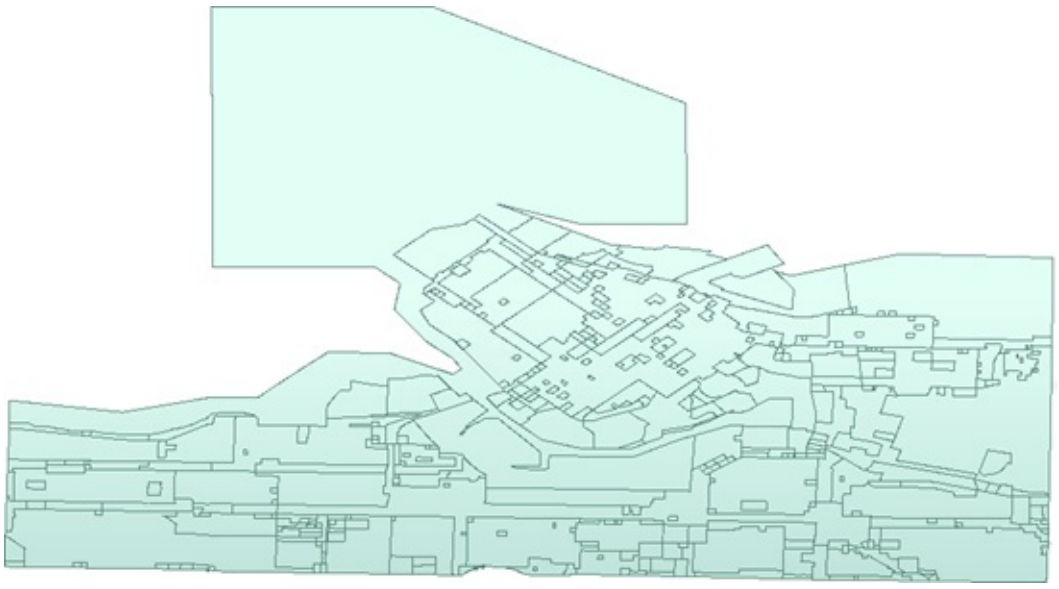
2）打开数据集
FME Data Inspector将启动并以空视图显示开始。

要打开数据集，请从菜单栏中选择“文件”>“打开数据集”。出现提示时，按如下所示填写“选择数据集”对话框中的字段：

读模块格式	**GeoJSON**（地理**JavaScript**对象表示法）
读模块数据集	C:\FMEData2018\Output\Training\Zones.json

注意：如果找不到数据集 - 也许您没有完成第一次练习，或者将数据写入其他位置 - 那么您可以按照练习2中的描述打开原始分区数据集。

GeoJSON数据集如下所示：



3）浏览数据
使用工具栏上的窗口工具浏览数据集，仔细检查。使用“查询”工具查询单个要素并检查“要素信息”窗口中的信息。

尝试右键单击不同的Data Inspector窗口，以发现上下文菜单中存在的要素。

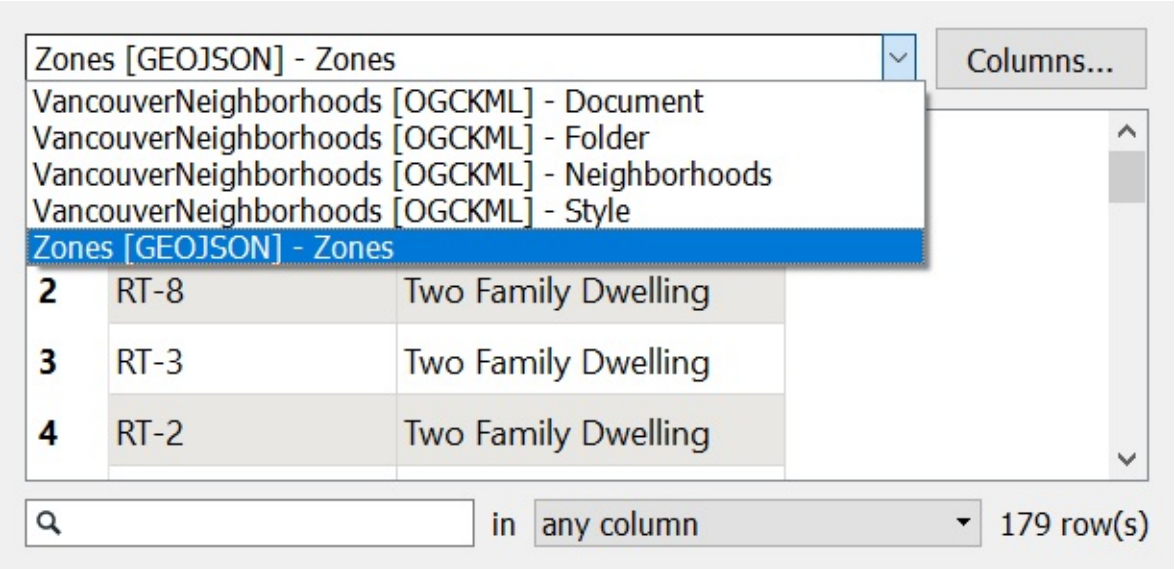
4）添加数据集
让我们在显示屏上添加第二个数据集，以便与我们的分区数据进行比较。该数据集将是街区边界的KML文件。然后我们将能够看到每个区域重叠的街区。

要添加数据集，请从菜单栏中选择“文件”>“添加数据集”。出现提示时，按如下所示填写“选择数据集”对话框中的字段：

读模块格式	谷歌**KML**
读模块数据集	C:\FMEData2018\Data\Boundaries\VancouverNeighborhoods.kml

显示现在看起来像这样：

使用“表视图”练习检查每种要素类型的表格数据。单击表视图顶部的下拉箭头，在Zones.json和Neighborhoods.kml表之间来回切换：



虽然视图窗口中的图层看起来很混乱，但我们可以清理显示，我们很快就会发现。

恭喜
通过完成本练习，您已学会如何： • 在FME Data Inspector的新视图中打开数据集 • 使用FME Data Inspector中的窗口和检查工具 • 使用FME Workbench功能在FME Data Inspector中打开数据集

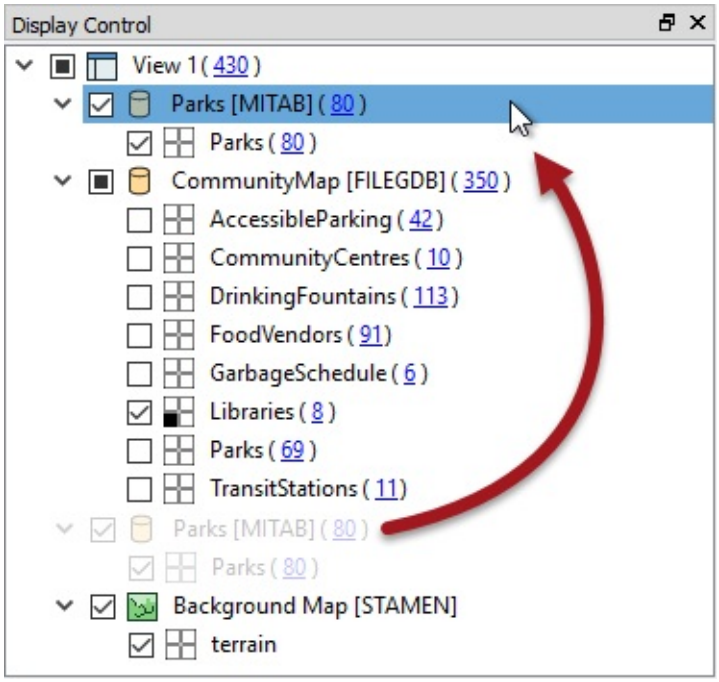
Data Inspector显示控件

FME Data Inspector具有许多控件以帮助以有序的方式显示数据。

显示控制窗口

要素显示顺序的管理在“显示控制”窗口中执行。

可以将每个数据集和要素类型拖到任何其他数据集和要素上，以在“视图”窗口中提升其显示顺序：

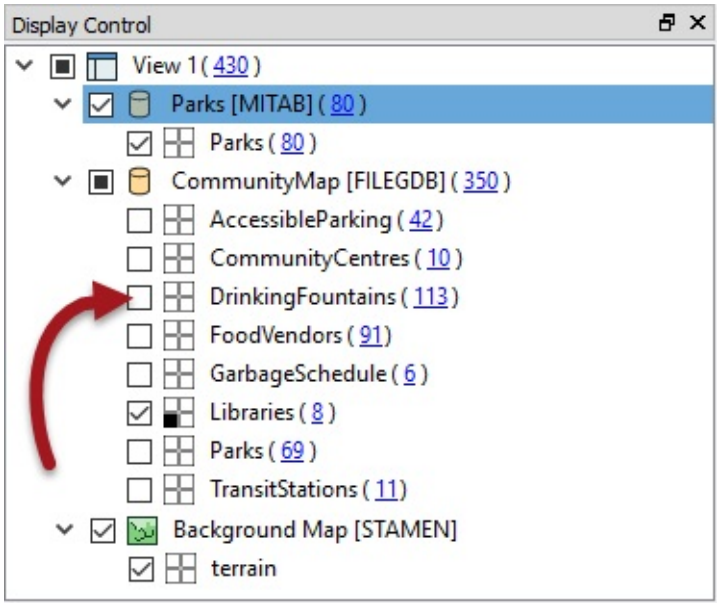


要素类型只能在其容器数据集中进行排序。

在显示中，“库”可以提升到“GarbageSchedule”之上，但它们本身不能提升到单独的Parks数据集之上。

显示状态

“显示控制”窗口的每个级别都有一个复选框，用于在该级别打开和关闭数据：

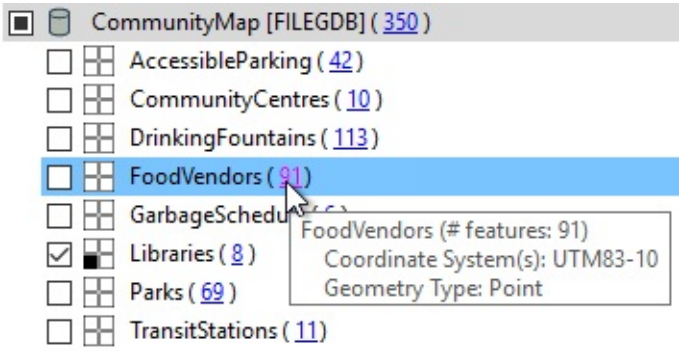


您可以选择一次关闭单个图层或整个数据集。

要素计数

“显示控制”窗口中的每个“要素类型”都标有其包含的要素数，位于要素类型名称后面的括号中。因此，在上一张图片中，我们可以看到温哥华市有113个饮水机（或113个名为DrinkFountains的要素）。

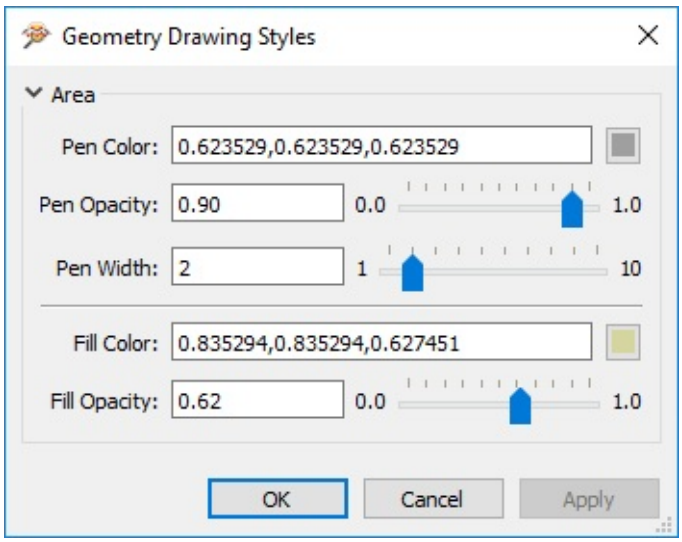
单击要素计数将选择要在“要素信息”和“表格视图”窗口中显示的所有要素，并激活工具“文件”>“将所选数据另存为：”：



样式

可以为每种要素类型指定适用于其几何类型的不同颜色或样式。要访问此要素，请在“显示控制”窗口中单击该要素类型的样式图标：

“绘图样式”对话框允许您为要素设置所有方式的符号系统。这包括要素颜色（和透明度），点图标/符号，点大小，线条粗细等：

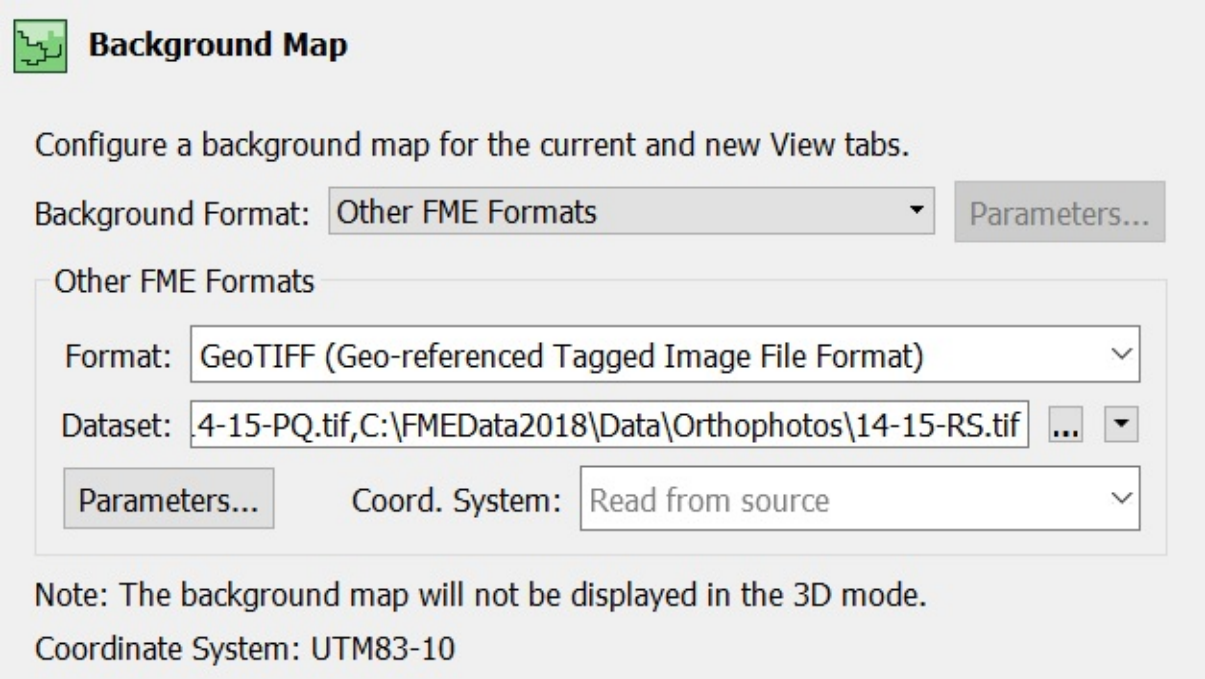


请注意，该对话框仅显示可用几何类型的符号系选项。示例是一个多边形要素。其他几何类型显示不同的选项。

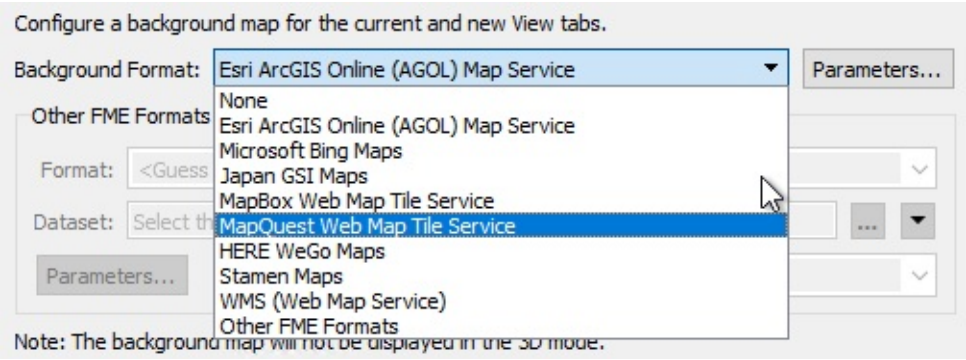
Data Inspector中的背景地图

过菜单栏上的工具> FME选项下的工具激活将地图（或其他图像）作为背景查看空间数据的功能。

背景地图对话框允许用户选择（用于任何FME支持的格式）的现有数据集作为背景，如下所示：

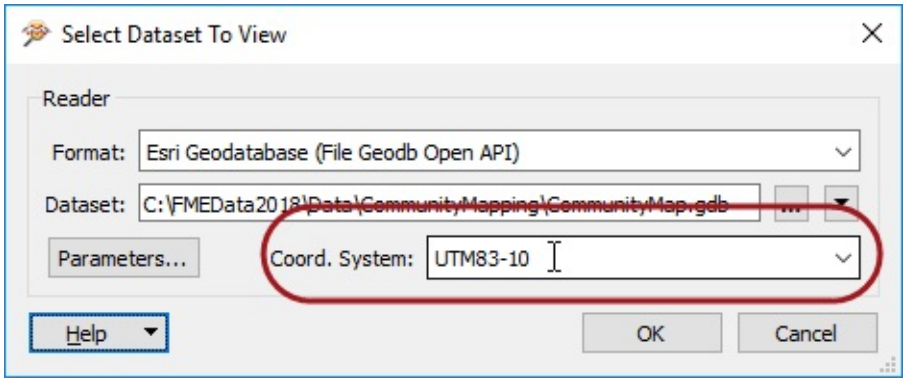


也可以使用按需提供映射的Web服务。其中一些 - 例如ArcGIS Online - 需要已有帐户：



坐标系

必须使用有效坐标系引用源数据才能使用背景地图查看它。如果坐标系未记录在数据集本身中，则可以在打开数据集时将其输入到字段中：



即使存在多个不同坐标系的源数据集，FME也可以根据背景地图显示源数据。FME通过将数据重新投影到背景地图使用的坐标系来完成此操作。因此，建议您在想要检查原始形式的数据时关闭背景地图。

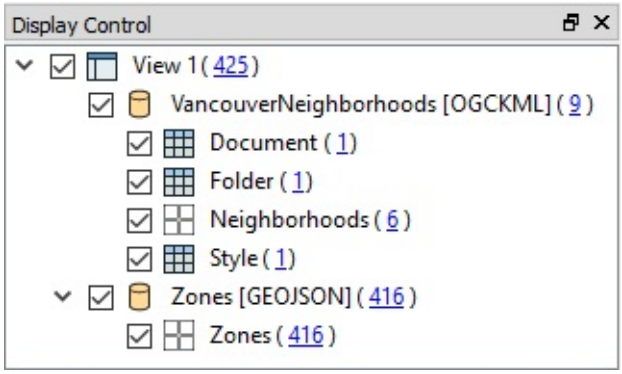
警察局长**Webb-Mapp**说...
您可以在“显示控制”窗口中调整背景地图的符号系统和显示顺序，就像对任何普通数据集一样。

练习：FME Data Inspector

练习4	FME Data Inspector
数据	分区数据（GeoJSON） 街区（Google KML）
总体目标	设置图层和数据集显示
演示	在Data Inspector中使用显示控件和背景图
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Basics-Ex4-Begin.fmw
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Basics-Ex4-Complete.fmw

在上一个练习中，我们检查了转换中的一些数据并添加了第二个数据集。现在我们可以重新排列数据以使显示更清晰。

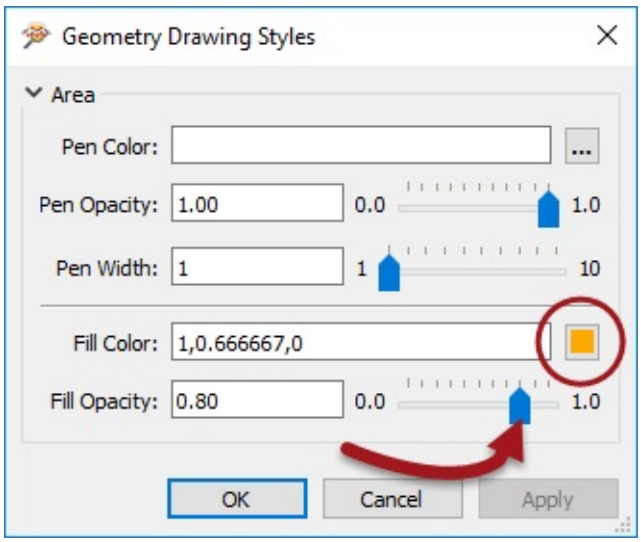
1）启动**FME Data Inspector**
继续上一练习中的FME Data Inspector。您应该同时拥有转换后的zone数据（如GeoJSON）和KML中Neighborhoods边界的数据集。“显示控制”窗口如下所示：



2）设置街区符号
显示控制窗口显示了VancouverNeighborhoods数据集中的许多不同图层。实际上，其中大多数是表格（非空间）项目。我们感兴趣的层称为Neighborhoods。

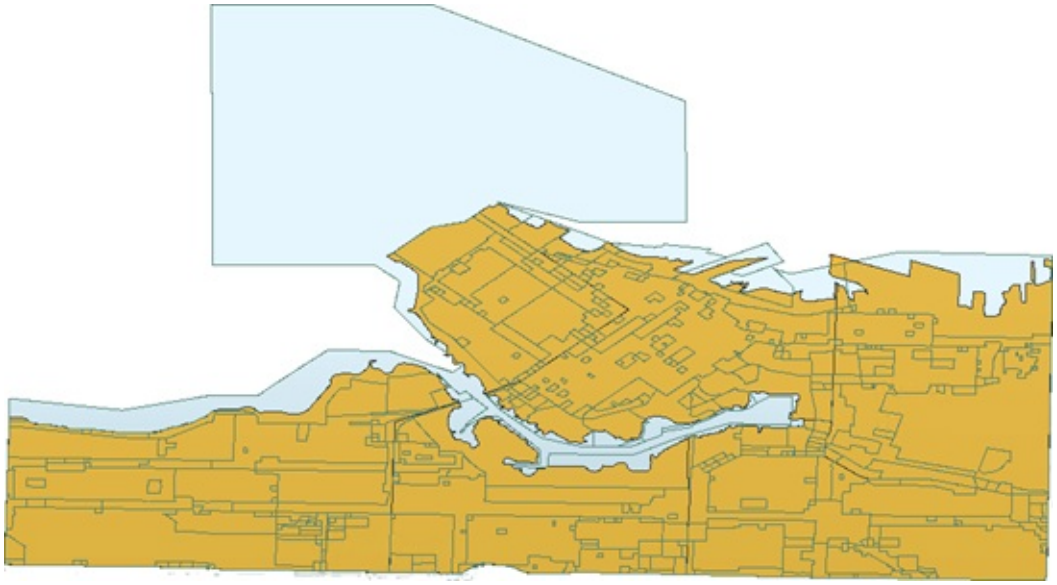
在“显示控制”窗口中单击“Neighborhoods”数据的符号系统图标：

将颜色设置为中性色调（如橙色）并将不透明度值增加到0.8：



3）设置绘制顺序
先前的更改清楚地表明区域要素位于绘图顺序中的街区之下。要解决此问题，请将“区域”数据集拖动到“显示控制”窗口中的VancouverNeighborhoods数据集上方。

同时为区域数据填充颜色设置颜色，并将不透明度值减小到0.1。主视图现在看起来像这样：



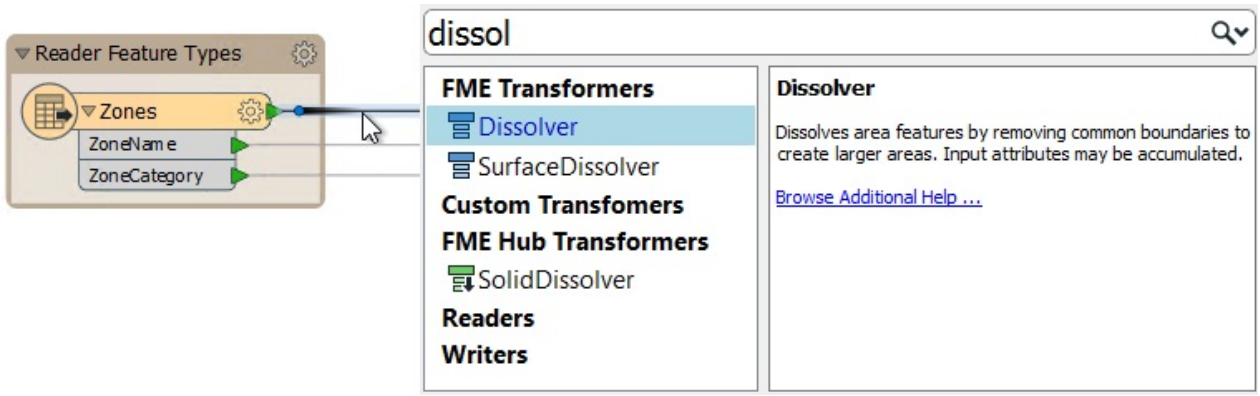
如果查询区域要素，您将看到它同时具有ZoneCategory和ZoneName属性。您可能没有意识到，但这些属性之间存在关联。每个ZoneName属于特定的ZoneCategory，其中Category：Name是1：Many关系。

让我们通过将每个ZoneCategory的所有要素合并为一个要素来阐明显示。我们可以在FME Workbench中做到这一点。

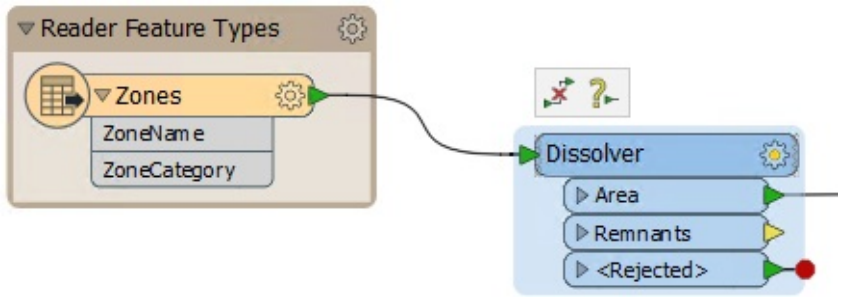
4）返回工作空间
返回FME Workbench。打开练习2中保存的工作空间或上面列出的工作空间。

我们在这里要做的是使用我们所谓的转换器。我们将在“数据转换”一章中更详细地介绍此对象。它是以某种方式转换数据的对象。

单击连接Reader Feature Type和Writer Feature Type的深色线。开始输入“Dissolver”一词：

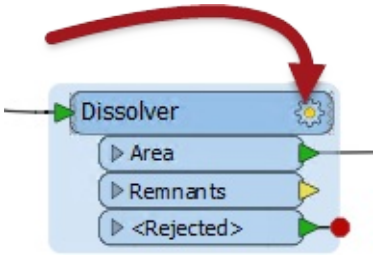


当您看到Dissolver转换器出现在列表中时，双击它以将其放入工作空间。结果将如下所示：

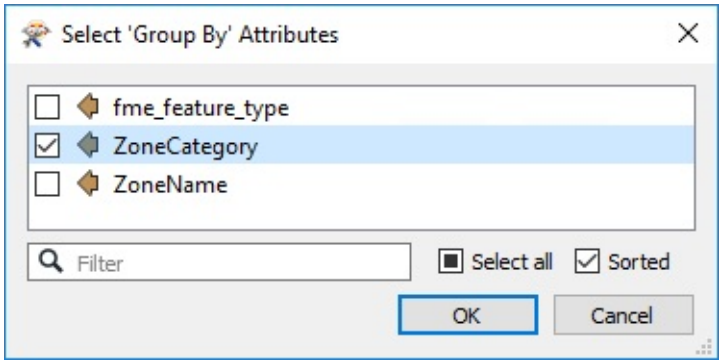


Dissolver参数将使用公共属性值合并所有要素。

5）设置**Dissolver**参数
点击Dissolver转换器上的小齿轮图标（它可能是黄色的）：



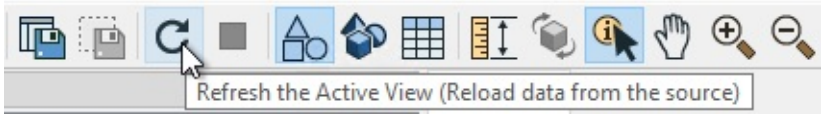
这样做将打开转换器的参数对话框。单击“分组依据”参数旁边的省略号（...）按钮。在打开的对话框中，选择ZoneCategory属性，然后单击“确定”。



再次单击“确定”关闭参数对话框。

保存工作空间并再次运行它。转换将运行，数据将被覆盖。

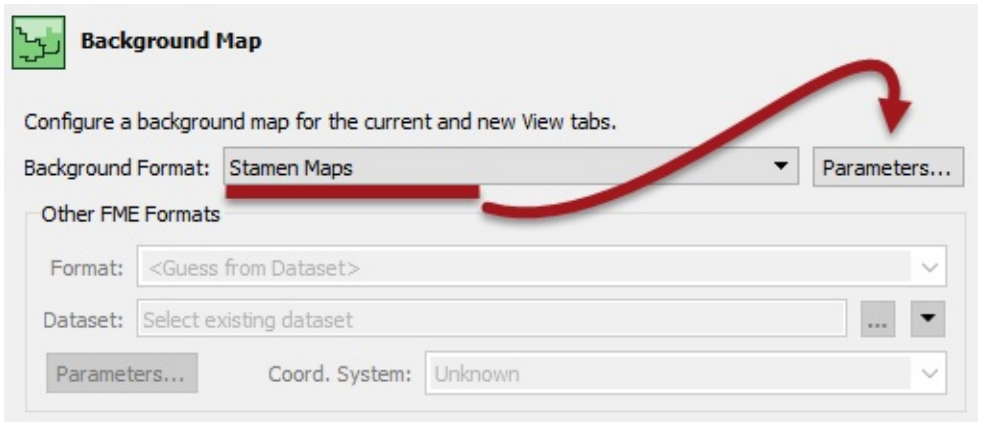
6）刷新**Data Inspector**视图
返回Data Inspector，而不是重新打开数据，让我们刷新它的视图。为此，请单击工具栏上的“刷新”按钮：



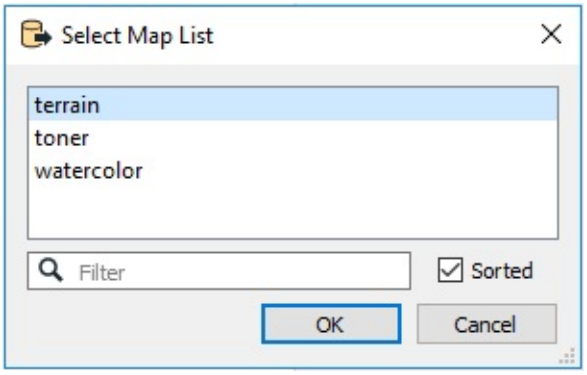
您现在会发现要素较少，因为许多要素已合并。

7）设置背景
检查数据时，将有助于获得背景图以提供位置感。FME Data Inspector能够显示来自多个不同地图服务的背景。

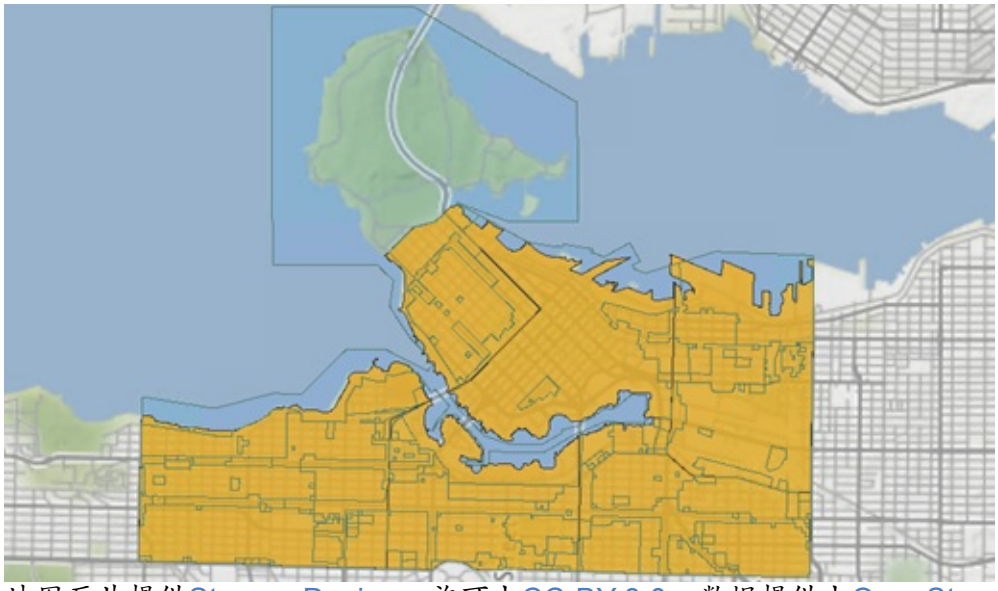
从菜单栏中选择工具> FME选项。在“背景地图”部分中，选择背景地图格式。如果您拥有特定提供商的帐户，请随时使用。否则，请选择Stamen Maps：



单击“参数”按钮。将打开地图约束（类型）对话框。单击浏览按钮并选择“terrain”：



单击确定再次单击确定以关闭这些对话框。背景地图将添加到显示中。请注意，数据被重新投影以匹配所选背景的坐标系。例如，Stamen Maps背景导致数据重新投影到Spherical Mercator，形状有明显变化：



地图瓦片提供Stamen Design，许可由CC-BY-3.0。数据提供由OpenStreetMap，许可由CC-BY-SA。

8）过滤数据
让我们尝试在Data Inspector中过滤我们的数据。

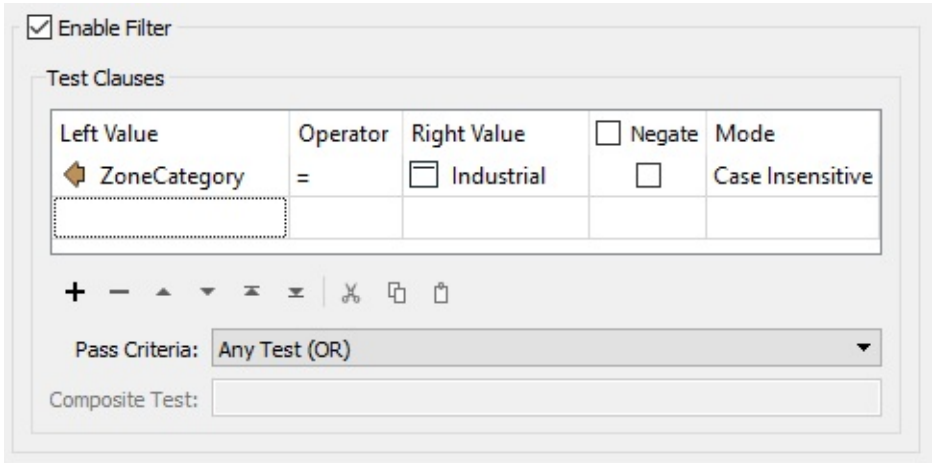
警告
使用FME 2018.0可能会导致您的数据在下一步中无法正确过滤。FME 2018.1中不存在此问题。

从菜单栏中选择“工具”>“过滤器要素”，或单击工具栏按钮以获取过滤器：

在“过滤器要素”对话框中，双击“左值”字段，单击下拉箭头，然后选择“属性值”。选择ZoneCategory作为要筛选的属性，然后单击“确定”。

对于“运算符”字段，如果尚未选择，则选择=（等于）符号。

对于Right Value字段，单击字段并键入**Industrial**（不要担心，它不区分大小写）。

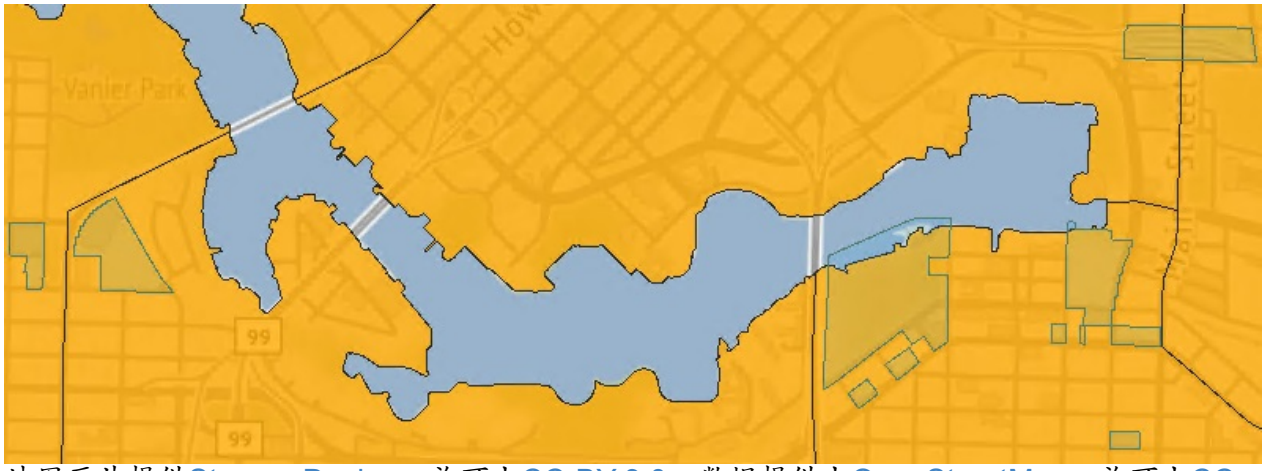


Data Inspector中的过滤应用于所有可见数据（如果您现在单击确定街区数据也将被过滤掉），因此我们还必须添加一个子句以使街区数据保留在屏幕上。

使用与以前相同的技术创建第二个测试子句。这次选择属性NeighborhoodId并选择运算符**Attribute has a value**：

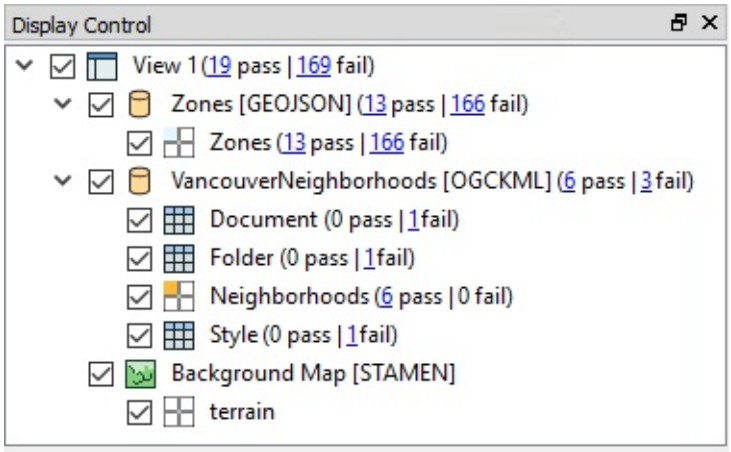
Pass Criteria参数应设置（或保留为）“One Test（OR）”。

现在单击“确定”关闭对话框。显示将被过滤以仅显示街区要素加上工业类型的区域：



地图瓦片提供Stamen Design，许可由CC-BY-3.0。数据提供由OpenStreetMap，许可由CC-BY-SA。

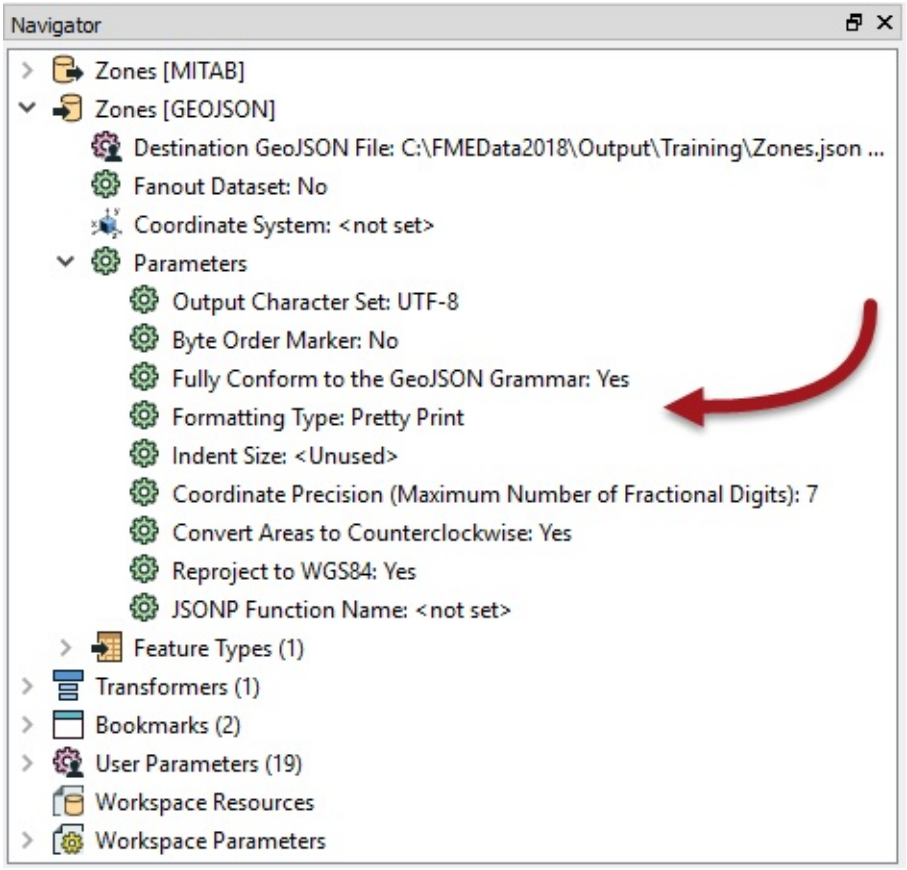
显示控制窗口还将显示过滤器的效果：



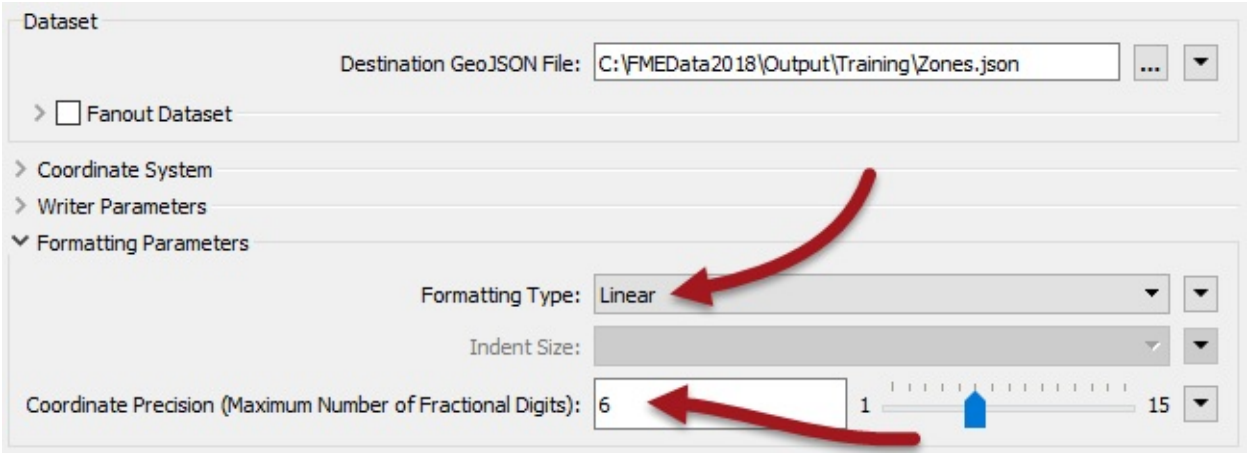
9）设置写模块参数
最后一项任务：输出GeoJSON文件用于通过Internet分发，并且您被要求使文件大小尽可能小。检查当前文件大小。它将是大约362kb。

有几个写模块参数可用于减少GeoJSON文件大小。

浏览FME Workbench中的导航器窗口，查找GeoJSON写模块参数。双击其中一个参数，例如格式化类型：



在打开的对话框中，将“格式化类型”设置为“线性”，将“坐标精度”设置为6：



重新运行工作空间。您应该会发现结果相同，但文件大小现在接近232kb。

恭喜

通过完成本练习，您已学会如何：

- 在FME Data Inspector中为要素设置符号系统
- 设置FME Data Inspector的背景地图
- 在FME Workbench中添加转换器并设置其参数
- 更新源数据时刷新数据检查器
- 在“导航”窗口中设置写模块参数

模块复习

该模块向您介绍了FME并研究了FME数据转换的基础知识。

您应该从这个模块中学到什么

以下是本次课程要学习的重点：

理论

- FME是一种转换和变换数据的工具，重点是空间数据。
- FME Workbench是一个以图形方式定义转换及其中的数据流的应用程序。转换的定义称为工作空间，可以保存到文件中供以后使用。
- 快速转换是执行FME生成的转换的技术，无需进一步编辑。
- “生成工作空间”对话框是在FME Workbench中设置快速转换的主要方法。
- 数据检查是一种在转换之前，期间和之后检查和验证数据的技术。FME Data Inspector是一种检查数据的工具。

FME技能

- 能够在FME Workbench中打开工作空间并运行它
- 能够启动FME Workbench并设置快速转换
- 能够启动FME Data Inspector，以及打开和添加数据集
- 能够导航数据集并查询其中的要素
- 能够控制Data Inspector符号系统和显示特征
- 能够在Data Inspector中设置背景地图

进一步阅读

为了进一步阅读，为什么不在我们的博客上浏览有关数据格式的文章或用**Data Inspector**标记的文章？

Q + A答案

以下是本章问题的答案。

Vector小姐说.....

ETL是.....的首字母缩写？
1.地外生命形态
2.提取，转换，加载
3.快速收费车道
4.吃，变换，爱

Vector小姐说.....

FME可以在如此多的格式之间进行无缝转换，因为它具有.....
1.一个有感觉的数据字典
2.一个复古的增压器
3.一个丰富的数据模型
4.一个独角兽的核心

Vector小姐说.....

以下哪些应用程序不属于FME Desktop？
1. FME Workbench
2. FME Integration Console
3. FME Server Console
4. FME Data Inspector

Vector小姐说.....

FME Workbench允许您以哪种方式定义数据流...
1.图形化
2.心灵感应
3.有问题的
4.通过用C ++或Java编写大量代码

Vector小姐说.....

这些中的哪一个是FME Workbench中的一个窗口？
1.数学窗口
2.地理窗口
3.英国文学窗口
4.历史窗口

Vector小姐说.....

其中哪些不是FME Workbench中的窗口布局？
1.堆叠
2.浮动
3.双层玻璃
4.选项卡式的

Vector小姐说.....

问）在生成工作空间对话框中，为什么在浏览源数据之前设置数据格式有用呢？
A）因为浏览器对话框只显示其扩展名与格式类型匹配的文件

Vector小姐说.....

其中哪一项不是设置翻译格式的方法？
1.键入格式名称
2.从下拉列表中选择格式
3.浏览格式库中的格式
4.选择具有已知文件扩展名的数据集 5.以上都不是（它们都是设置格式的有效方式）

Vector小姐说.....

哪个键是运行工作空间的快捷方式？
1. F4
2. F5
3. F5.6
4. F＃

Vector小姐说.....

在检查模式时，您要验证什么？
1.要素的颜色和线型
2. 要素 的数量
3.要素类型（层，类，表）及其属性
4.最近的咖啡店是哪里

Vector小姐说.....

启动FME Data Inspector并打开数据集。在“表视图”窗口中，右键单击记录和列标题以查看上下文菜单。以下哪项不是可用的菜单选项：
1.排序（按字母顺序或数字）
2.检查数值
3.剪切/复制/粘贴
.将所选数据另存为

练习：旅游局项目

练习**5**	旅游局项目
数据	社区测绘/食品供应商（Esri Geodatabase）
总体目标	为当地旅游局创建与GPS兼容的食品供应商数据集
演示	基础数据转换和数据检查
启动工作空间	无
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Basics-Ex5-Complete.fmw

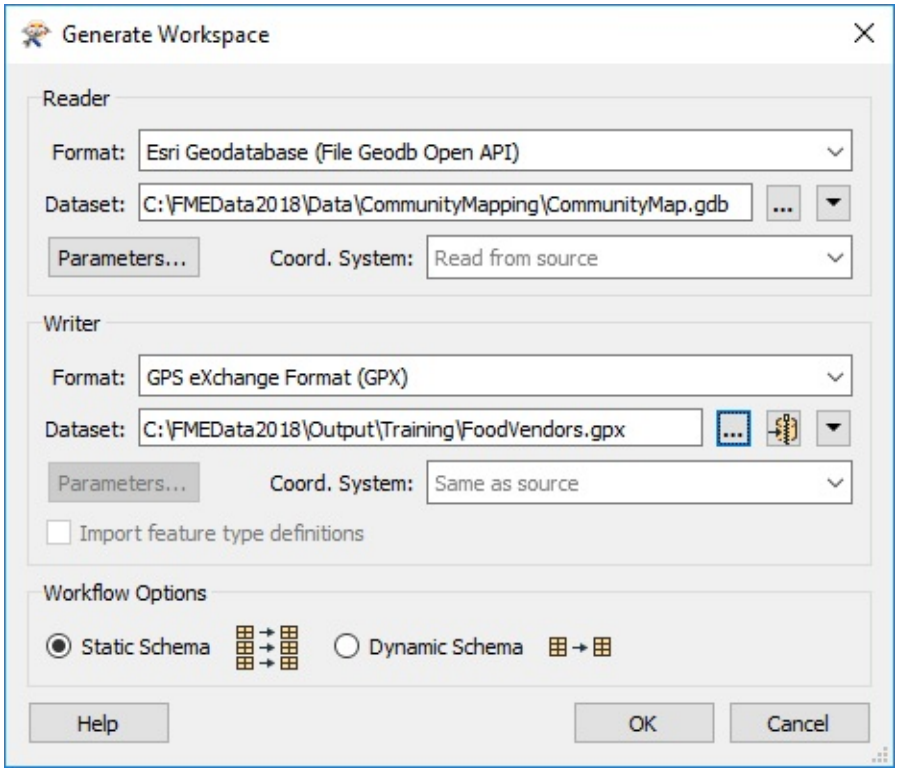
你刚刚开始你的新工作，但要求来的很快!

当地旅游局正在进行促销活动，为游客提供GPS设备，帮助他们访问该市的街头食品摊贩。您的经理想知道您是否可以使用FME生成此方案中使用的数据。

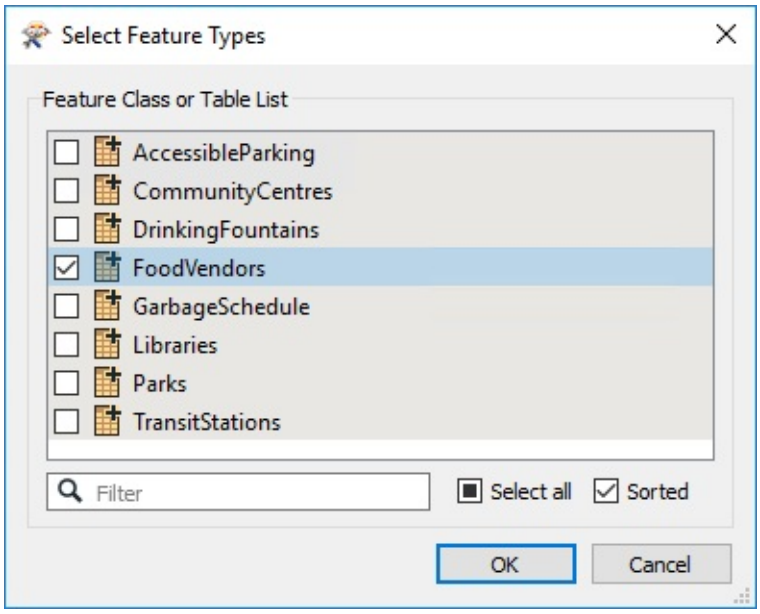
我们马上开始吧，好吗?

1）启动**FME Workbench**
启动FME Workbench。在“开始”窗口的“创建工作空间”部分中，选择“生成（工作空间）”选项。提示时使用以下参数生成转换：

读模块格式	**Esri**地理数据库（**File Geodb Open API**）
读模块数据集	C:\FMEData2018\Data\CommunityMapping\CommunityMap.gdb
写模块格式	GPS交换格式（GPX）
写模块数据集	C:\FMEData2018\Output\Training\FoodVendors.gpx

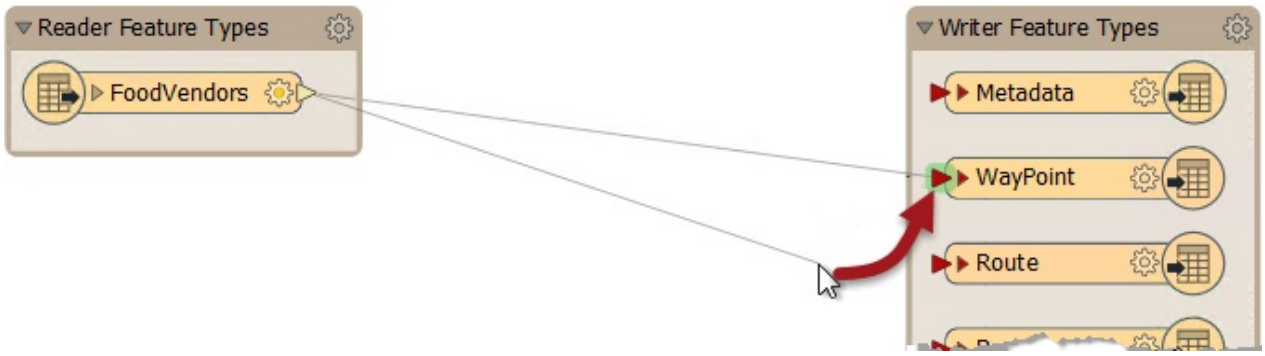


单击“确定”接受参数。当提示从源数据中使用哪些表（有几个）时，取消选择除FoodVendors之外的所有表，然后单击“确定”以创建工作空间。

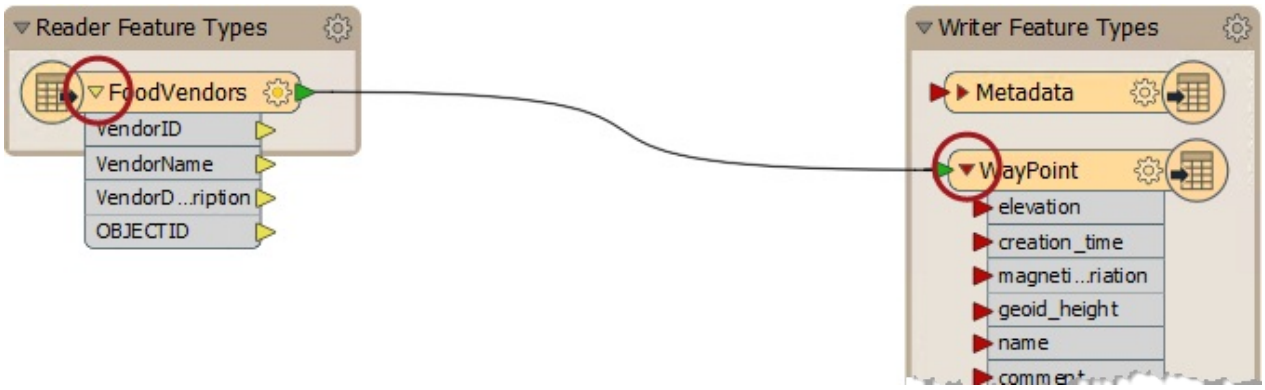


2）连接读/写模块

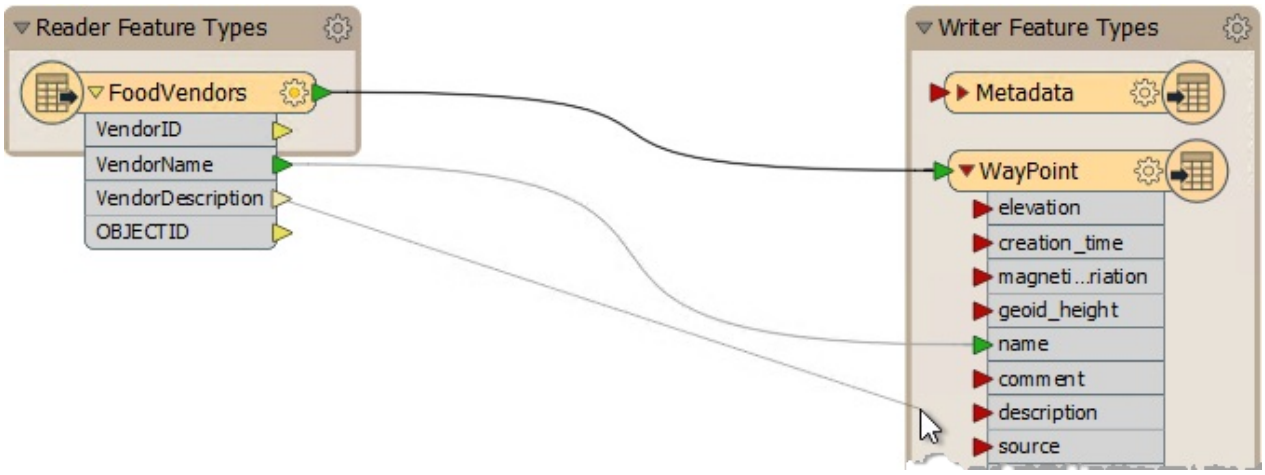
首次创建时，读模块和写模块未连接到此工作空间。通过将连接从读模块要素类型的输出端口拖动到标记为WayPoint的写模块要素类型的输入端口来连接它们：



单击两个对象上的展开按钮以显示每个对象的属性列表：



现在拖动读模块属性*VendorName*和写模块属性名之间的连接。对*VendorDescription*和描述重复此过程：



连接这样的对象的技术称为模式映射，稍后我们将对它进行更多的了解。

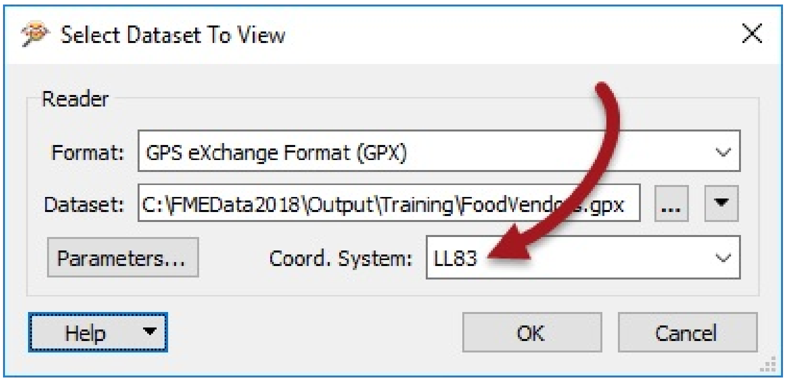
3）运行工作空间
保存工作空间，以便获得它的副本，然后按绿色运行按钮运行工作空间。工作空间运行，数据写入Garmin POI数据集：

```
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
                            Features Read Summary
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
FoodVendors                                                                   91
================================================================================
Total Features Read                                                           91
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
                           Features Written Summary
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
WayPoint                                                                      91
================================================================================
Total Features Written                                                        91
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
Translation was SUCCESSFUL with 0 warning(s) (1 feature(s) output)
FME Session Duration: 1.6 seconds. (CPU: 0.7s user, 0.3s system)
END - ProcessID: 2592, peak process memory usage: 99252 kB, current process memory usage: 89276 kB
Translation was SUCCESSFUL
```

4）检查数据

转到FMEData Inspector。从菜单栏中选择“文件”>“打开数据集”。这样做会打开标题为“选择要查看的数据集”的对话框。

设置格式类型并选择GPX数据集。但是，GPX格式不会在数据集内记录其坐标系，因此要包含背景贴图，还必须在此对话框中设置坐标系（LL83）。

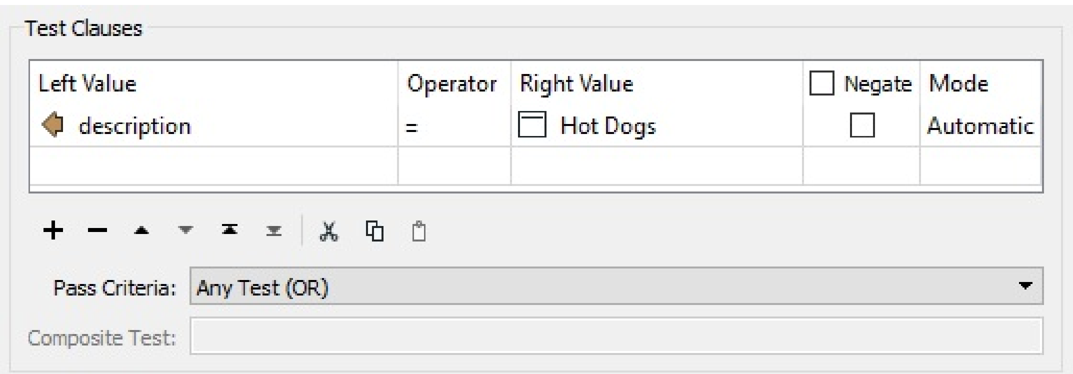


注意： 由于此格式的坐标系限制，您无法使用*Workbench*中的检查选项。那是因为数据将直接传递给*Data Inspector*，而无需设置坐标系。您必须如上所述手动打开它。

单击“确定”，将打开数据集以验证其是否正确。

5）过滤数据

所有关于食物的谈话都让你感到饥饿。一定是午餐时间。要找到一个快速午餐的地方，过滤数据以显示该市的热狗供应商：



恭喜

通过完成本练习，您证明了您知道如何：

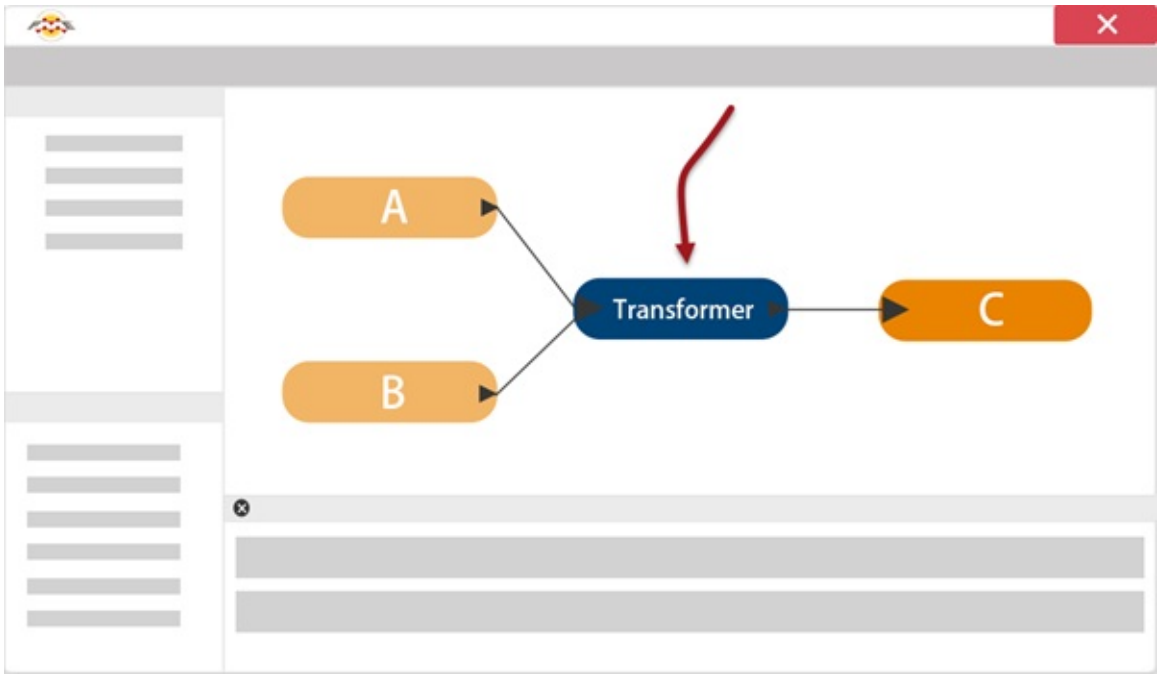
- 在FME Workbench中创建并运行工作空间
- 在FME Workbench中执行基本的“模式映射”
- 在FME Data Inspector中打开数据集

数据转换

数据转换是在数据转换过程中处理数据的能力，即使输出大于输入之和!

什么是数据转换

数据转换 是FME处理数据的能力。转换步骤发生在格式转换过程中。读取，转换数据，然后将其写入所选格式。



数据转换类型

数据转换可以细分为两种不同的类型： 结构转换 和 内容转换.

结构转型

结构转型可能更好地称为“重组”。它指的是FME在几乎无限数量的安排中将数据从源传送到目标的能力。

这包括合并数据的能力（如上图所示），划分数据，重新排序数据和定义自定义数据结构。

通过处理其模式来执行变换数据集的结构。

内容转换

内容转换可能更好地称为“修订”。它指的是改变数据集内容的能力。

操纵要素的几何或计算新的属性值是FME如何转换内容的最佳示例。

内容转换可以独立进行，也可以与结构转换同时进行。

Flibble先生说...

Flibble先生 - 经过认证的FME小丑 - 来这里招待你。这里有一个谜语，但你能解决它吗？

使用FME定义的最常见格式转换是从Esri Shapefile到哪种格式？
1. Esri Geodatabase
2. AutoCAD DXF/DWG
3. Google KML
4. Esri Shapefile

如果您在课堂上，请进行分组投票！

结构转换

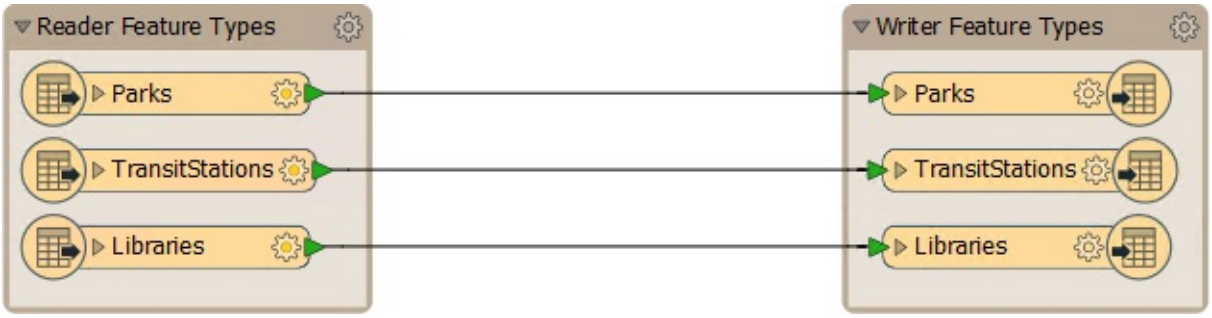
转换数据集的结构需要使用FME来操作模式。FME使用术语“模式”，但您可能将其视为数据模型。

模式概念

模式定义了一个数据集的结构。每个数据集都有其独特的模式; 它包括定义或限制其内容的图层，属性和其他规则。

模式表示

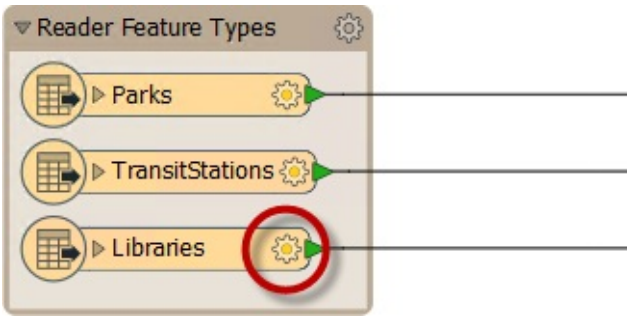
创建新工作空间时，FME会扫描源数据集。它创建了一个读模块，其图层显示在工作空间画布的左侧，而一个写模块的图层显示在工作空间画布的右侧：



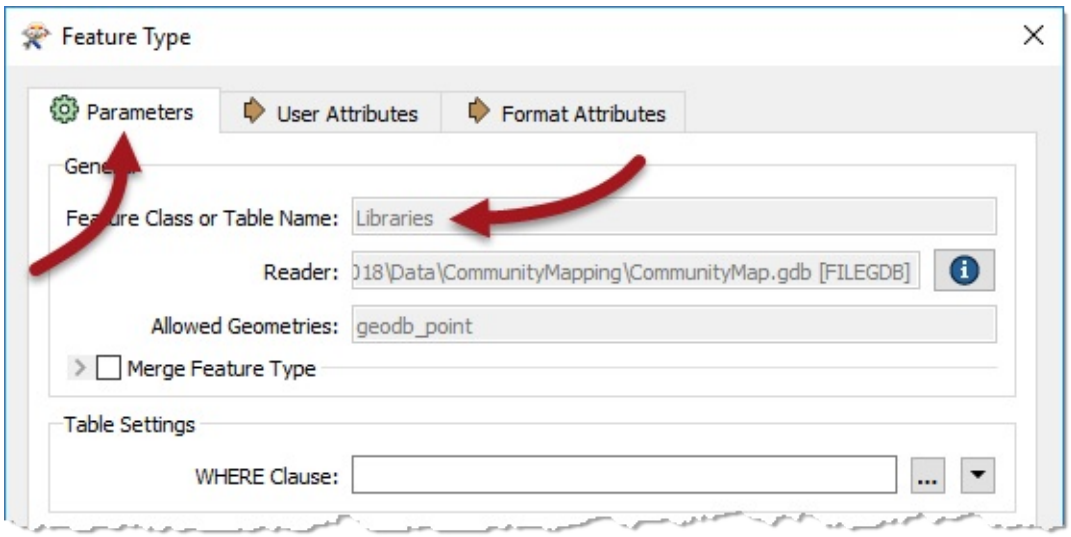
技巧
此图中的每个对象表示源数据集中的细分。在FME术语中，这些对象称为要素类型。多个要素类型表示源数据集中有多个图层。

读模块模式

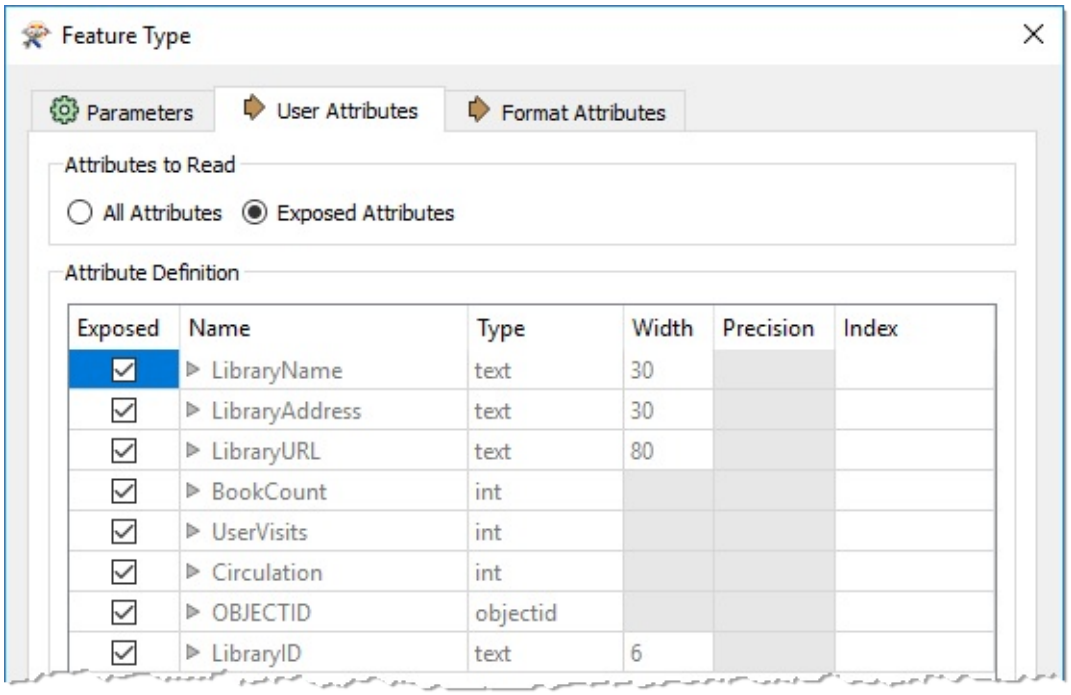
对于读模块，通过单击每个要素类型对象上的齿轮图标，可以显示有关模式的更多信息：



此“要素类型”对话框有许多选项卡。在“参数”选项卡下面是一组常规参数，例如要素类型的名称（在本例中为“库”），允许的几何类型以及父数据集的名称：



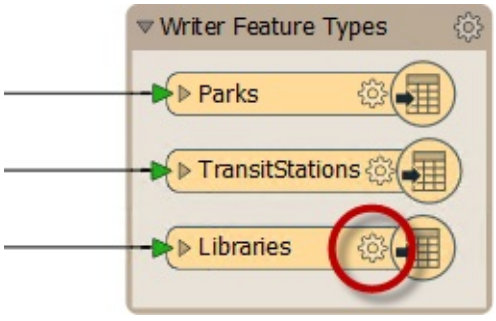
“用户属性”选项卡显示属性列表。每个属性由其名称，数据类型，宽度和小数位数定义：



每个图层都有不同的名称，也可以有一组完全不同的属性。所有这些信息都构成了读模块模式。字面意思是***“我们拥有的”***。

写模块模式

与读模块一样，每个写模块都有一组详细的模式信息，可通过打开要素类型的对话框来访问：



默认情况下，写模块模式（***“我们想要的”***）是源的镜像，因此转换的输出将是输入的副本。此要素允许用户无需进一步编辑即可从格式转换为格式（快速转换）。

如果“我们想要的”与默认模式定义不同，我们必须使用称为模式编辑的技术来更改它。

技巧
FME支持400多种格式，数据细分的方式几乎一样多。最常见的术语是图层，表，类，类别，级别或对象。 虽然这些细分的一般FME术语是要素类型，但FME Workbench中的所有对话框都使用特定格式的术语，其中正确的术语适用。

模式编辑

模式编辑是更改写模块模式以自定义输出数据结构的过程。一个很好的例子是重命名属性字段。

在编辑之后，源模式仍然代表“我们拥有的”，但目标模式现在确实代表“我们想要的”。

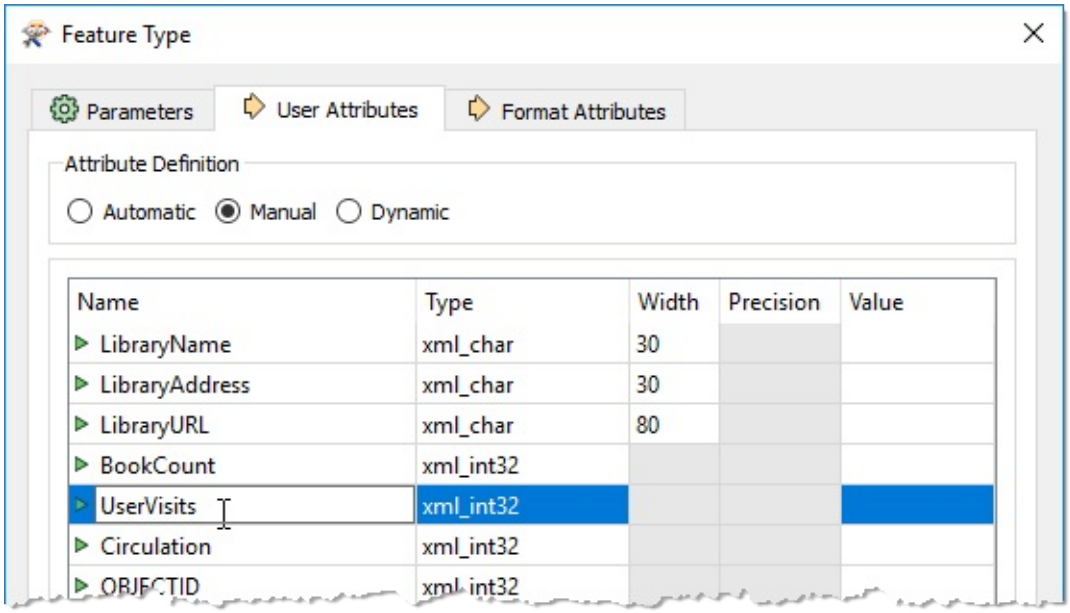
可编辑的组件

可以执行许多编辑，包括但不限于以下内容：

属性重命名

可以重命名目标模式上的属性，例如将POSTALCODE重命名为PSTLCODE。

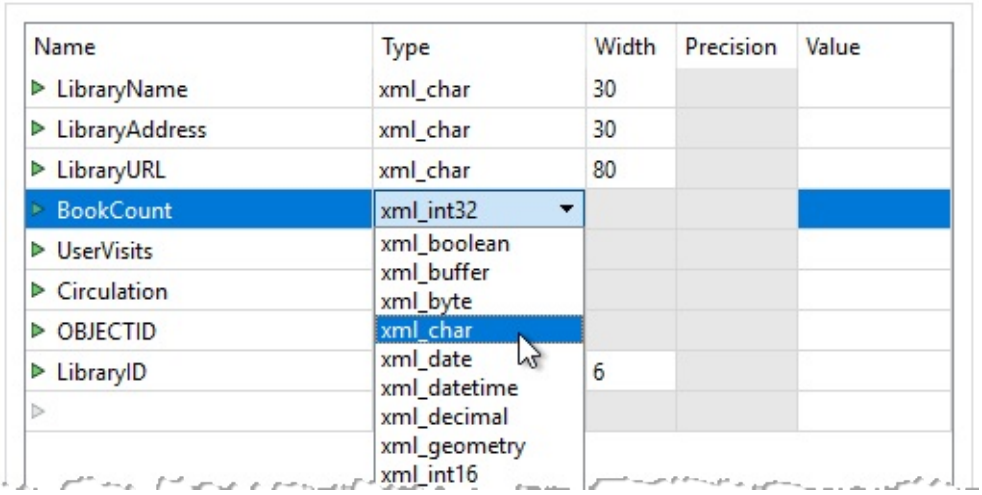
要重命名属性，请打开“功能类型”对话框，然后单击“用户属性”选项卡。单击要重命名的属性并输入新名称。



属性类型更改

写模块模式的任何属性都可以更改类型; 例如，从整数字段更改为字符字段。

要更改属性类型，请打开“用户属性”选项卡，并将“类型”字段设置为新类型：

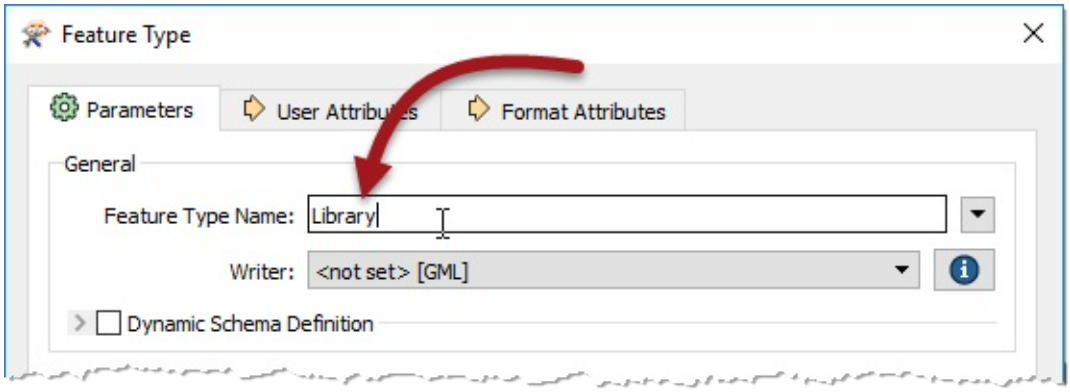


技巧

属性的“类型”列仅显示与该数据格式的允许类型匹配的值。例如，Oracle模式允许varchar或clob的属性类型。MapInfo不支持这些数据类型，因此它们永远不会出现在MapInfo模式中。上面的屏幕截图显示了GML格式的数据类型。

要素类型重命名

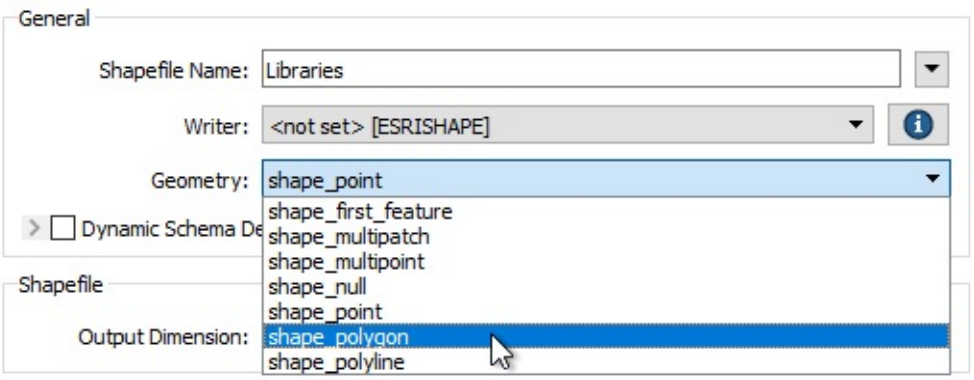
要重命名写模块要素类型，请打开“参数”选项卡，单击“要素类型名称”字段（标签可能因格式类型而异）并根据需要编辑名称：



您还可以通过在Workbench画布中单击某个要素类型并按F2键来重命名该要素类型。

几何类型更改

要更改要素类型的允许几何对象（例如，将允许的几何对象从线条更改为点），请打开“参数”选项卡并更改允许的几何对象类型：

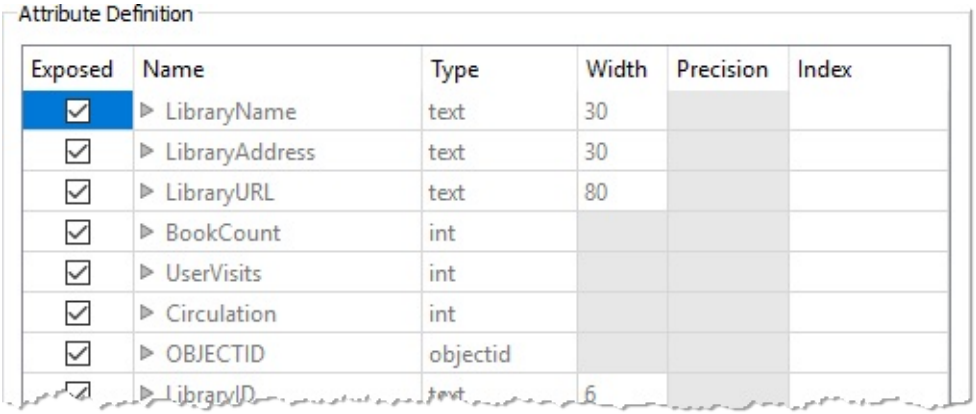


此字段仅在格式要求严格决定几何类型时可用。如果格式允许在单个要素类型中混合几何对象，则不会显示此设置。

一旦用户对写模块模式进行了所有必需的更改，工作空间就会反映“我们想要的”- 但它尚未指定读模块模式应如何连接在一起。这是下一个任务，它被称为模式映射。

技巧

您可能想知道“如果我编辑了读模块模式而不是写模块，会发生什么”？好吧，默认情况下你是不能的！读模块的模式字段是灰色的以防止这种情况，因为更改会使模式定义与源数据集不同步：有一些罕见的用例 - 但它们超出了本培训课程的范围！



模式映射

模式映射是FME数据重组的第二个关键部分。

在FME Workbench中，工作空间的一侧显示读模块模式（“我们拥有的”），另一侧显示写模块模式（“我们想要的”）。最初，创建工作空间时会自动连接这两个模式，但是在编辑时，这些连接通常会丢失。

模式映射是将读模块模式连接到写模块模式的过程，其方式是确保将正确的读模块要素发送到正确的写模块要素类型，并将正确的读模块属性发送到正确的写模块属性。

FME允许以任何所需的布置从源到目的地的映射。可以映射哪些要素类型或属性没有限制。

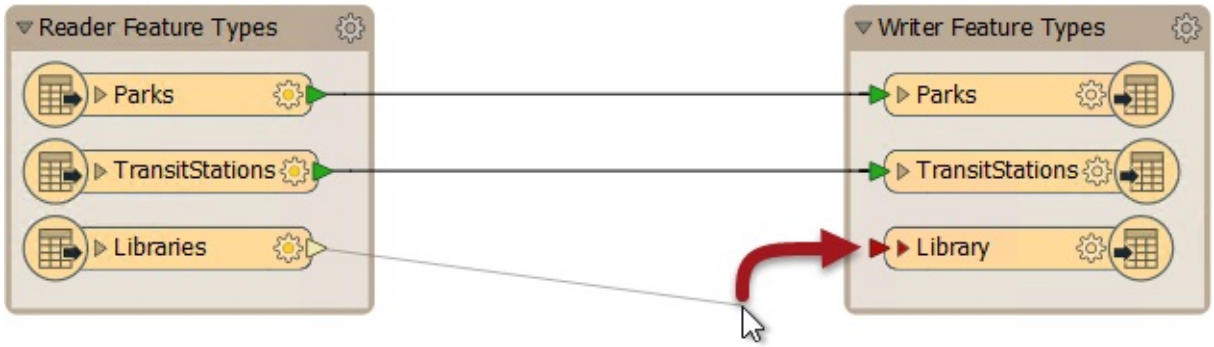
分析师女士说.....
你好。我是分析师，你在这个城市的同事之一。我认为模式编辑和模式映射是重组数据的一种手段。 一个很好的类比是衣柜里装满了衣服。当衣柜重新组织时，你扔掉你不再需要的东西，为你计划得到的新衣服预留空间，并将现有物品移至更有用的布置中。 空间数据重组也是如此：它是重组数据以使其更有用的行为。

要素映射

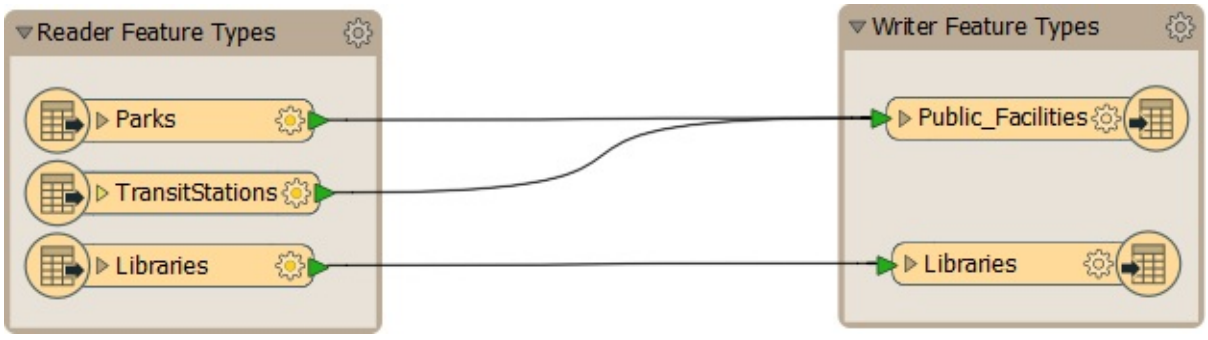
要素映射是将源要素类型连接到目标要素类型的过程。

通过单击读模块要素类型的输出端口，将箭头拖动到写模块要素类型的输入端口，然后释放鼠标按钮来执行要素映射。粗黑线表示连接。

这里，通过将箭头从源拖动到目标，创建从源到目标要素类型的连接线：



允许合并和拆分数据：



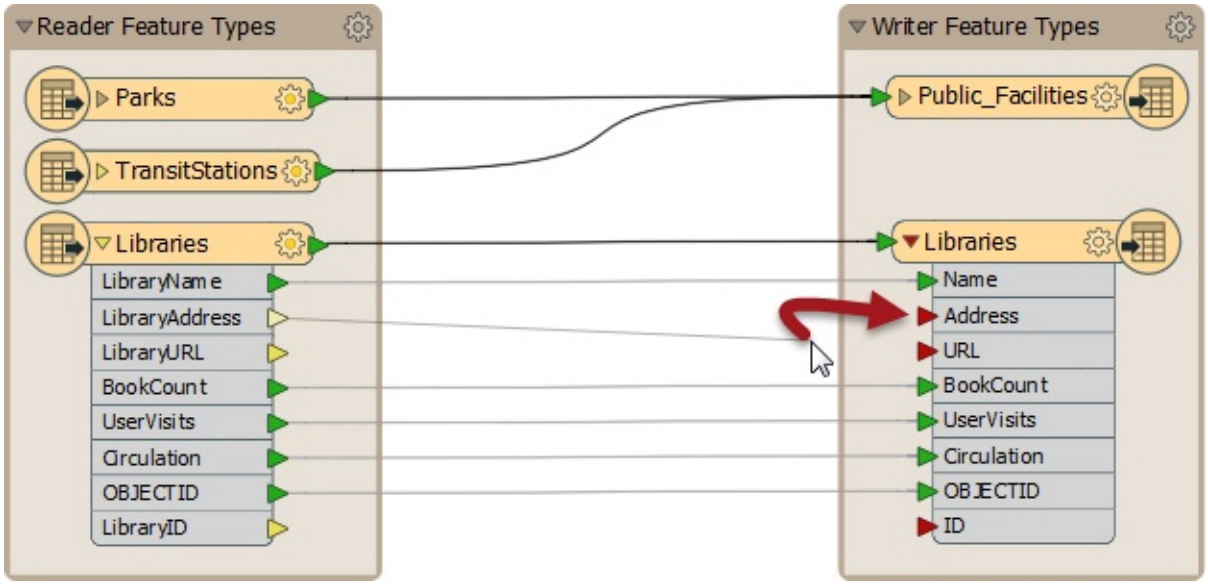
在这里，用户要求Parks，TransitStation和Libraries都在输出中的单个图层上，因此将这三种读模块要素类型合并为一个写模块要素类型（PublicFacilities）。

属性映射

属性映射是将读模块属性连接到写模块属性的过程。

通过单击读模块属性的输出端口，将箭头拖动到写模块属性的输入端口，然后释放鼠标按钮来执行属性映射。属性连接显示为更细的灰线。

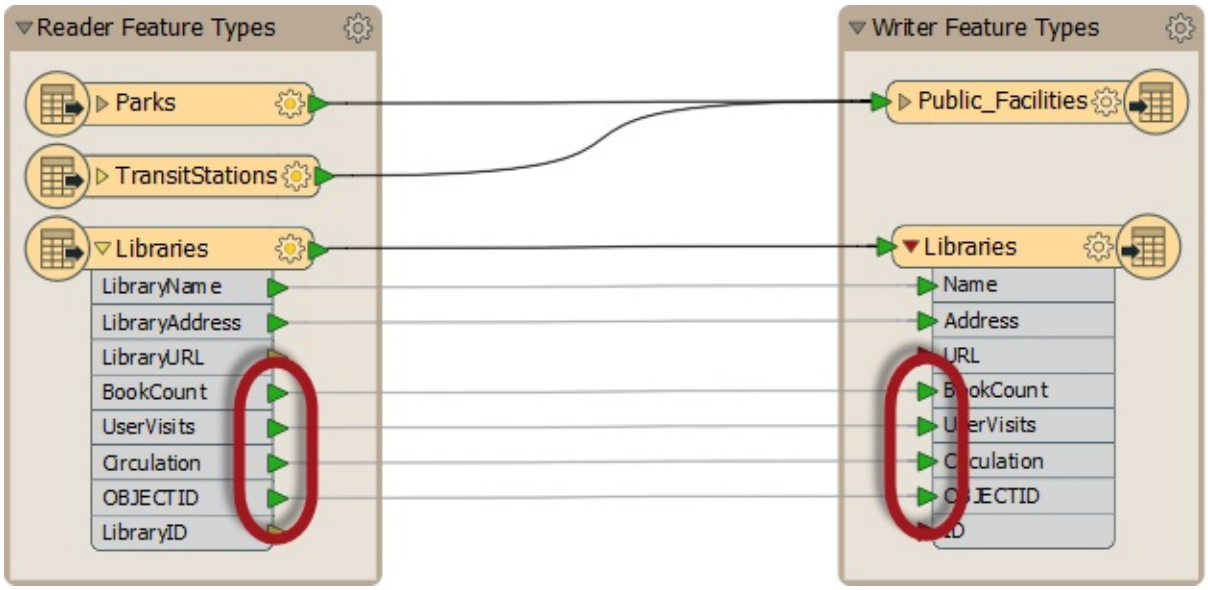
这里已经进行了要素映射，并且正在进行属性连接：



请注意绿色，黄色和红色颜色编码，指示连接了哪些属性。

绿色端口表示已连接的属性。黄色端口表示未与写模块连接的读模块属性。红色端口表示未与读模块连接的写模块属性。

读模块和写模块要素类型中具有相同名称的属性会自动连接，即使连接线可能不可见; 端口颜色是关键：



警告
名称区分大小写，因此ROADS与roads，Roads或rOADS不同。 知道您是否依赖自动属性连接，这一点很重要！

练习：地面维护项目 - 模式编辑

练习**1**	地面维护项目 - 模式编辑
数据	城市公园（MapInfo TAB）
总体目标	计算城市中每个公园的大小和平均大小，以用于草地切割，树篱修剪等的地面维护估算。
演示	结构转换，模式编辑
启动工作空间	无
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex1-Complete.fmw

您刚刚在当地城市的GIS部门担任技术分析师。

负责维护公园和其他草地区域的团队需要了解每个公园的面积和设施，以便计划下一年的预算。您已被分配到此项目，并将使用FME提供此信息的数据集。

此示例的第一步是重命名现有属性并创建新属性，以便为以后的面积计算做准备。

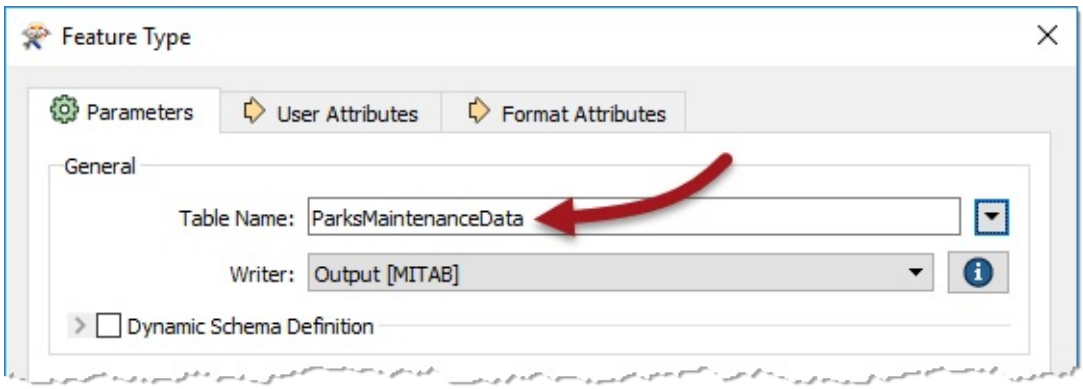
1）启动**Workbench**
使用“生成工作空间”对话框，使用以下参数创建工作空间：

读模块格式	**MapInfo TAB**（**MITAB**）
读模块数据集	C:\FMEData2018\Data\Parks\Parks.tab
写模块格式	MapInfo TAB（MITAB）
作家数据集	C:\FMEData2018\Output\Training

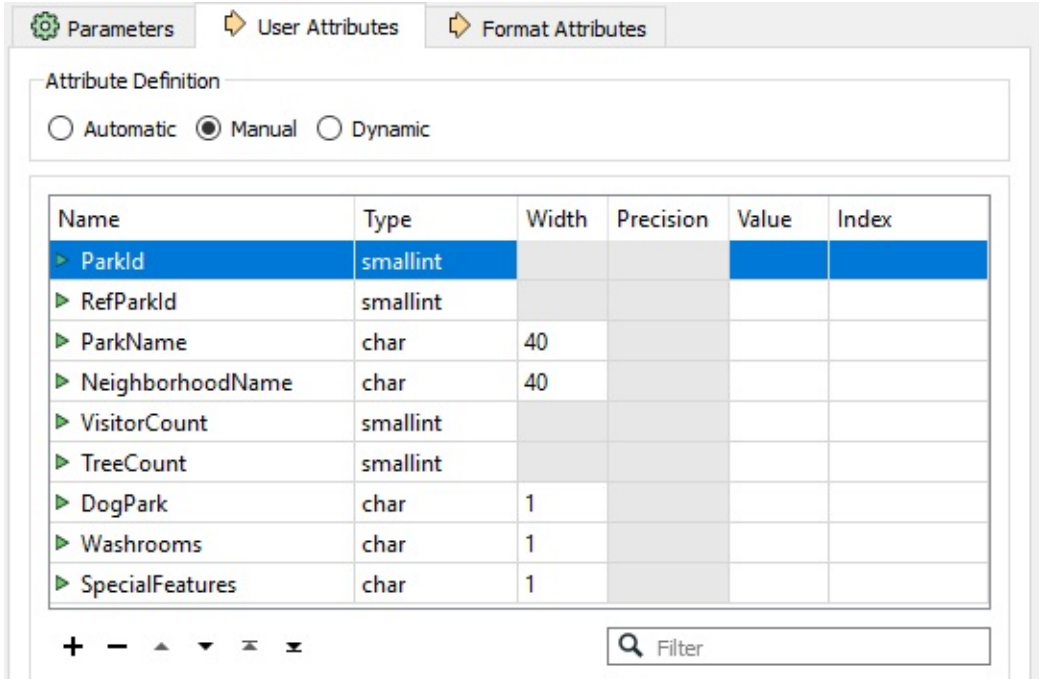
是! 在这里，我们回写我们正在读取的相同格式的数据！您可以单击“生成工作空间”对话框中的“参数”按钮以检查读模块/写模块参数，但在本练习中不需要更改它们。请注意，写模块只需要一个文件夹而不是特定的文件名。

2）重命名要素类
FME创建一个工作空间，其中目标模式与源匹配。但是，数据的最终用户已请求清除部分模式。

检查写模块要素类参数。单击标记为表名的字段（请记住此标签是特定于格式的，并在MapInfo中处理“表”）并将名称从Park更改为ParksMaintenanceData：



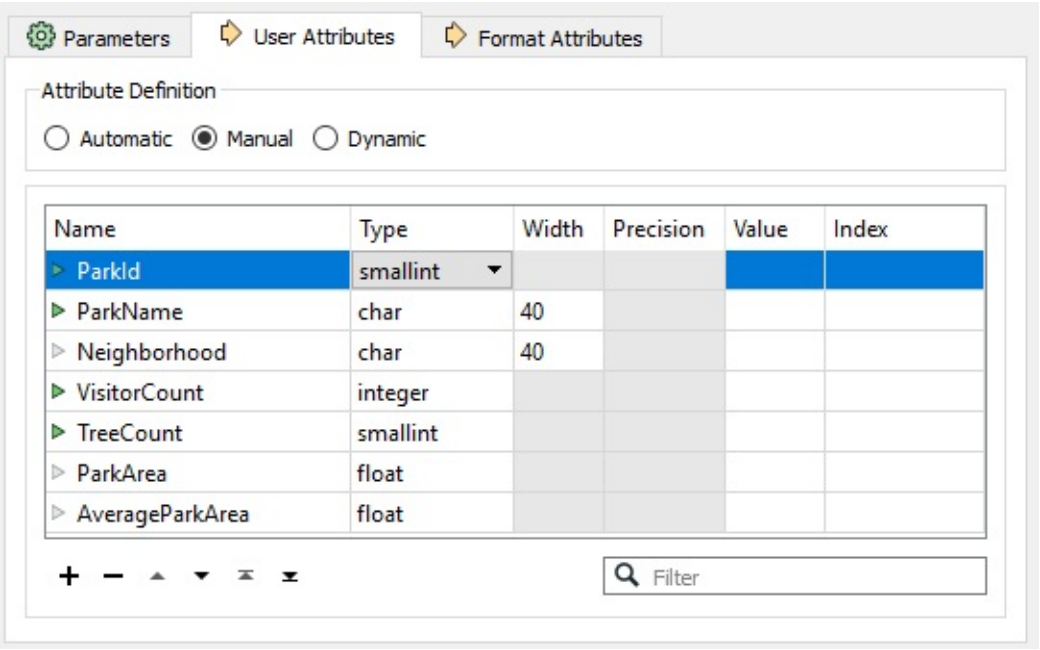
3）更新属性
现在检查用户属性。它们看起来像这样：



必须清理这些内容，以便删除不必要的信息。需要更新其他属性，并添加一些额外的属性来存储计算结果。因此，请执行以下操作：

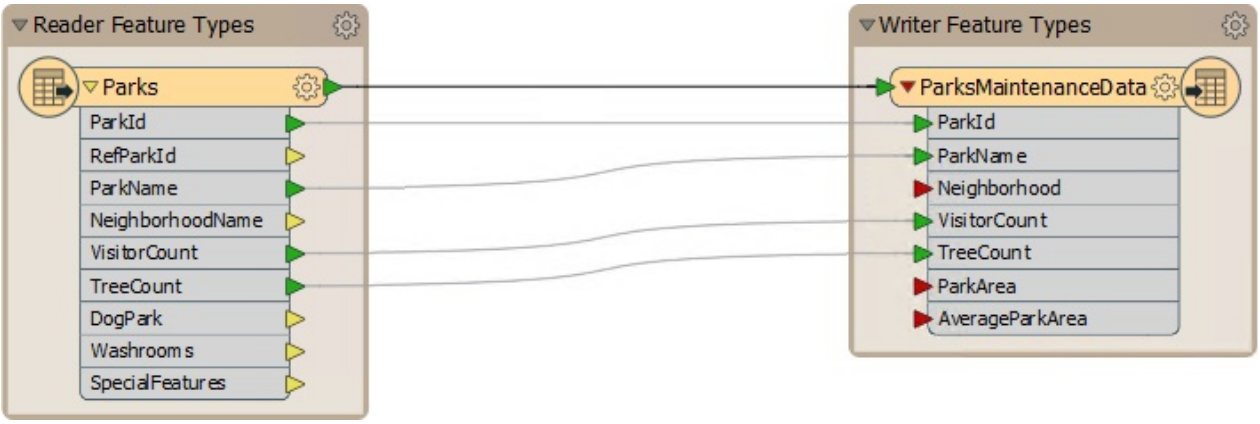
删除属性	**RefParkID**	
删除属性	DogPark	
删除属性	Washrooms	
删除属性	SpecialFeatures	
重命名属性	从：NeighborhoodName	到：Neighborhood
更改类型（VisitorCount）	从：短整数	到：整数
添加属性	ParkArea	类型：浮点数
添加属性	AverageParkArea	类型：浮点数

...然后单击参数编辑器“应用”按钮。现在，属性列表应如下所示：



现在，当运行工作空间时，输出将命名为ParksMaintenanceData.tab，并将包含更新的属性模式。

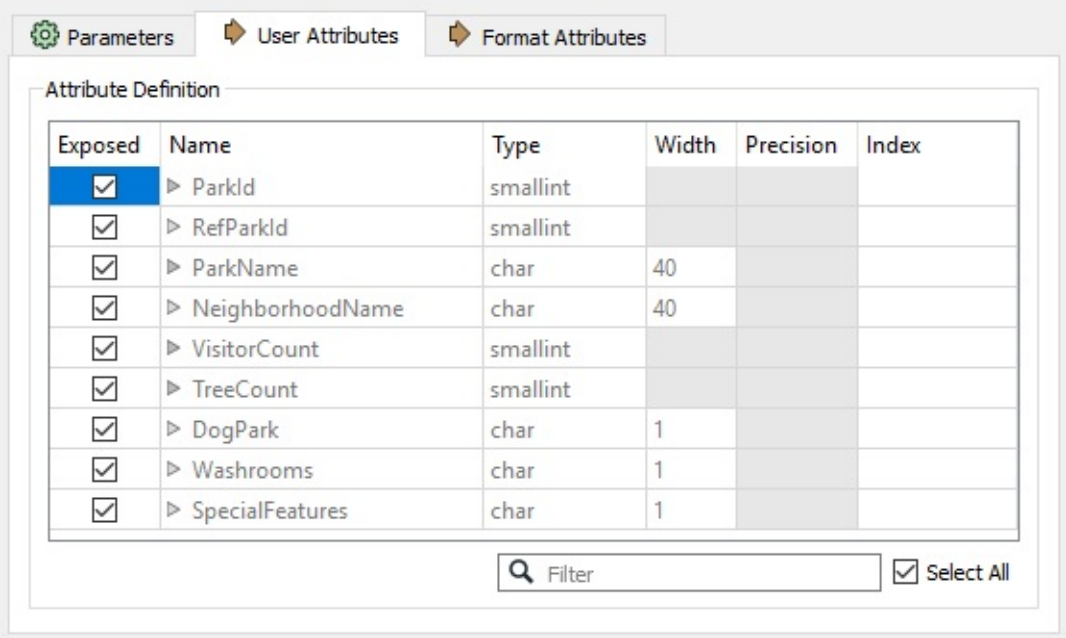
4）取消暴露源属性
工作空间现在看起来像这样：



技巧
如果要素类型名称或属性名称太大而无法以默认大小正确显示，则可以调整画布上的对象（如上面的屏幕截图所示）。要素类型周围的棕色标记称为书签。它们也可以调整大小以更好地适应其内容。

请注意，有几个源属性不会在工作空间中使用或发送到输出。我们可以通过隐藏它们来整理工作空间。

检查读模块要素类型参数上的“用户属性”选项卡。它看起来像这样：



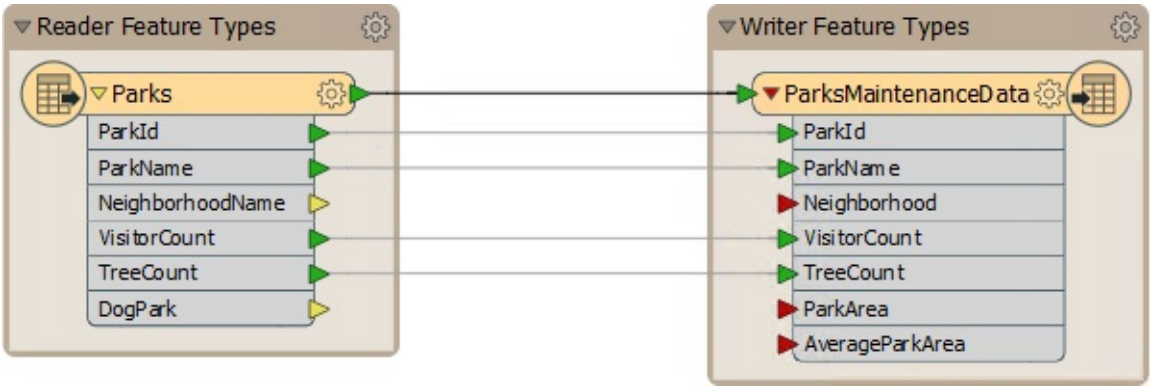
取消选中我们不需要的以下属性的复选框：

- RefParkID
- Washrooms
- SpecialFeatures

这是我们删除的属性列表，除了DogParks，我们将在转换中使用它们。

单击“应用/确定”以确认更改。

5）保存工作空间
保存工作空间 - 将在其他示例中完成。它现在应该是这样的：



警察局长**Webb-Mapp**说.....
一些写模块属性（ParkArea和AverageParkArea）有红色连接箭头，因为没有任何内容可以映射到它们，而另一个（Neighborhood）只是未连接。 没关系。如果你保证以后联系他们，我会谨慎地告诉你。您仍然可以运行此工作空间，以查看输出的内容。

恭喜
通过完成本练习，您已学会如何： • 编辑写模块模式的属性 • 编辑写模块模式的输出层名称 • 隐藏读模块模式上的属性

用转换器进行转换

除了模式编辑和模式映射之外，还可以使用称为转换器的对象进行转换。

什么是转换器？

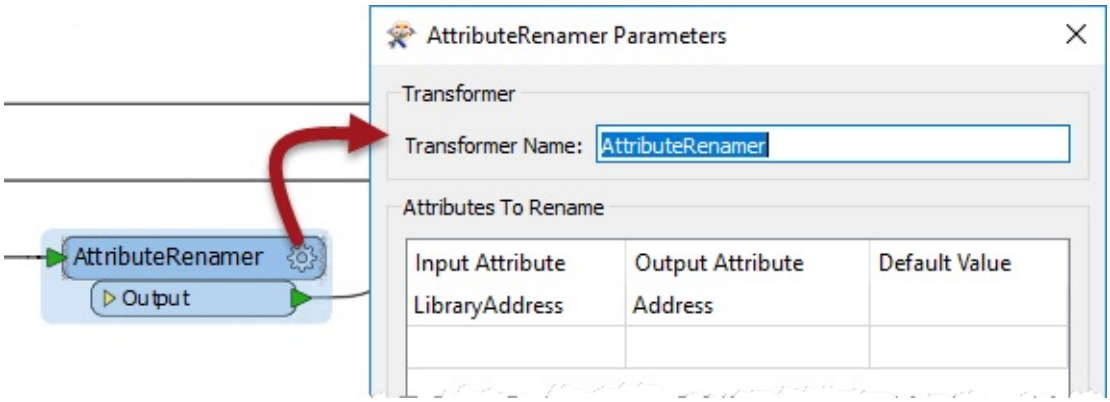
顾名思义，转换器是一个FME Workbench对象，用于执行要素转换。有许多FME转换器，每个转换器执行许多不同的操作。

转换器连接在读模块和写模块要素类型之间，以便数据从读模块，转换过程流向写模块。

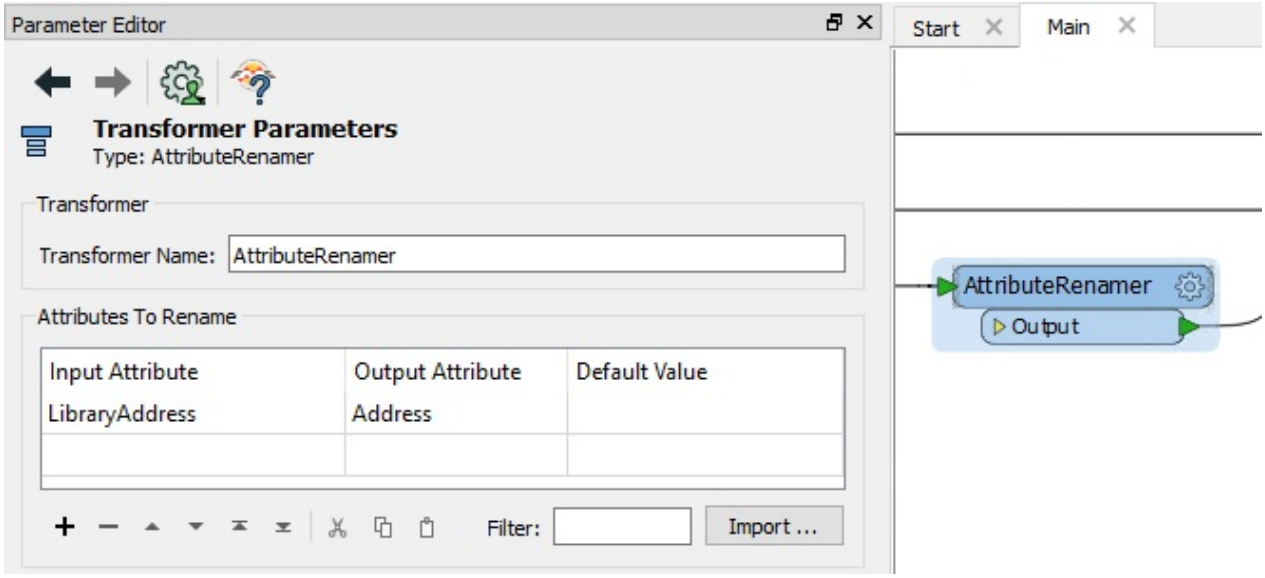
转换器通常在画布窗口中显示为矩形浅蓝色物体。

转换器参数

每个转换器可能有许多参数（设置）。单击齿轮图标可以访问参数（如要素类型）：



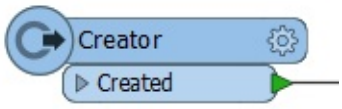
或者，如果参数编辑器窗口打开，只需单击转换器（或任何其他画布对象）即可在此处找到参数：



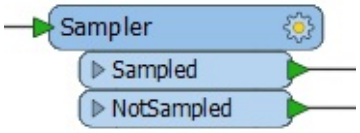
颜色编码的参数按钮

转换器上的参数按钮采用颜色编码，以反映设置的状态。

蓝色参数按钮表示已根据需要检查和修改转换器参数，并且转换器已准备就绪。

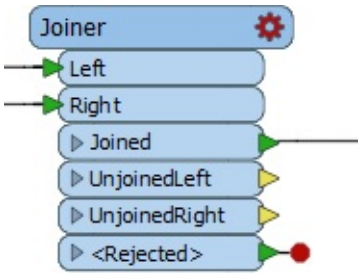


黄色参数按钮表示尚未检查默认参数。转换器可以在这种状态下使用，但结果可能无法预测。



技巧
如果参数编辑器窗口打开，那么很少会看到黄色图标，因为转换器的参数会自动打开并假定要进行检查。如果该窗口打开，您应该确保检查它以确保您不会错过设置所需的参数。

红色参数按钮表示至少有一个参数，FME无法提供默认值。在使用转换器之前，必须为参数提供一个值。

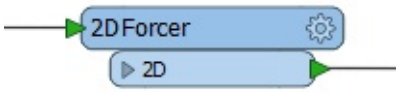


副驾驶**Transformer**说.....
大家早上好，我是副驾驶Transformer，我想欢迎你参加今天的培训。 在尝试起飞之前，请务必检查您的参数。如果其中有任何带有红旗标志的转换器，您的工作空间将无法飞行！

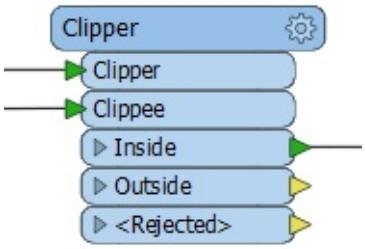
转换器端口

转换器不仅没有单个输入和输出，还可以有多个输入端口，多个输出端口或两者。

此2DForcer转换器具有单个输入和输出端口。

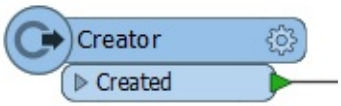


此Clipper具有多个输入和输出端口。请注意，并非所有这些都是 - 或需要 - 连接。



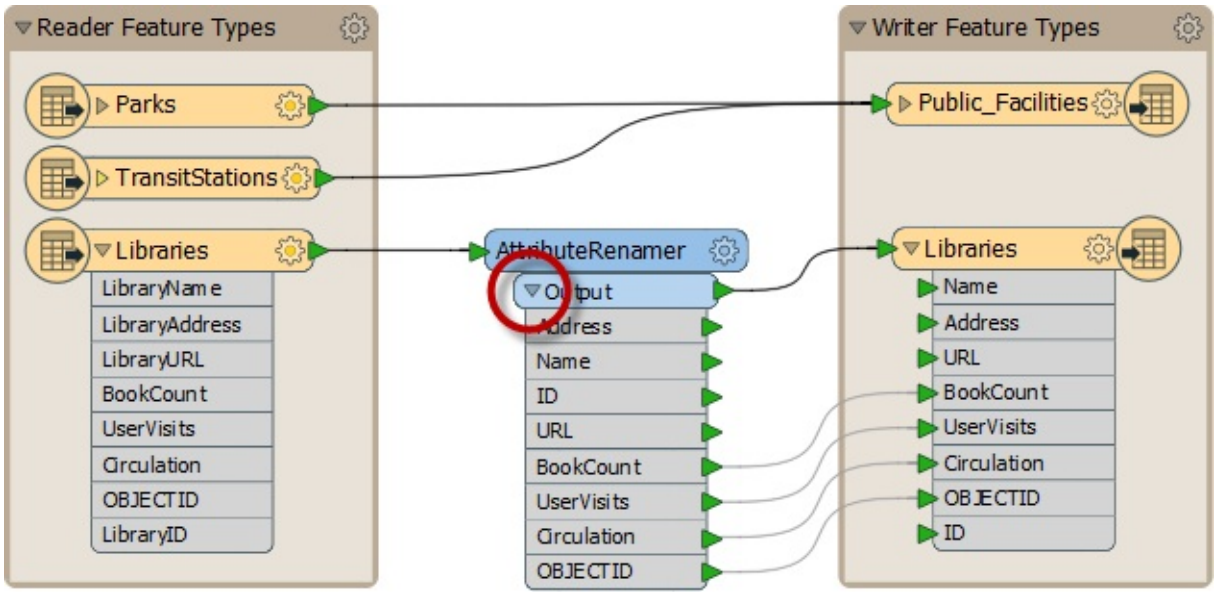
这个Inspector只有一个输入端口.....

...而这个Creator只有一个输出端口！



转换器属性

单击转换器输出端口的下拉箭头以查看退出转换器的所有属性。此列表包括转换器中应用的所有更改。



此要素可以显示在转换器中创建的，丢失的或以其他方式转换的属性。

练习：地面维护项目 - 结构转型

练习**2**	地面维护项目 - 结构转换
数据	城市公园（MapInfo TAB）
总体目标	计算城市中每个公园的大小和平均大小，以用于草地切割，树篱修剪等的地面维护估算。
演示	用转换器进行结构转换
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex2-Begin.fmw
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex2-Complete.fmw C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex2-Complete-Advanced.fmw

让我们继续您的地面维护项目。

如果您忘记了，负责维护公园和其他草地区域的团队需要了解每个公园的面积和设施，以计划下一年的预算。

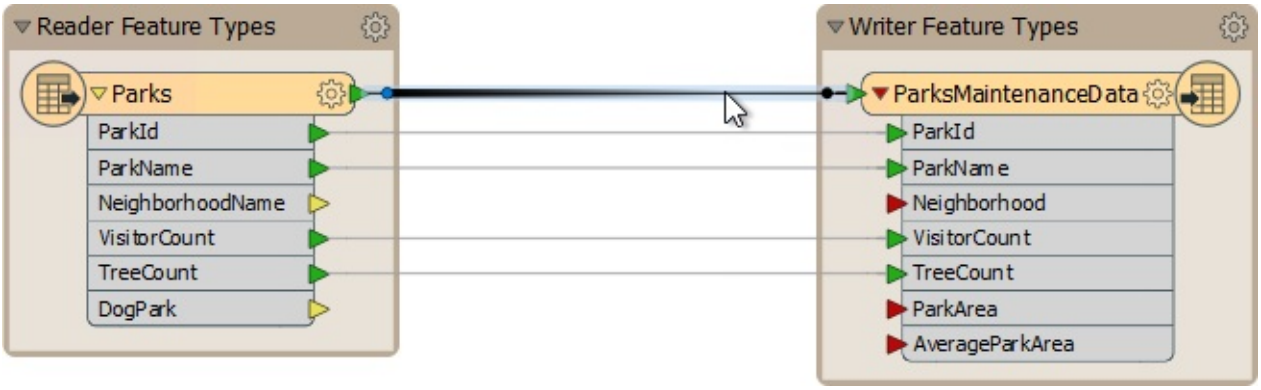
在项目的这一部分，我们将从源数据中过滤出狗公园（因为它们具有不同的维护成本）并将它们写入日志窗口。我们还将处理重命名的属性NeighborhoodName。

1）启动**FME Workbench**
启动FME Workbench并从上一个练习中打开工作空间。或者，您可以打开C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex2-Begin.fmw。

2）添加转换器
让我们先处理源属性NeighborhoodName，它在写模块上重命名为Neighborhood但尚未连接。

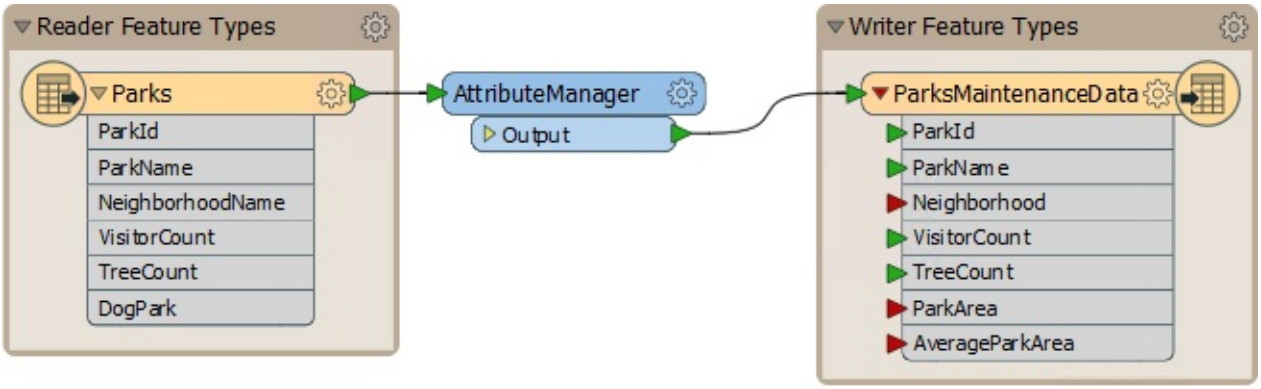
我们可以通过简单地绘制连接来做到这一点，但通常最好使用转换器。我们可以使用两种转换器。一个称为AttributeRenamer，另一个 - 我们将在这里使用 - 是AttributeManager。

因此，单击从读模块到写模块要素类型的要素连接：



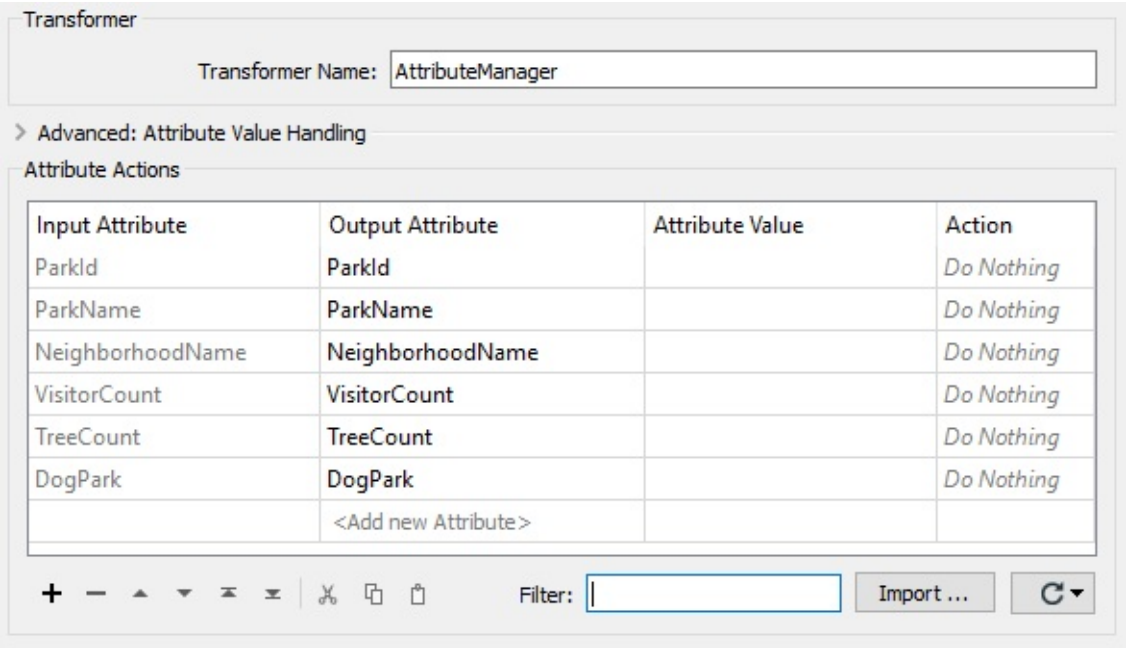
开始输入短语“AttributeManager”。这种方法是我们如何将转换器放置在工作空间中。键入时，FME会搜索匹配的转换器。当列表足够短，您可以看到AttributeManager时，从对话框中选择它（双击它）：

这样做将放置一个AttributeManager转换器：



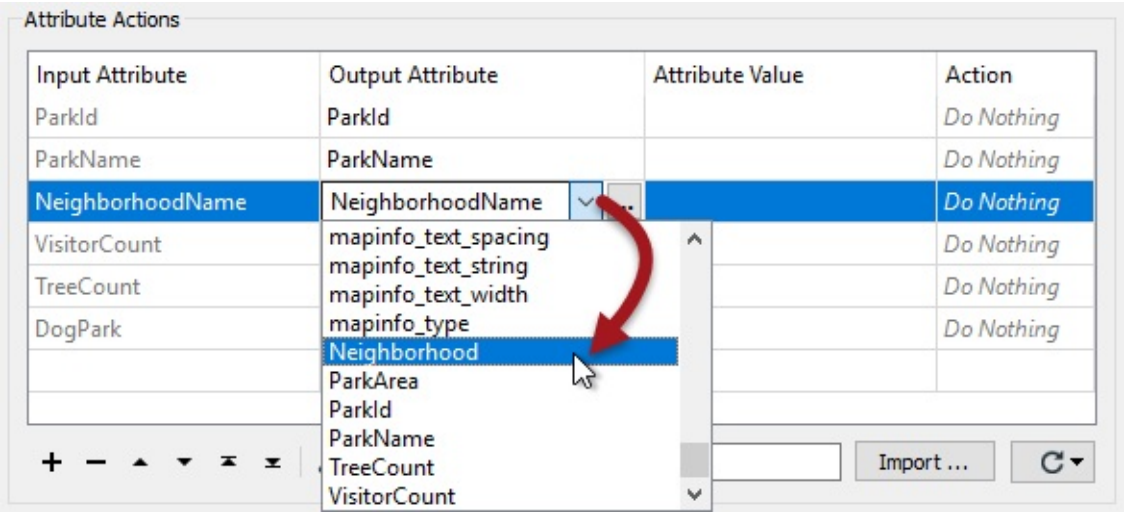
技巧
关于添加转换器的一个很好的建议，请参阅我们有史以来十大FME技巧列表中的#5！

3）设置参数
查看AttributeManager参数（其对话框或参数编辑器窗口）。它看起来像这样：



请注意，它所连接的流上的所有属性都将自动显示在对话框中。

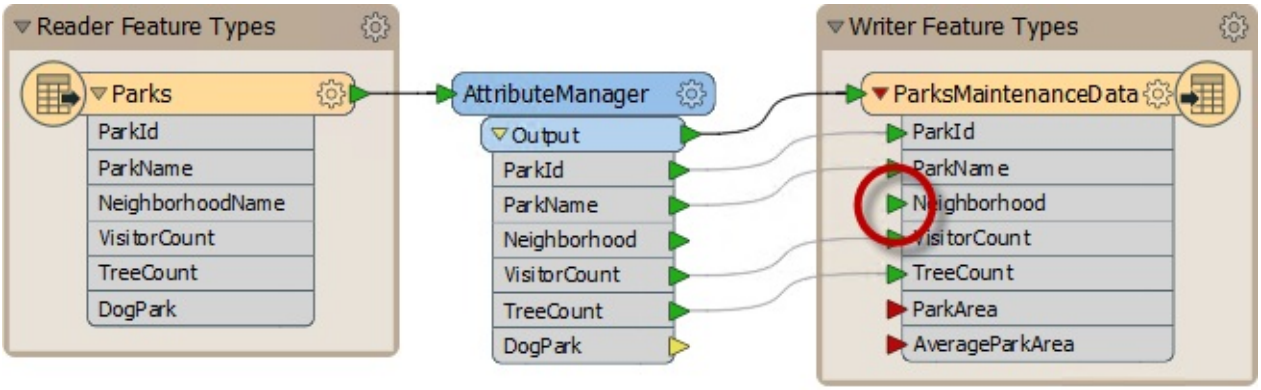
如果“输入属性”字段设置为“ NeighborhoodName”，请单击“输出属性”字段。单击下拉列表的按钮，然后选择Neighborhood作为要使用的新属性名称：



作为响应，Action字段将更改为读取*Rename*。

技巧
街区（NeighborhoodName）只出现在列表中，因为它已存在于写模块模式中。如果我们在编辑写模块模式之前完成了此步骤，则必须在此对话框中手动输入新的属性名称。

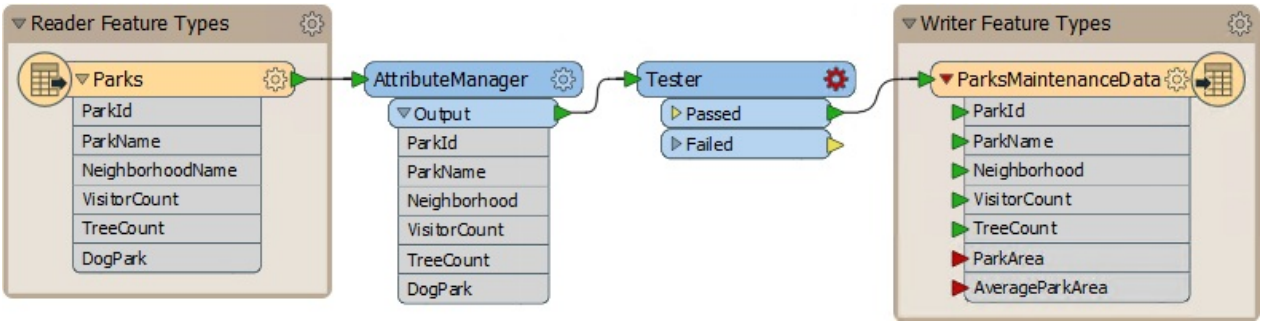
单击“确定”关闭对话框。现在，在Workbench画布窗口中，您将看到Neighborhood属性标有绿色箭头，以确认正在为该属性提供值。



4）添加转换器
现在我们应该从数据中删除狗公园，因为它们有自己的一组成本。

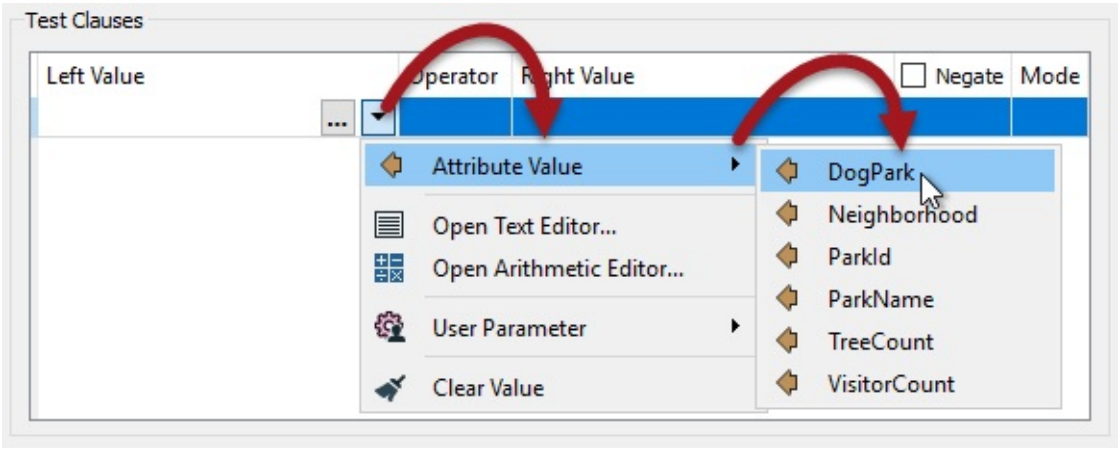
我们可以使用Tester转换器来做到这一点。单击从AttributeManager输出端口到写模块上的ParksMaintenanceData要素类型的连接。

开始输入单词Tester。当您发现Tester转换器时，双击它以将其中一个添加到工作空间。在整理画布布局后，工作空间现在看起来像这样：

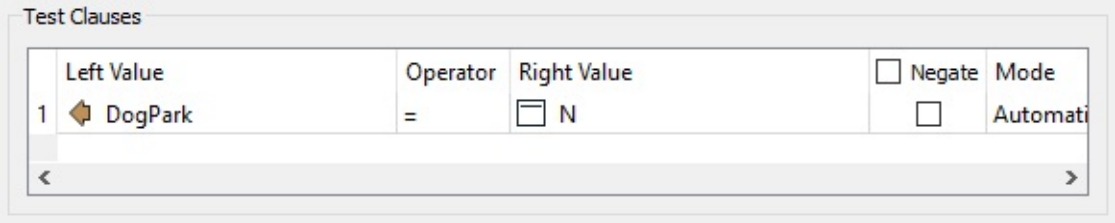


请注意，Passed输出端口是默认连接的输出端口。

5）设置参数
检查Tester转换器的参数。单击左值字段，然后单击向下箭头并选择属性值> DogPark：



对于正确的值，单击该字段并键入值N.运算符字段应自动填充为等号（=），这是我们在这种情况下所需的。

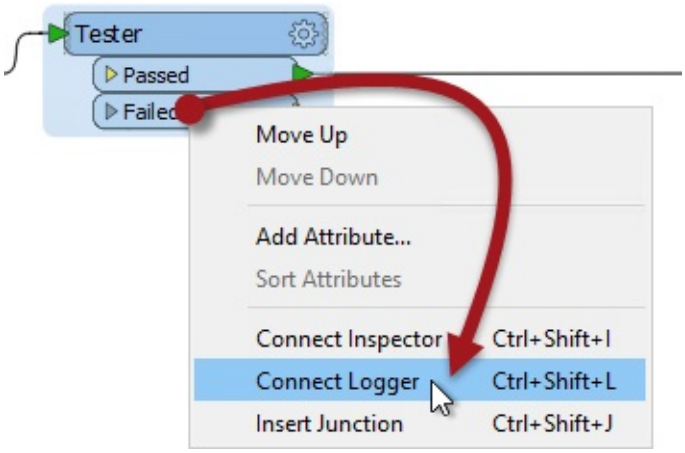


单击“确定”接受值并关闭对话框。

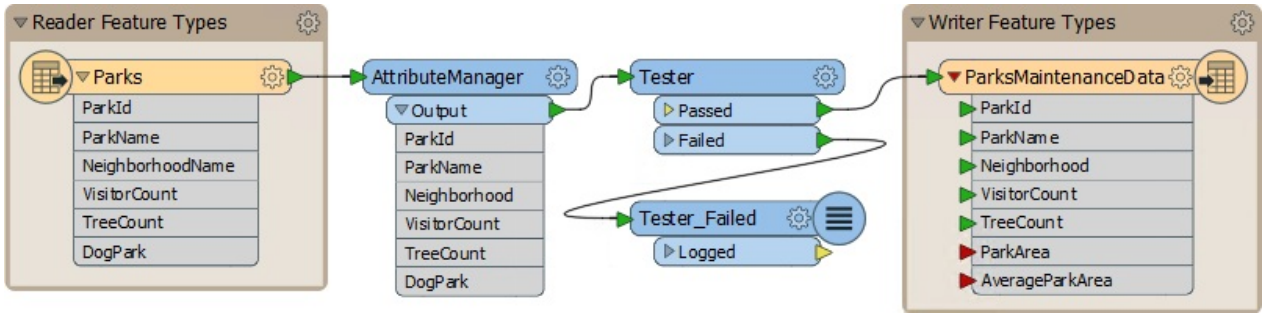
技巧
测试是针对DogParks = N因为我们想要保留这些要素，并且它是连接的Passed端口。如果测试是DogParks = Y，那么Failed端口将是连接的端口。

6）添加转换器
我们现在使用属性值测试从我们的数据中过滤掉狗公园。问题是，我们应该如何处理我们过滤掉的要素？我们可以做很多事情，但是现在，我们会将信息记录到FME日志窗口。

要执行此操作，请右键单击Tester Failed端口，然后选择Connect Logger选项：



Logger转换器将添加到工作空间。此转换器将所有传入要素记录到日志窗口：



通过此方法插入的Loggers以以下方式命名 - 并在日志中报告 - 它们连接的输出端口，此处为Tester_Failed。

7）运行工作空间
保存并运行工作空间。它尚未完成，但运行它将证明到目前为止一切正常。

高级练习
画布上的读模块和写模块自动在其周围添加了书签对象。您可以通过选择全部转换器并单击工具栏上的“书签”或按Ctrl + B键对转换器执行相同操作。为书签指定您选择的名称，以描述正在执行的操作。 书签是我们在FME中称为最佳练习的示例。

恭喜

通过完成本练习，您已学会如何：

- 将转换器添加到工作空间
- 使用AttributeManager转换器执行模式映射
- 使用Tester转换器过滤数据
- 使用Logger转换器记录数据
- 将书签添加到工作空间

内容转换

内容转换是对要素组件进行操作的内容转换。

什么是要素？

一个要素在FME是转换中的单个项目。对于空间数据，要素通常是几何对象（具有或不具有一组相关属性）。

对于表格数据，要素通常是数据库中的记录，电子表格中的行或文本文件中的行。每个列或单元称为属性。

FME中的要素具有灵活的通用表示，与其源自的格式无关。这意味着任何转换器都可以在任何FME要素上运行，无论其源格式如何。有时内容转换在单个要素上运行，有时在多个要素上运行。

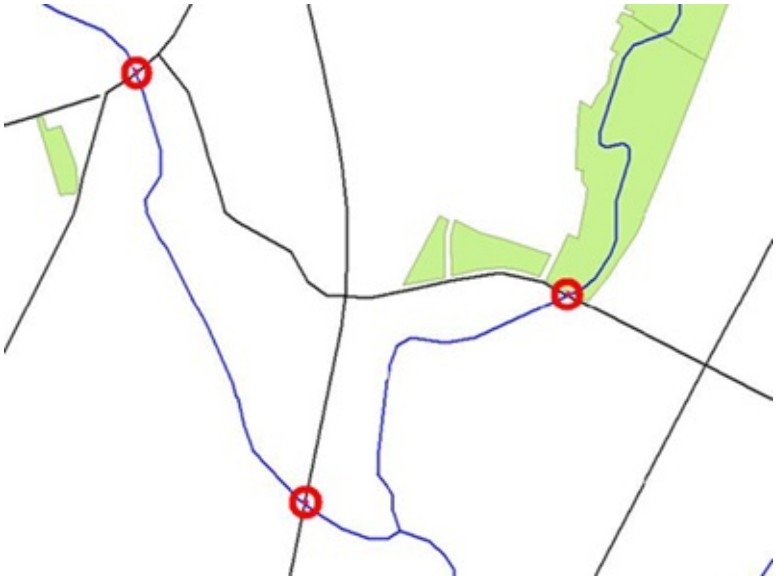
分析师女士说.....
您可以将内容转换视为更改或编辑数据。 衣柜类比仍然适用于此。您可以将衣服从衣柜中取出来进行清洁，或改变它们，或修理它们，或将它们染成新的颜色或各种其他任务，然后再将它们送回原位。 对于空间数据转换也是如此：它是将您的数据修复为更干净且符合您真正想要的风格的行为。

几何转换

几何转换是重构FME要素的空间组件的行为。换句话说，要素的几何形状经历某种形式的改变以产生不同的输出。

几何转换的一些示例包括以下内容：

- 泛化 - 一种重构数据的制图过程，因此可以在给定的地图比例下轻松显示
- 变形 - 调整一组要素的大小和形状，以更接近地匹配一组参考数据
- 拓扑计算 - 将一组线性要素转换为节点/线结构
- 线交叉 - 计算线要素之间的交点



在这里，道路与河流相交，产生了标志着桥梁位置的点。

属性转换

属性转换是重构FME要素的表格组件的行为。换句话说，属性经历某种形式的改变以产生不同的输出。

属性转换的一些示例是：

- 连接 - 将两个或多个属性连接在一起
- 拆分 - 将一个属性拆分为多个，这与连接相反
- 测量 - 测量要素的长度或面积以创建新属性
- **ID**创建 - 为特定要素创建唯一的ID号

属性连接作为属性转换的示例。

地址的每一行被连接起来以返回单行地址。

```
 Address1     Suite 2017,+
Address2     7445-132nd Street,+
City           Surrey,+
Province     British Columbia,+
PostalCode     V3W 1J8

= Address     Suite 2017, 7445-132nd Street, Surrey, British Columbia, V3W 1J8
```

Vector小姐说.....

你想念我吗？想念？好吧，我会为你解决一些新问题！

哪三种颜色代表检查过，需要检查和取消设置转换器对象的参数？
1.蓝色，黄色，红色
2.绿色，黄色，红色
3.红色，绿色，蓝色
4.绿色，蓝色，黄色

如果我使用转换器去除多边形边界的不规则（如自相交环），它是什么类型的转换？
1.属性的结构转换
2.几何的结构转换
3.属性的内容转换
4.内容几何的转换

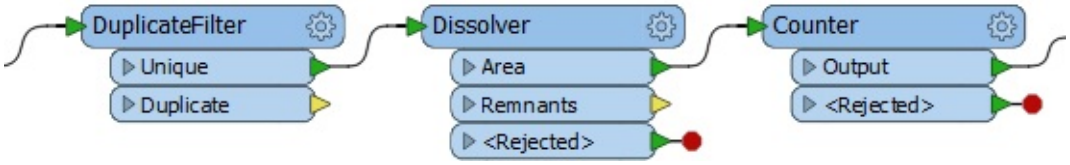
转换器系列

与电路中的一组元件非常相似，可以连接一系列Workbench转换器，以对一组要素产生累积效应。

链接转换器

即使在FME中有大量的转换器，用户也经常需要一个组合或一个变压器链，而不是一个单一的变压器。

一串以图形方式表示整个工作流的转换器是FME的一个关键概念：



在此示例中，DuplicateFilter转换器删除重复的多边形要素。Dissolver转换器将每个剩余（唯一）多边形与其邻居合并，其中存在公共边界。最后，每个合并区域从Counter转换器获得一个ID号。

练习：地面维护项目 - 计算统计数据

练习**3**	地面维护项目 - 计算统计
数据	城市公园（MapInfo TAB）
总体目标	计算城市中每个公园的大小和平均大小，以用于草地切割，树篱修剪等的地面维护估算。
演示	内容转换。模式映射
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex3-Begin.fmw
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex3-Complete.fmw C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex3-Complete-Advanced.fmw

让我们继续您的地面维护项目。

如果您忘记了，负责维护公园和其他草地区域的团队需要了解每个公园的面积和设施，以计划下一年的预算。

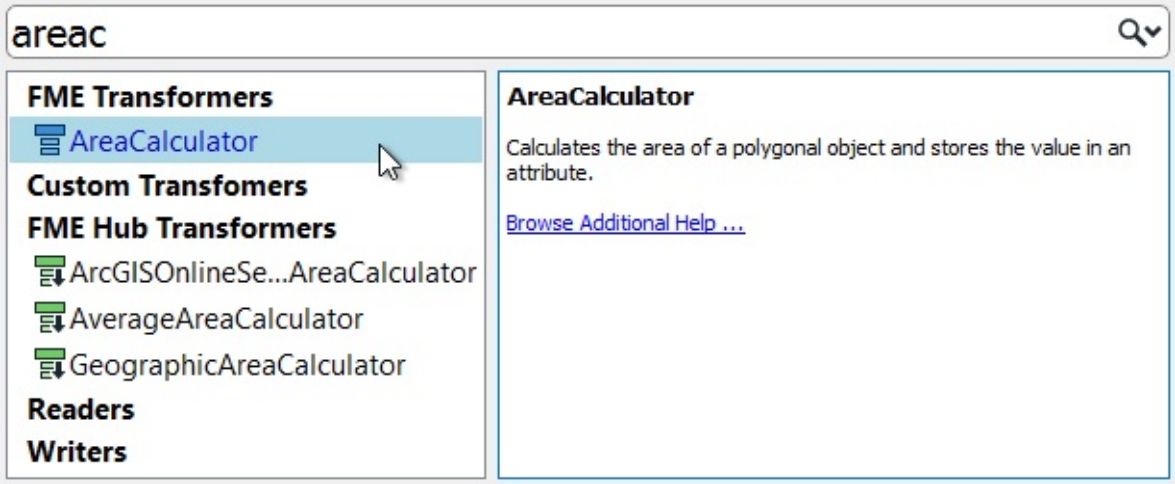
在项目的这一部分中，我们将计算每个园区的大小和平均大小，并确保信息正确映射到目标模式。

1）启动**Workbench**
启动Workbench（如有必要）并从练习2打开工作空间。或者，您可以打开C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex3-Begin.fmw

2）添加**AreaCalculator**转换器
要测量每个Park功能的面积，必须使用AreaCalculator转换器。“计算器”是FME计算新属性值时的术语。

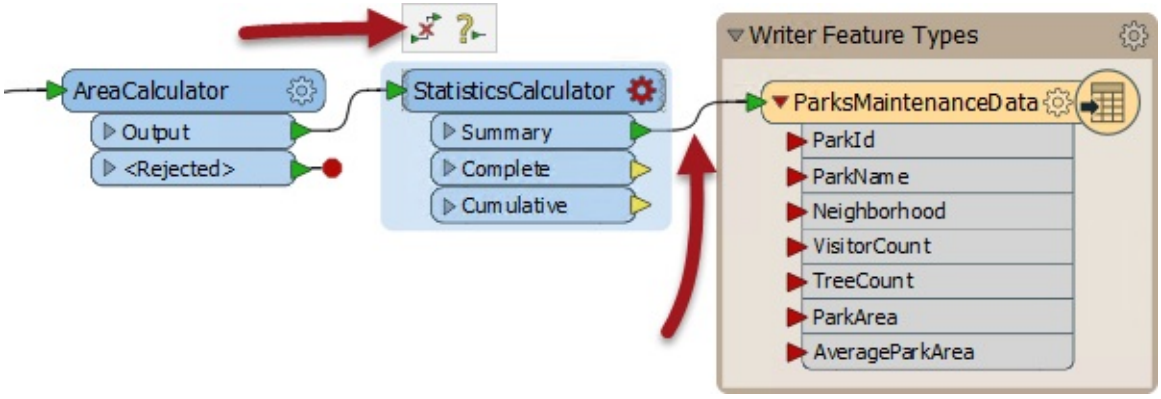
单击Tester：Passed port和写模块要素类型*ParksMaintenanceData*之间的连接。开始输入字母“areac”。您将看到下方显示匹配转换器的“快速添加”列表。

通过双击选择名为AreaCalculator的转换器：

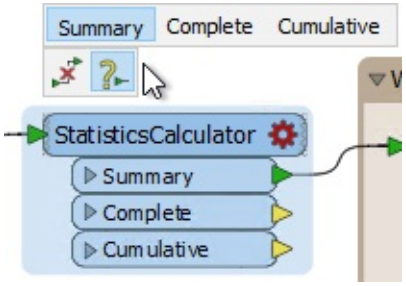


3）添加**StatisticsCalculator**转换器
使用相同的方法，在AreaCalculator：Output端口和*ParksMaintenanceData*要素类型之间放置StatisticsCalculator转换器。<

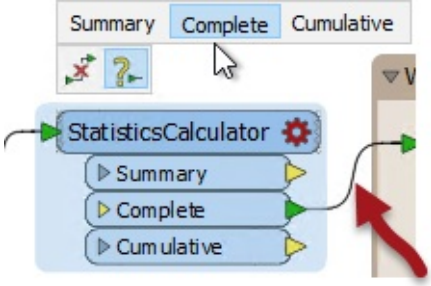
但是！不要点击其他任何东西！转换器现在看起来像这样：



默认情况下，摘要端口已连接，我们需要连接完整端口。但请注意顶部的小弹出图标。单击右侧图标（带有？字符的图标）。这样做会弹出更多端口列表：



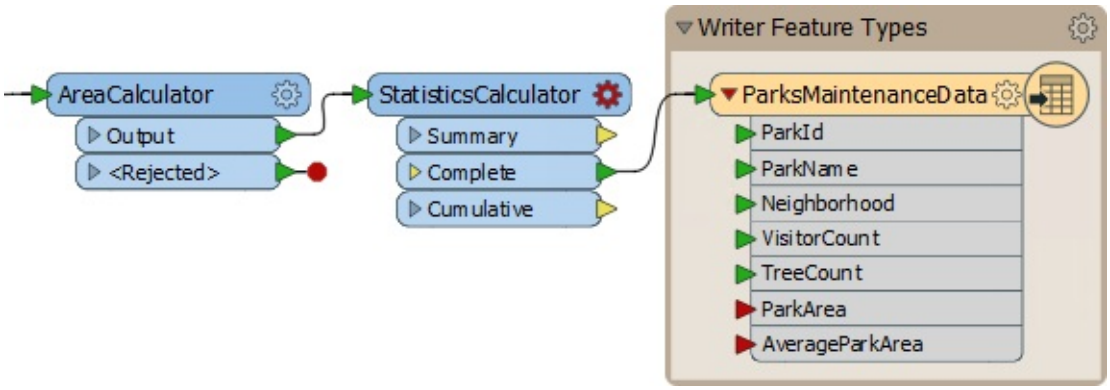
单击Summary端口条目以断开连接，然后在Complete端口条目上连接：



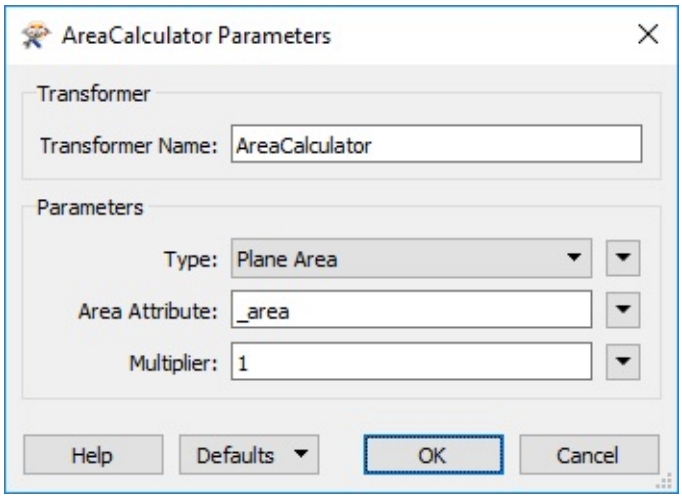
请注意连接如何从错误的摘要端口更改为正确的完整端口。

技巧
这些弹出菜单对模式映射和其他要素连接有很大帮助。

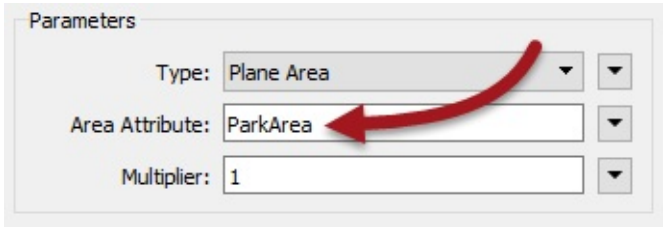
工作空间的后半部分现在看起来像这样：



4）检查**AreaCalculator**设置
检查AreaCalculator参数：



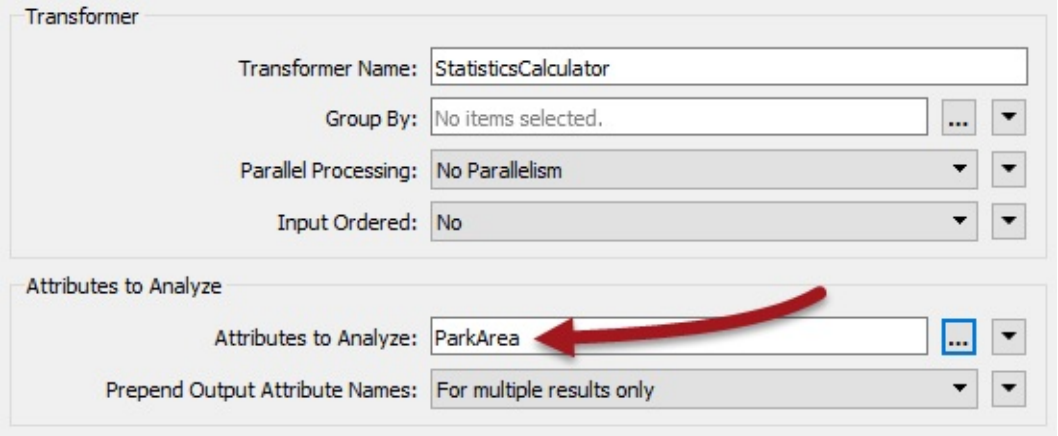
默认设置会将计算值放入名为_area的属性中。但是，*ParksMaintenanceData*模式需要一个名为ParkArea的属性，因此请更改此参数以创建正确的属性：



单击“确定”或“应用”后，请注意写模块要素类型上的属性现在标记为已连接。

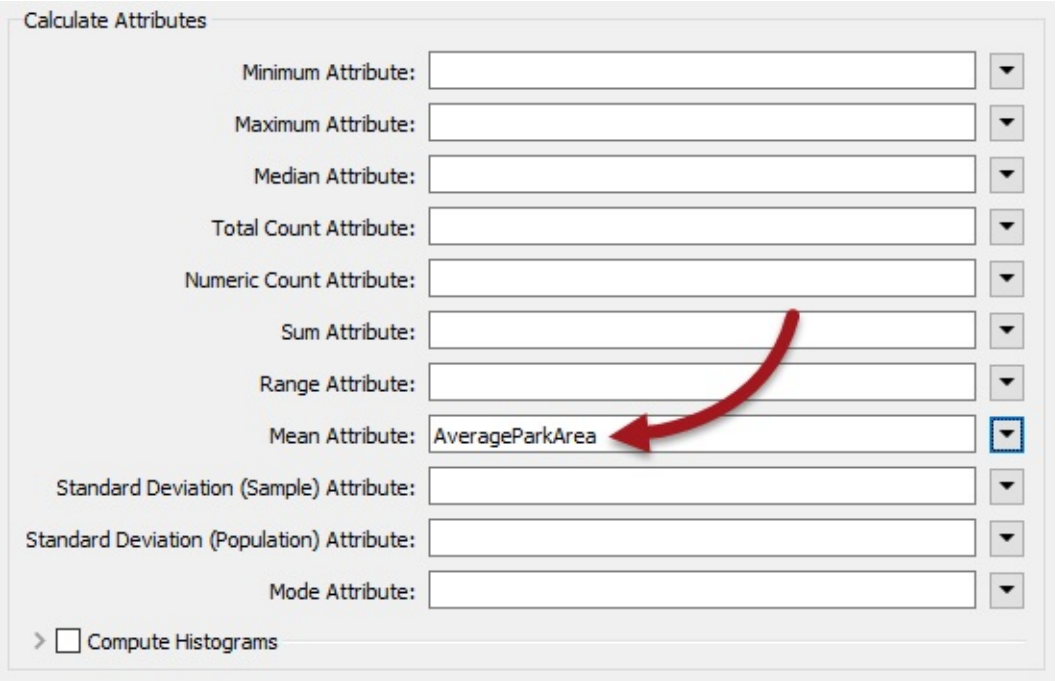
5）检查**StatisticsCalculator**设置
红色图标表示StatisticsCalculator具有需要定义的参数。

检查StatisticsCalculator转换器的参数。要分析的属性是包含计算面积的属性; 所以选择ParkArea：



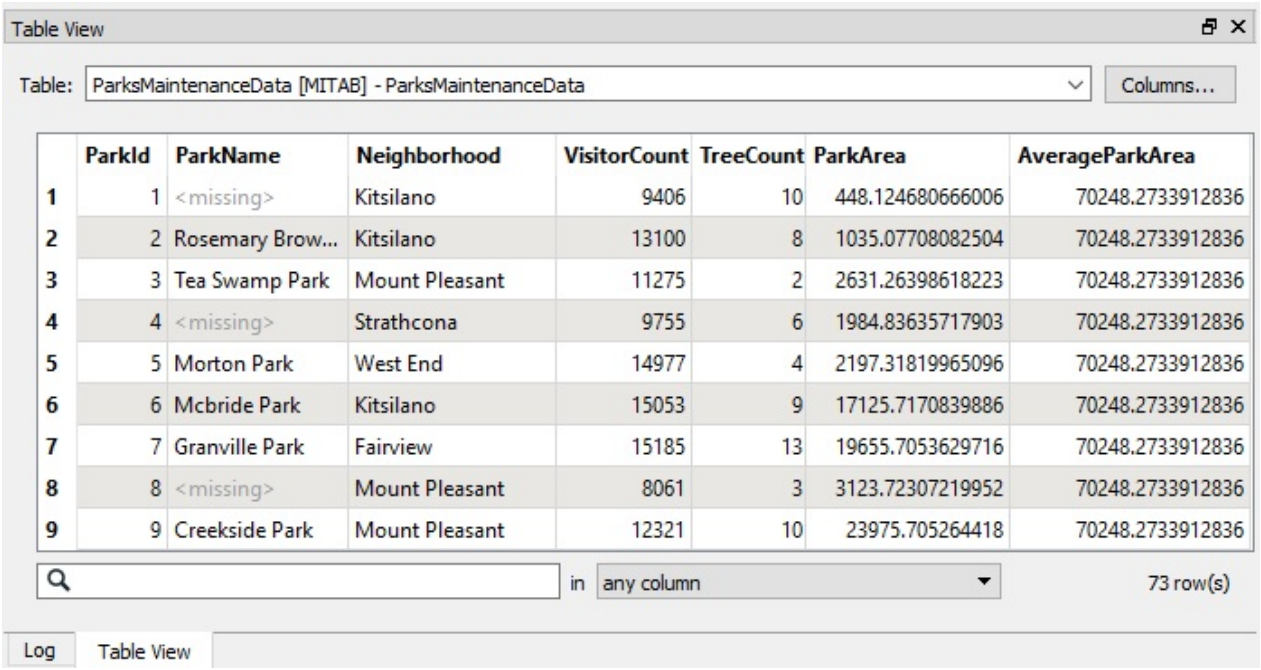
检查公园平均大小的属性名称的默认设置。目前，它与*ParksMaintenanceData*模式不匹配，后者需要名为AverageParkArea的属性。

将属性从_mean更改为AverageParkArea。出于最佳实践原因，删除/取消设置不需要的任何StatisticsCalculator输出属性（例如_range和_stdev）。



6）运行工作空间

如果需要，添加另一个书签，运行工作空间，并使用FME Data Inspector检查转换结果：



	ParkId	ParkName	Neighborhood	VisitorCount	TreeCount	ParkArea	AverageParkArea
1	1	<missing>	Kitsilano	9406	10	448.124680666006	70248.2733912836
2	2	Rosemary Brow...	Kitsilano	13100	8	1035.07708082504	70248.2733912836
3	3	Tea Swamp Park	Mount Pleasant	11275	2	2631.26398618223	70248.2733912836
4	4	<missing>	Strathcona	9755	6	1984.83635717903	70248.2733912836
5	5	Morton Park	West End	14977	4	2197.31819965096	70248.2733912836
6	6	Mcbride Park	Kitsilano	15053	9	17125.7170839886	70248.2733912836
7	7	Granville Park	Fairview	15185	13	19655.7053629716	70248.2733912836
8	8	<missing>	Mount Pleasant	8061	3	3123.72307219952	70248.2733912836
9	9	Creekside Park	Mount Pleasant	12321	10	23975.705264418	70248.2733912836

“表视图”窗口显示每个公园的面积和所有公园的平均面积。

7）保存工作空间
保存工作空间 - 它将在后面的示例中完成。

高级练习

请注意，表视图中的数字显示结果已计算为12位小数。这超出了您所需的精度。作为一项高级任务 - 如果你有时间 - 使用AttributeRounder转换器将值减少到只有2位小数。 如果您愿意，您还可以计算最小，最大和总面积; 但是如果希望它们出现在输出中，请不要忘记将它们添加到写模块模式中。

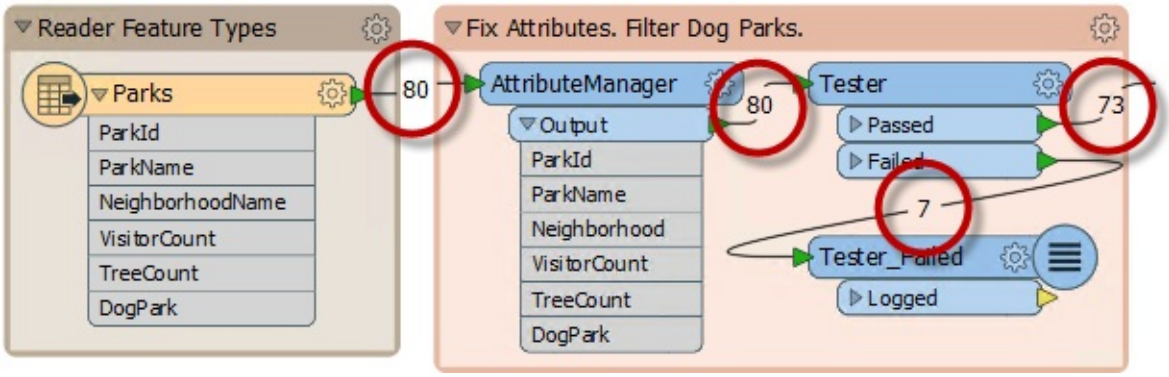
恭喜

通过完成本练习，您已学会如何：

- 用转换器进行内容转换（AreaCalculator，StatisticsCalculator）
- 使用弹出按钮管理转换器连接
- 使用转换器参数创建与写模块模式匹配的属性

要素计数显示

回顾前面例子的转换。您是否注意到在工作空间运行时，每个连接都更新了它传递的要素数量？



最后一项要素显示，有80个要素被读取，73个通过Tester，7个失败（作为狗公园）并继续使用Logger。

“日志”窗口确认已写入的要素数，并列出Tester未通过的要素。

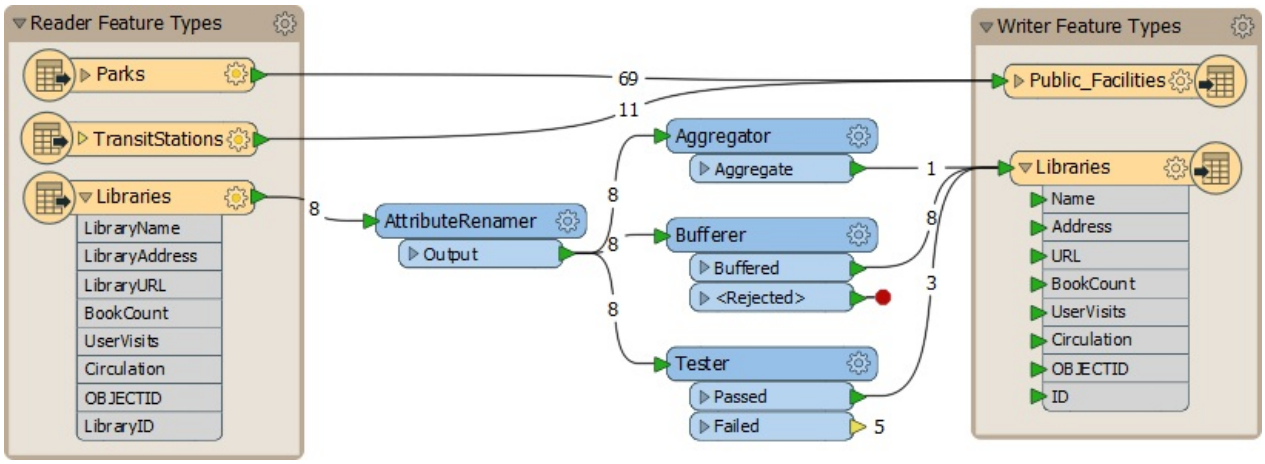
这些数字有助于分析工作空间的结果，并在输出与预期不同时提供调试参考。

转换器并行

“流”是通过在工作空间中的连接表示的数据的流程。FME中的一个关键概念是能够在工作空间内拥有多个并行流。

多个流

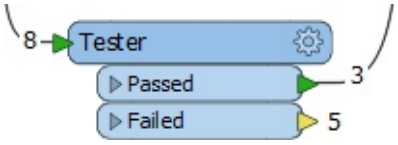
当用户需要以多种不同方式处理相同数据时，多个流非常有用。工作空间作者可以根据需要将一个流转换为多个，或将多个数据流合并为一个：



在这里，作者正在创建三个数据流，每个数据流分别处理，然后组合回一个流。

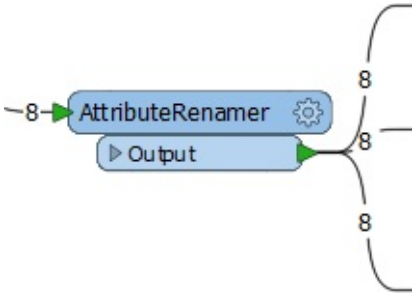
创建多个流

可以以多种方式创建多个数据流。有时，具有多个输出端口的转换器（Tester转换器就是一个很好的例证）将使用几种可能的输出流来划分（或过滤）数据：



这里数据分为两个流，其中一个流与任何东西都没有连接。

此外，只需从单个输出端口进行多个连接即可复制完整的数据流。实际上，它为每个连接创建一组数据：

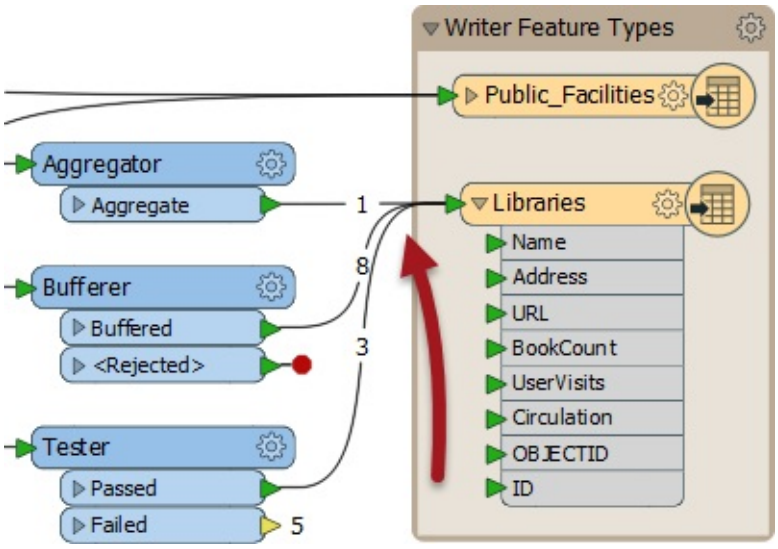


这里FME读取8个要素，但由于存在多个连接，因此会创建多个数据副本：

将多个流组合在一起

当多个流连接到同一输入端口时，不会发生合并。数据简单地聚合到单个流中。这通常被称为联合。

这里，三个数据流汇聚成一个写模块要素类型：



没有合并发生; 数据只会聚合到输出数据集中的12个不同要素中。可以把它想象成三管水排入一个容器。

要进行实际的数据合并，需要使用特定的转换器，例如FeatureMerger或FeatureJoiner。

练习：地面维护项目 - 标签功能

练习**4**	地面维护项目 - 标签要素
数据	城市公园（MapInfo TAB）
总体目标	计算城市中每个公园的大小和平均大小，以用于草地切割，树篱修剪等的地面维护估算。
演示	使用并行变换器进行内容转换
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex4-Begin.fmw
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex4-Complete.fmw C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex4-Complete-Advanced.fmw

让我们继续您的地面维护项目。

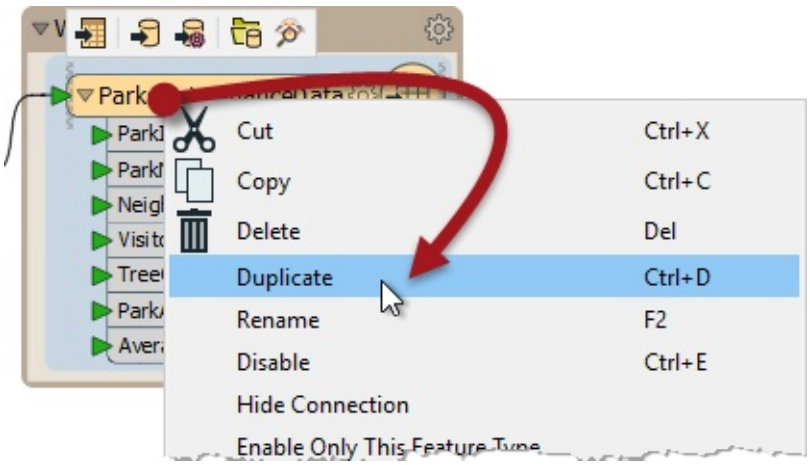
在项目的这一部分，我们将为每个公园创建一个标签，并将其写入新的输出层。此步骤最好使用并行数据流。

1）启动**Workbench**
启动Workbench（如有必要）并从练习3打开工作空间。或者，您可以打开C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex4-Begin.fmw

之前的练习使用AreaCalculator测量了公园面积。现在我们被要求将此信息作为标签添加到输出数据集中。

可以使用LabelPointReplacer转换器实现此步骤。

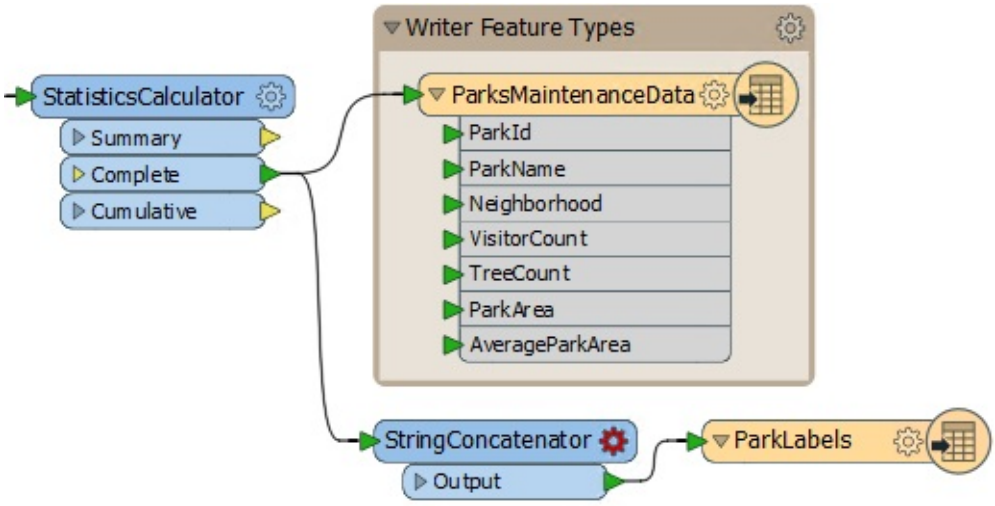
2）创建新的写模块要素类型
因为我们想要将标签要素写入输出中的单独层（表），我们需要在画布上创建另一个要素类型对象。在后面的章节中有更多相关内容，但是现在，右键单击编写器要素类型并选择“复制”选项。这样做会在输出数据集中创建新要素类型（图层）。



现在清理此要素类型的模式。查看要素类型的对话框，并将新类型重命名为ParkLabels。
在“用户属性”选项卡中，删除所有现有用户属性。

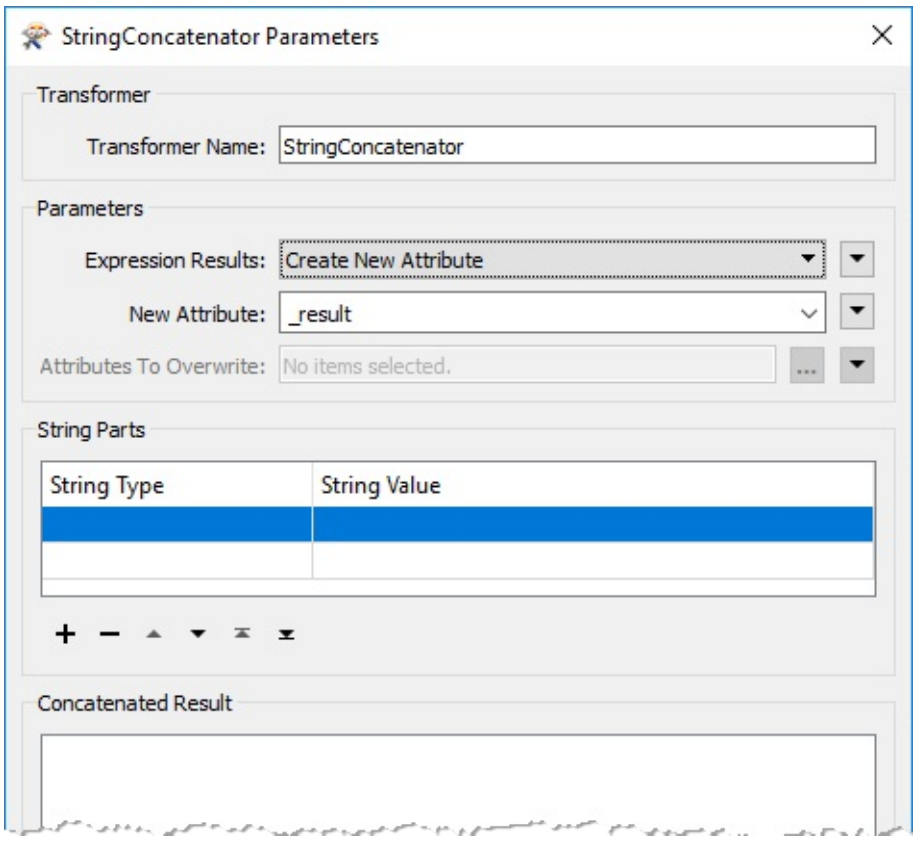
3）放置一个**StringConcatenator**转换器
单击画布的空白区域。键入“StringConcatenator”以添加此类型的转换器。

通过将第二个连接从那里拖到新变换器，将其连接到StatisticsCalculator的Complete端口。



从StringConcatenator建立到新要素类型的新连接。

4）检查转换器参数
查看StringConcatenator转换器的参数。有基本和高级对话框，基本对话框如下所示：

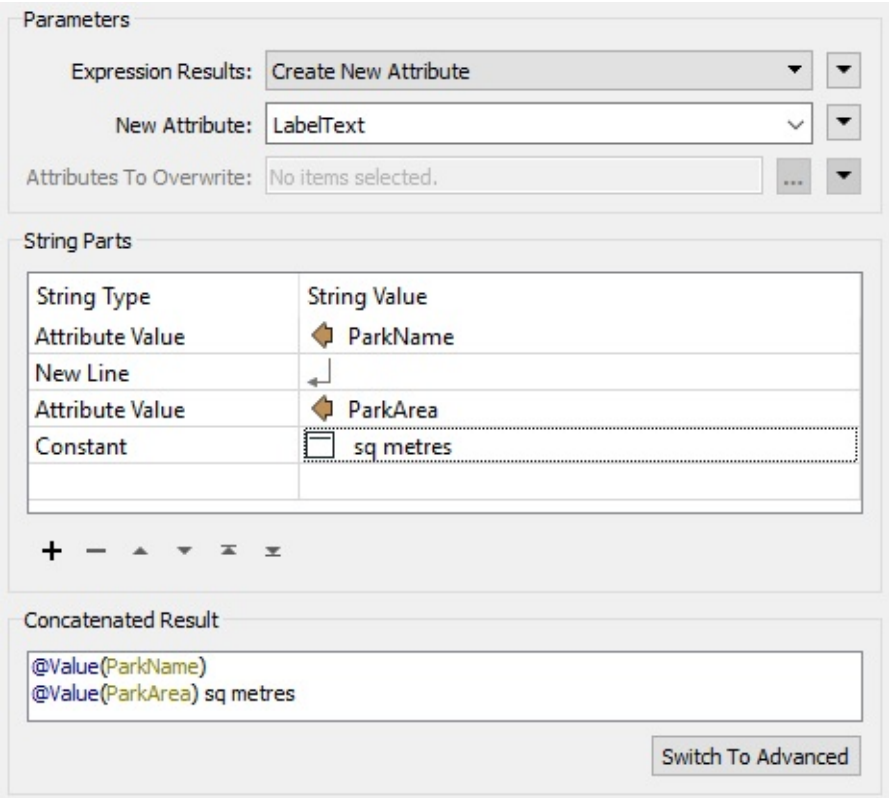


输入*LabelText*作为要创建的新属性的名称。

在“字符串部件”部分中，设置以下四个部分：

字符串类型	字符串值
属性值	ParkName
新线	
属性值	ParkArea
不变	平方米

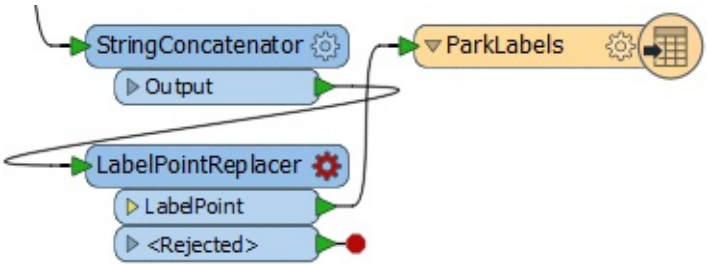
确保在“sq meters”之前的常量中包含空格字符。



技巧
您可能会发现切换到高级编辑器对话框并直接输入内容更快：@Value（ParkName）@Value（ParkArea）sq meters

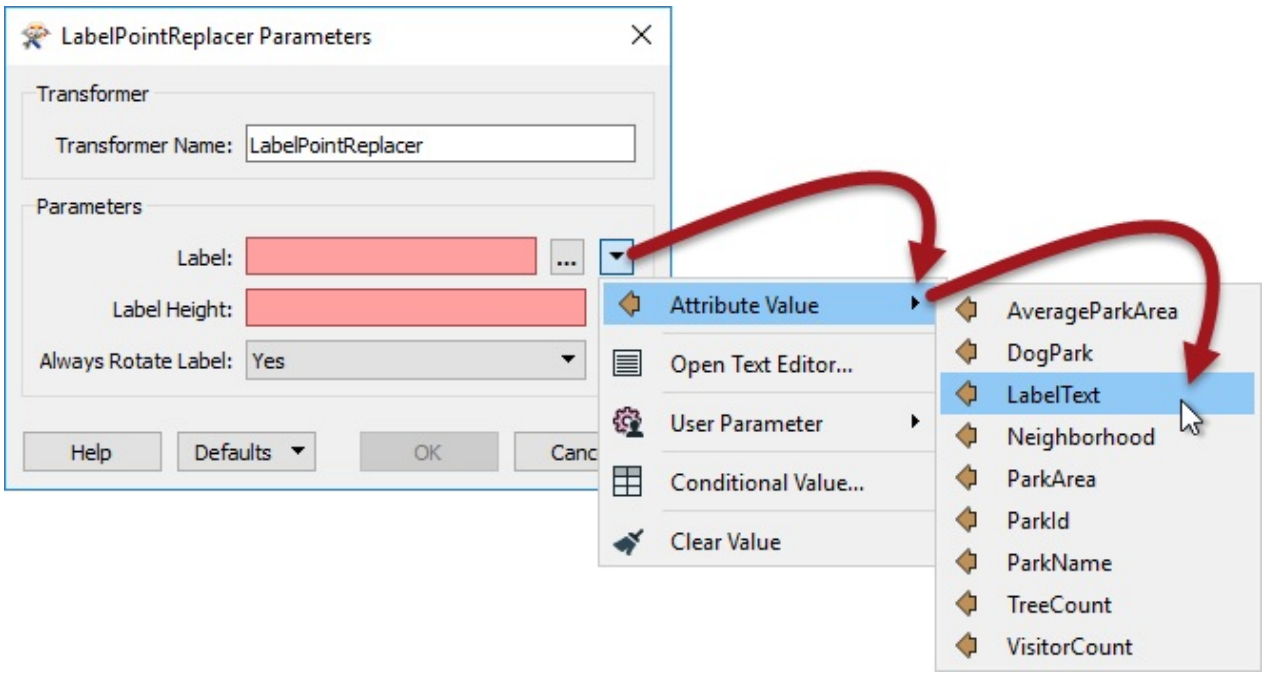
5）放置**LabelPointReplacer**转换器
单击StringConcatenator：Output和ParkLabels要素类型之间的连接。键入“LabelPointReplacer”以添加此类型的转换器。

将添加新转换器并自动连接这两个对象。



6）检查**LabelPointReplacer**参数
检查LabelPointReplacer参数。

首先单击Label参数右侧的下拉箭头：



选择“属性值”>“LabelText”以选择先前在StringConcatenator中定义的标签。

现在单击Label Height字段并键入25（即25个工作单位，在本例中为米）。

“始终旋转标签”参数可以保留其默认值。

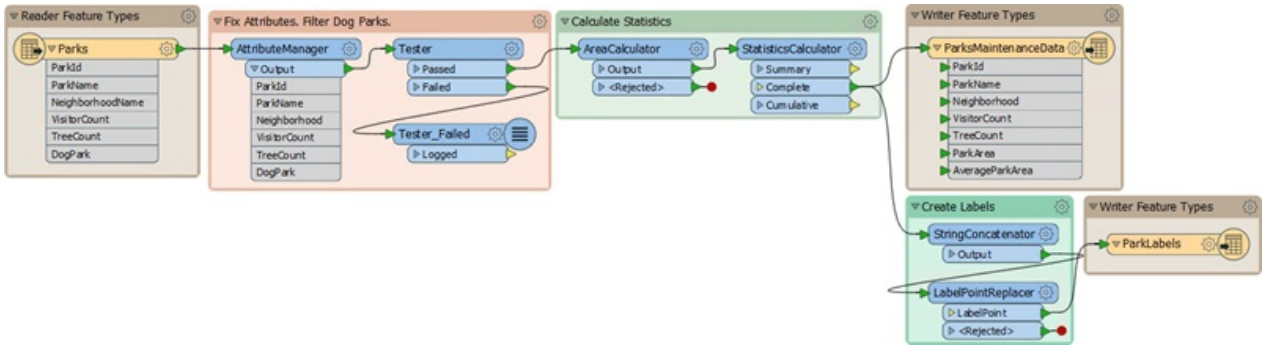
技巧
许多参数字段（如Label Height）可以设置为常量值（通过键入）或通过单击下拉箭头并选择属性值设置为属性。 并且 - 您很快就会看到 - 也可以直接在转换器设置中构造参数值

7）运行转换
如果需要，添加另一个书签，运行转换，并检查输出。

请注意，输出在两个文件中分为两层。使用FME Data Inspector在同一视图中打开两个输出文件。

地图瓦片由Stamen Design，许可由CC-BY-3.0。数据提供OpenStreetMap，许可由CC-BY-SA。

保存工作空间 - 将在其他示例中完成。



高级练习
现在您知道如何在输出中创建新要素类型（图层），如何测试数据以及如何使用并行流，为什么不尝试此任务：确定哪些公园小于平均值，哪些公园大于平均值，并将它们写出为于不同的要素类型。

恭喜
通过完成本练习，您已学会如何： • 创建新的写模块要素类型 • 在单个工作空间中使用多个转换器流 • 使用StringConcatenator构造一个字符串以供其他地方使用 • 使用属性作为转换器参数的值

分组处理

分组参数允许通过单个FME转换器分组处理要素。

什么是组？

FME转换器一次对一个要素进行变换，或一次对一整套要素进行变换。

例如，*AreaCalculator*转换器一次对一个要素进行操作（以测量单个多边形要素的面积）。我们称之为基于要素的转换器。

*StatisticsCalculator*一次处理多个要素（以计算所有这些的平均值）。在FME中，我们将这组要素称为组，转换器是基于组的转换器。

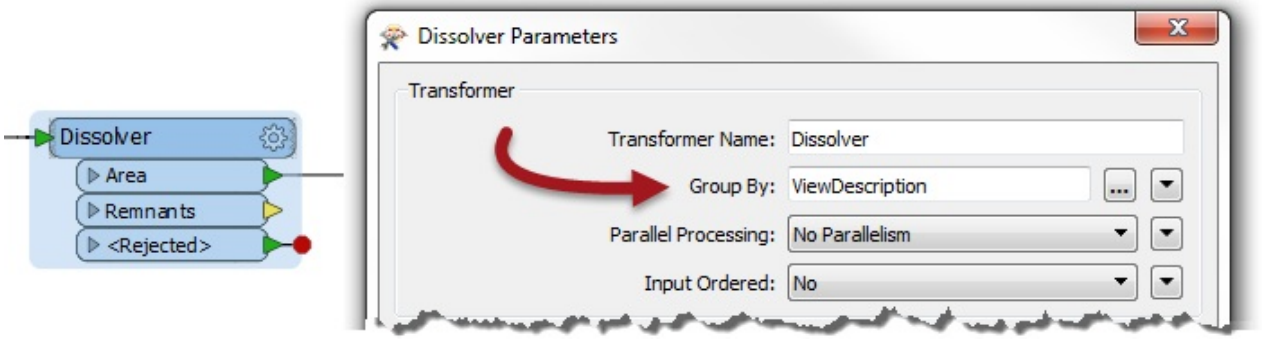
创建组

因此，组只是由转换器处理的一组已定义的要素。默认情况下，基于组的转换器会将其作为单个组提供的所有要素视为一个组。

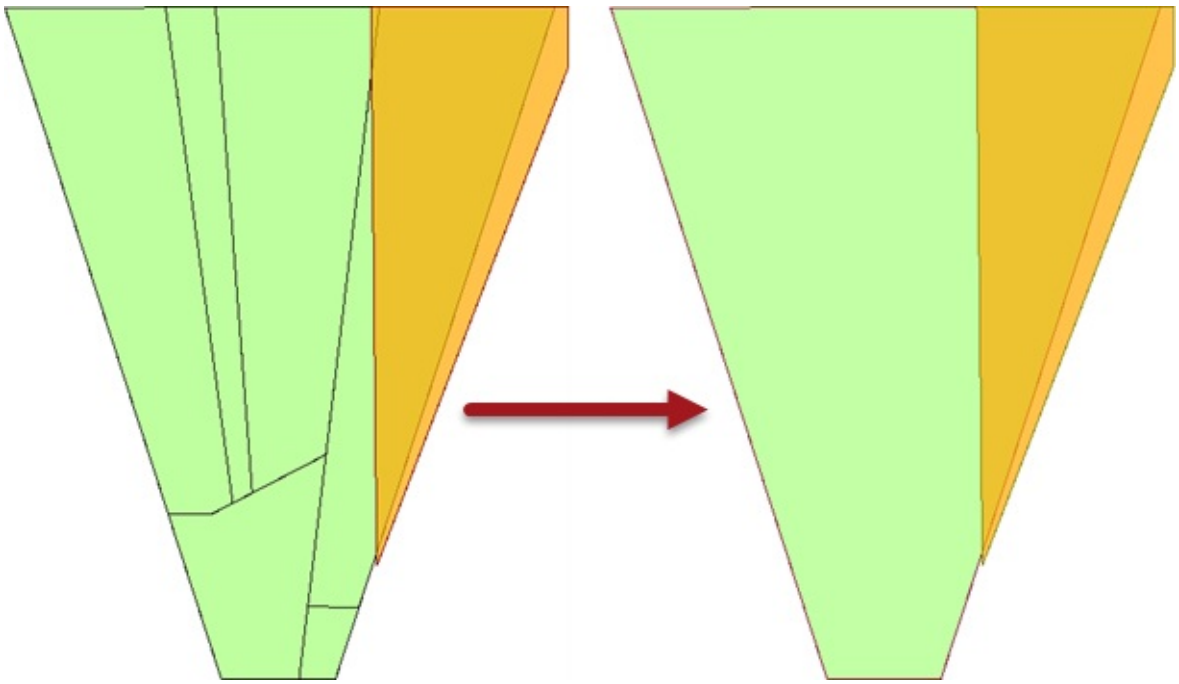
但是，此类转换器也具有***Group-By***参数。此参数允许用户根据属性的值定义多个组。

Statistics-Calculator先生，首席财务官说**.....**
你好。我想我们还没见过面。我是Statistics-Calculator先生。我打赌你猜不到我最喜欢的转换器！ 为了说明组，我们考虑计算FME用户的平均年龄。别担心，我会分门别类的（哈哈）！计算的默认组包括所有 FME用户。 但是你可以把所有人分成男女，创建两组，并计算每个性别的平均年龄。或者你可以将每个人分成他们的国籍，并计算每个国家的平均年龄。 这与在数据集中具有性别（或国籍）属性并在FME group-by参数中选择该属性相同。

这里，Dissolver转换器用于溶解（合并）许多面要素。选定的Group-By属性是*ViewDescription*：



FME创建一系列用于覆盖的组，其中每个组中的要素共享*ViewDescription*属性的相同值。实际结果是多边形溶解只发生在线要素共享相同描述的地方：



Vector小姐说.....

让我们看看你是否已经了解了基于群组的转换是什么。

您认为以下哪种转换器是“基于组”的？请随意使用Workbench来帮助您回答这个问题。

1. StringFormatter
2. Clipper
3. Rotator
4. AttributeRounder

练习：地面维护项目 - 街区平均值

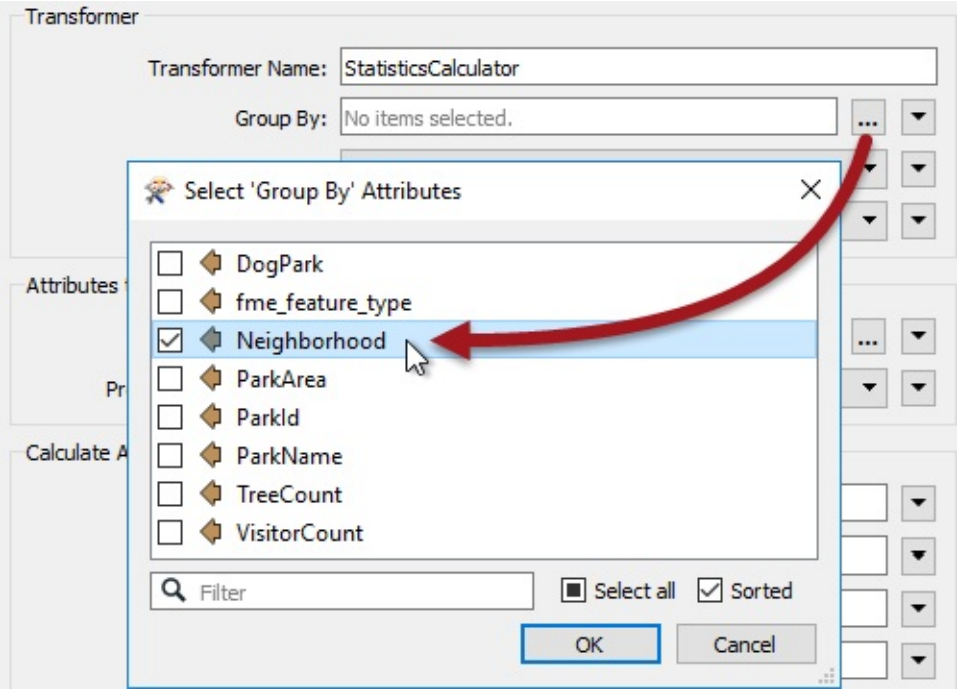
练习**5**	地面维护项目 - 街区平均值
数据	城市公园（MapInfo TAB）
总体目标	计算城市中每个公园的大小和平均大小，以用于草地切割，树篱修剪等的地面维护估算。
演示	分组处理
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex5-Begin.fmw
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex5-Complete.fmw

让我们继续您的地面维护项目。

公园团队已决定他们不希望整个城市的平均公园面积。相反，他们想要每个社区的平均公园大小; 让我们为他们这样做。

1）启动**Workbench**
启动Workbench（如有必要）并从练习**4**打开工作空间。或者，您可以打开C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex5-Begin.fmw

2）在**StatisticsCalculator**中设置**Group-By**
这是一项非常简单的任务。查看StatisticsCalculator转换器的参数，然后单击Group By参数旁边的“浏览”按钮。选择名为Neighborhood的属性：

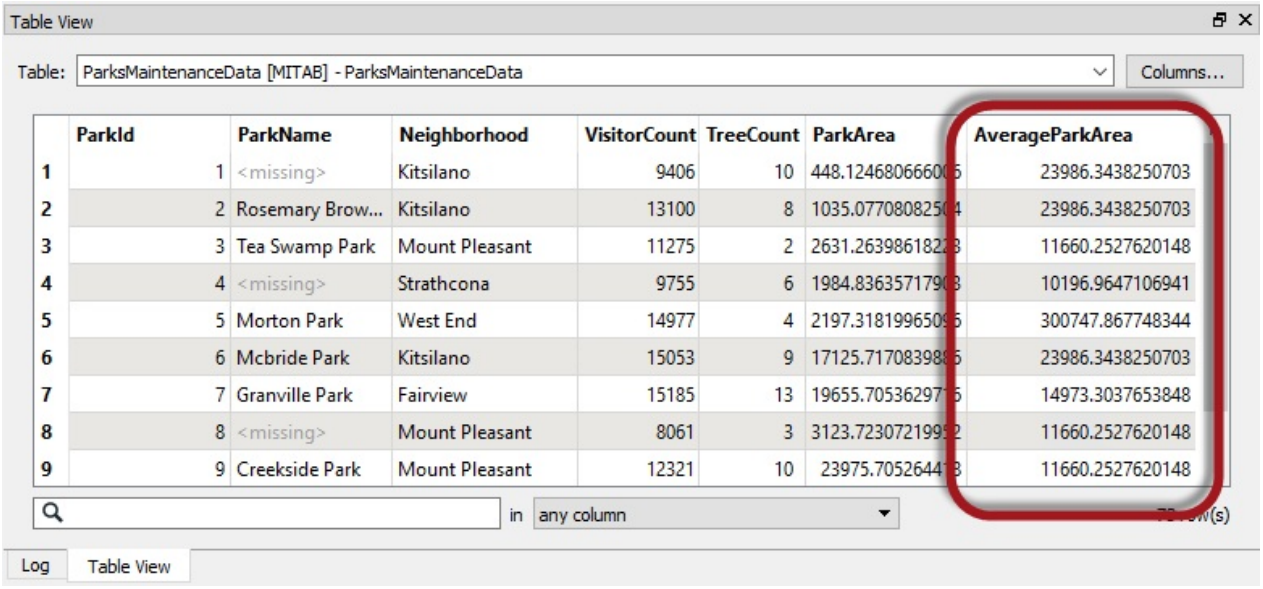


单击“确定”/“应用”以将更改应用于转换器。

3）运行工作空间
保存，然后运行工作空间。

检查FME Data Inspector的表视图窗口中的输出数据。

您应该看到每个街区现在都有自己的AverageParkArea值：



	ParkId	ParkName	Neighborhood	VisitorCount	TreeCount	ParkArea	AverageParkArea
1	1	<missing>	Kitsilano	9406	10	448.12468066600	23986.3438250703
2	2	Rosemary Brow...	Kitsilano	13100	8	1035.07708082504	23986.3438250703
3	3	Tea Swamp Park	Mount Pleasant	11275	2	2631.26398618223	11660.2527620148
4	4	<missing>	Strathcona	9755	6	1984.83635717903	10196.9647106941
5	5	Morton Park	West End	14977	4	2197.31819965095	300747.867748344
6	6	Mcbride Park	Kitsilano	15053	9	17125.71708398865	23986.3438250703
7	7	Granville Park	Fairview	15185	13	19655.70536297165	14973.3037653848
8	8	<missing>	Mount Pleasant	8061	3	3123.72307219952	11660.2527620148
9	9	Creekside Park	Mount Pleasant	12321	10	23975.70526447	11660.2527620148

恭喜

通过完成本练习，您已学会如何：

- 在FME转换器中使用分组参数

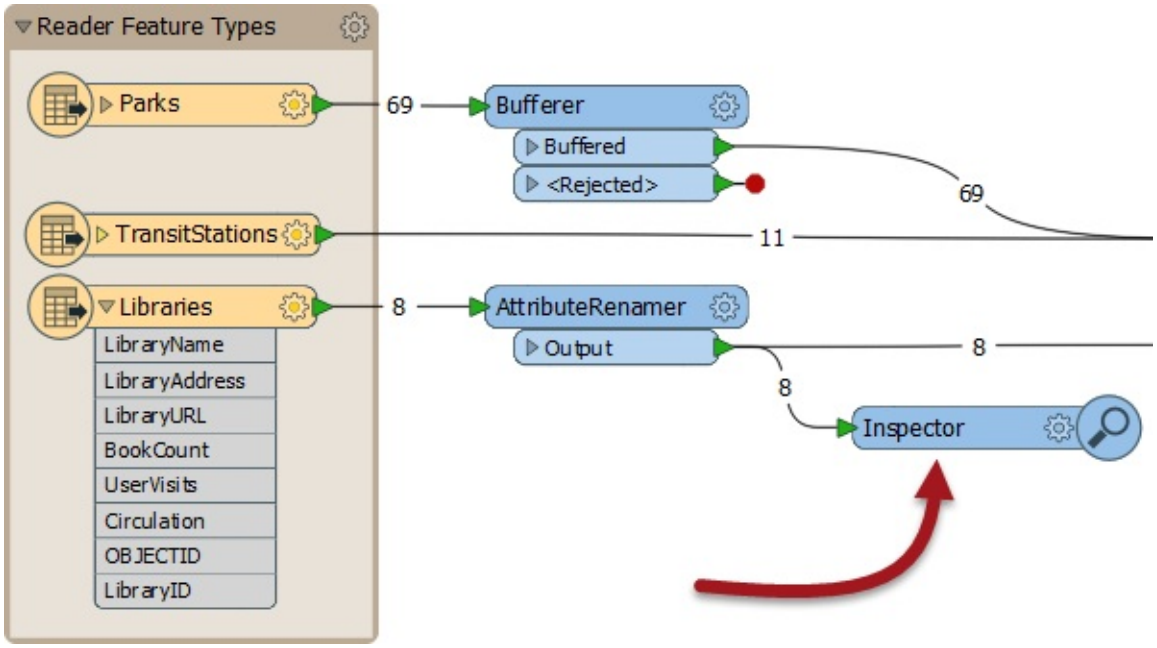
Workbench的数据检查

为了帮助检查数据，Workbench可以将数据从各个转换器路由到FME Data Inspector。

使用Inspector转换器

一个***Inspector***是一个具有独特外观和风格的Workbench转换器，它导致输入数据的数据被定向到FME Data Inspector，甚至在转换中间。

Inspector - 与任何转换器一样 - 可以应用于转换中的任何点，并且不会阻止数据输出到写模块。它还允许用户选择应检查哪些要素：

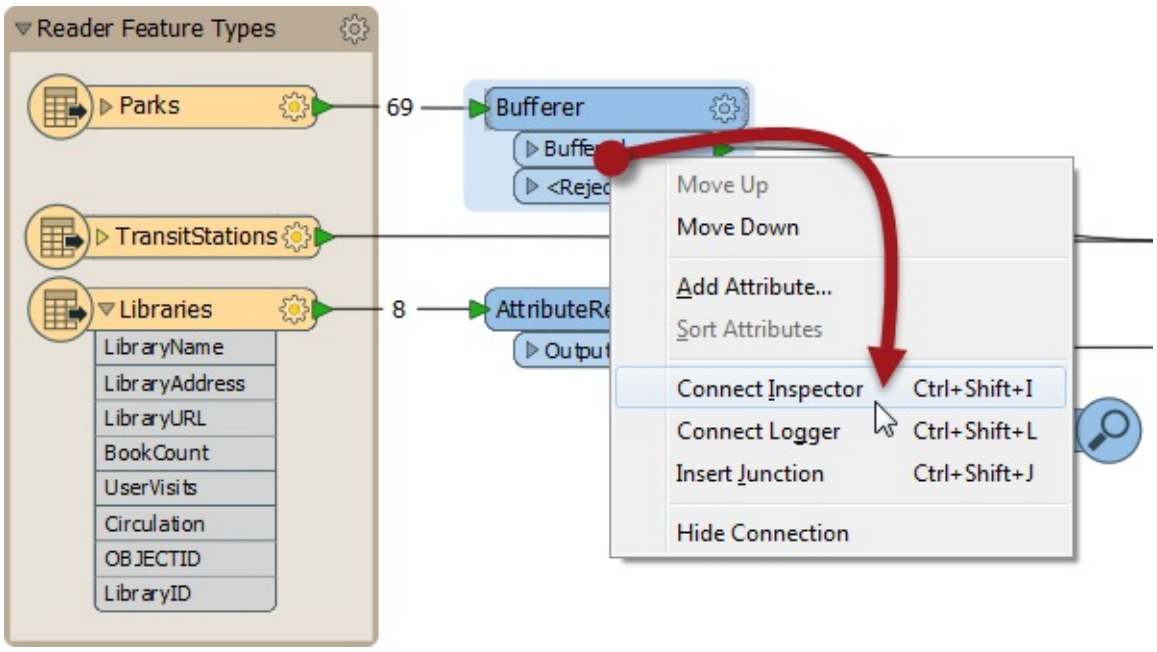


在这种情况下，它也是一种并行流形式，因为数据是重复的，但不一定是这种情况。

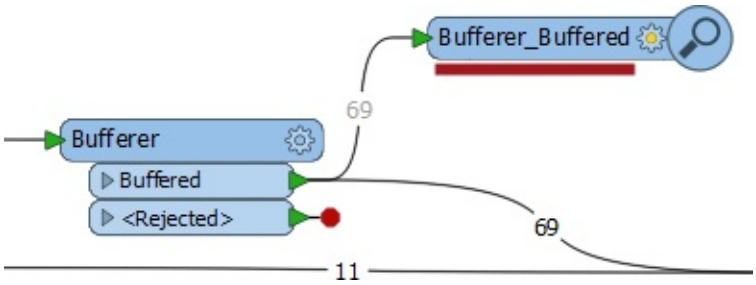
放置Inspector转换器

应用Inspector的最佳和最简单的方法是右键单击工作空间中对象的输出端口，然后选择“连接检查器”选项。

在这里，用户右键单击Bufferer转换器的缓冲端口，然后选择Connect Inspector选项：



请注意，使用转换器和输出端口名称自动命名Inspector。这里将是“Bufferer_Buffered”：



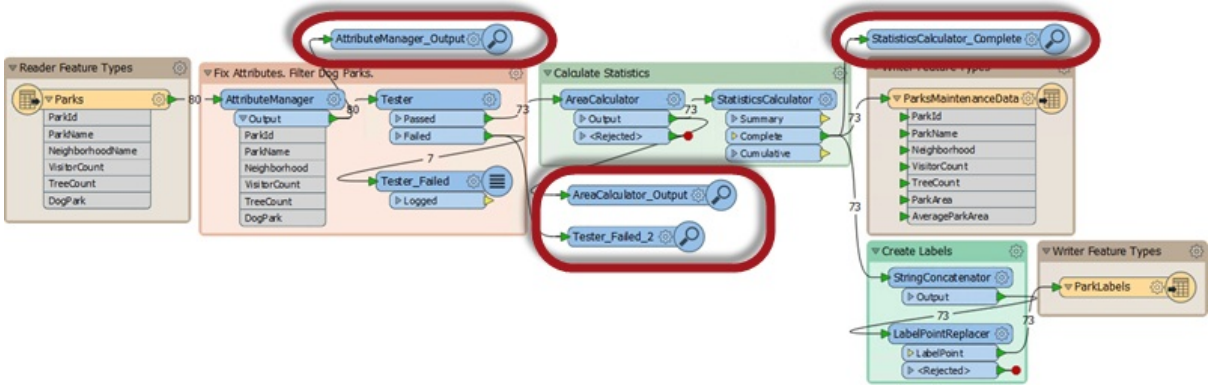
此名称有助于在FME Data Inspector中识别此Inspector中的数据（而不是其他任何数据）。

请注意，Inspector转换器仅在有要查看的要素时打开FME Data Inspector。如果要素为零，则数据检查器将无法打开！

技巧
Inspectors对于要投入生产的工作空间没有用，特别是在FME Server上。FME服务器引擎不具有打开Data Inspector的能力。因此，FME Server工作空间的作者倾向于使用Logger转换器而不是Inspector。

E. Dict先生（FME律师）说.....

根据您的培训课程协议中的第d30条（第43条），您需要重新打开上一个示例中的工作空间，并使用此要素添加Inspector转换器进行练习。



坐标系转换

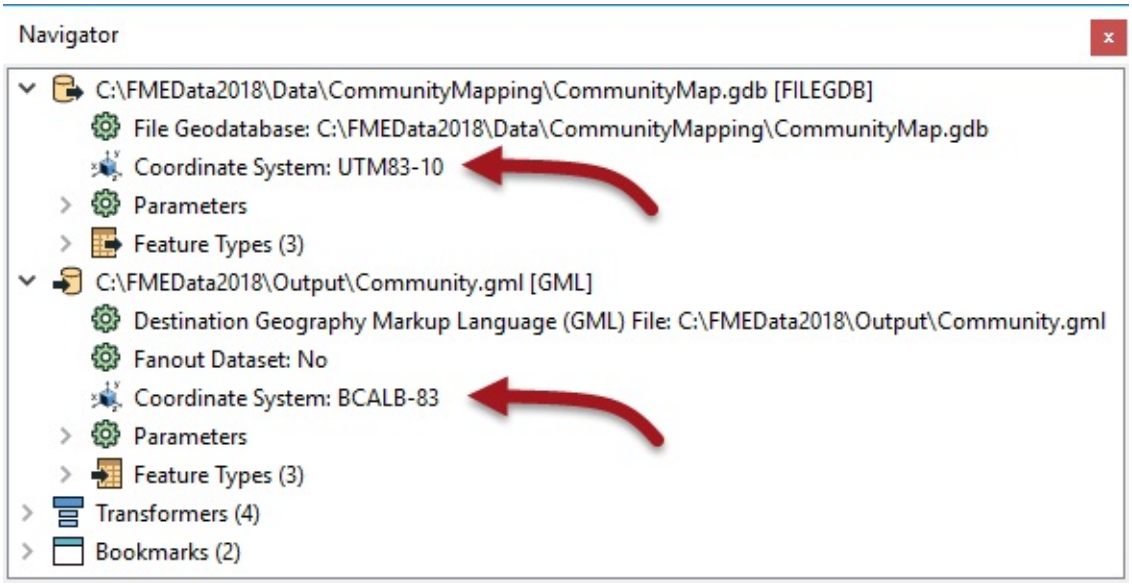
为了定位在地球表面上的特定空间中，大多数空间数据与特定空间基准有关。

一些用户将这个数据位置称为“投影”，但投影只是我们称之为坐标系. 的一个组成部分。坐标系包括坐标系包括投影，基准，椭球，单位，有时还有象限。

坐标系设置

可以为FME中的每个读写模块分配一个坐标系。该坐标系在Workbench的导航窗口中或生成工作空间对话框中设置。

与源模式一样，读模块坐标系是“我们拥有的”而写模块者坐标系是“我们想要的”. 这里源坐标系已定义为UTM83-10，目标为BCALB-83：



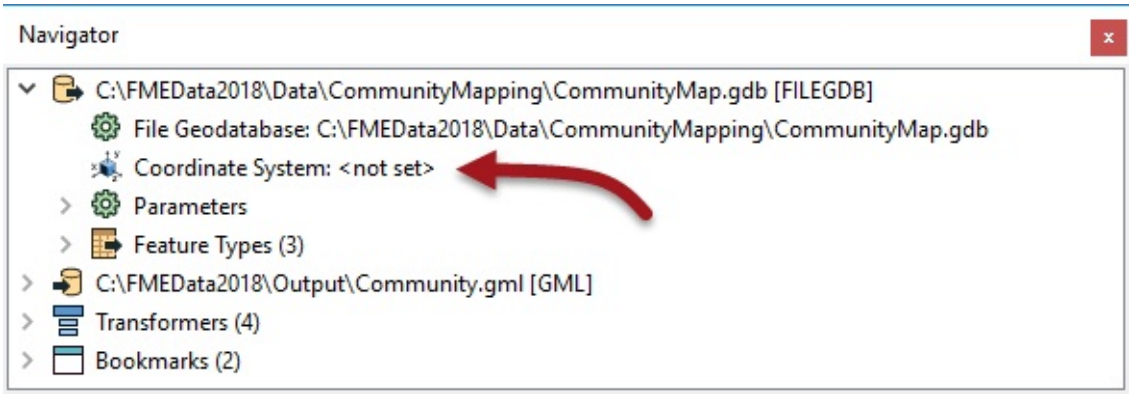
读模块处理的每个要素都用其参数中定义的坐标系标记。

当一个要素到达一个写模块时，如果它被标记为与该写模块定义的坐标系不同，则FME会自动重新投影数据，以便输出位于正确的位置。

技巧
一旦用坐标系标记，每个要素在整个转换过程中保留这一点; FME始终知道它属于哪个坐标系。 这在执行几何变换（如计算面积）或读取属于不同坐标系的多个数据集时很重要（是的，FME会处理）。

坐标系的自动检测

并不总是需要手动设置坐标系参数。某些数据格式（例如Esri Shapefile）能够存储有关它们所在的坐标系的信息，FME将在可能的位置检索此信息。



这里，因为读模块坐标系标记为<未设置>，FME将尝试从源数据集确定坐标系。如果不能，则该要素将使用的坐标系标记。

根据可用的坐标系（CS）信息的组合，可能会出现许多重投影情景：

数据集 **CS**	读模块 **CS**	写模块 **CS**	重投影
N	Y	Y	从读模块CS重新投影到写模块CS
Y	N	Y	从数据集CS重新投影到写模块CS
N	N	Y	Error: 错误：没有数据集或读模块CS无法重新投影/td>
Y	Y	Y	从读模块CS重新投影到写模块CS

如果未在写模块上设置坐标系，则除非输出格式需要，否则不会进行重新投影。例如，KML格式要求数据为纬度/经度。如果未定义源数据集或读模块坐标系，则转换将失败。

练习：地面维护项目 - 数据重投影

练习**6**	地面维护项目 - 数据重投影
数据	城市公园（MapInfo TAB）
总体目标	计算城市中每个公园的大小和平均大小，以用于草地切割，树篱修剪等的地面维护估算。
演示	数据重投影
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex6-Begin.fmw
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex6-Complete.fmw C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex6-Complete-Advanced.fmw

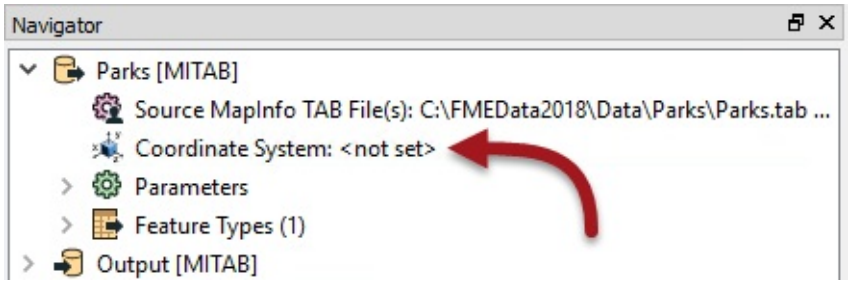
让我们继续您的地面维护项目。

公园团队已决定输出数据应采用Albers Equal Area投影（坐标系= BCALB-83）。他们认为设置它需要很长时间！我们将以不同的方式展示它们。

1）启动**Workbench**
启动Workbench（如有必要）并从练习5打开工作空间。或者，您可以打开C:\FMEData2018\Workspaces\DesktopBasic\Transformation-Ex6-Begin.fmw

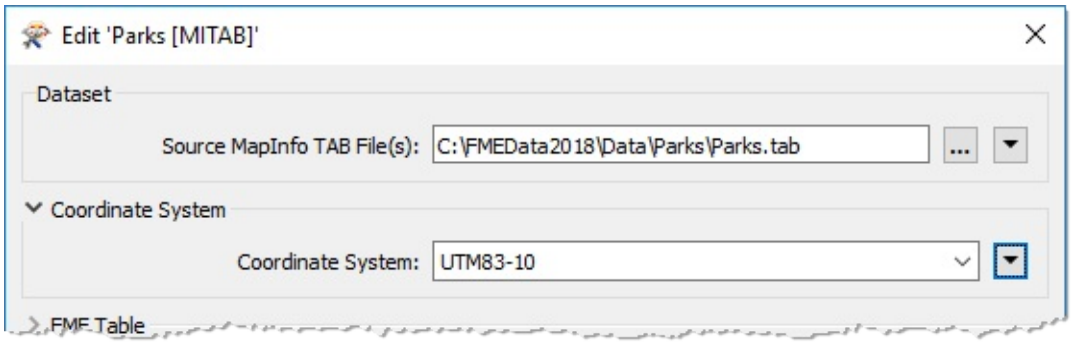
2）编辑读模块坐标系
在导航器窗口中找到Parks [MITAB]读模块，然后展开其设置列表。

找到标有“坐标系”的设置。原始值应<设置>：



双击读模块“坐标系”参数以打开编辑对话框。

在“坐标系”字段中输入名称UTM83-10，或者从下拉列表底部选择“更多坐标系...”从坐标系统库中选择它：



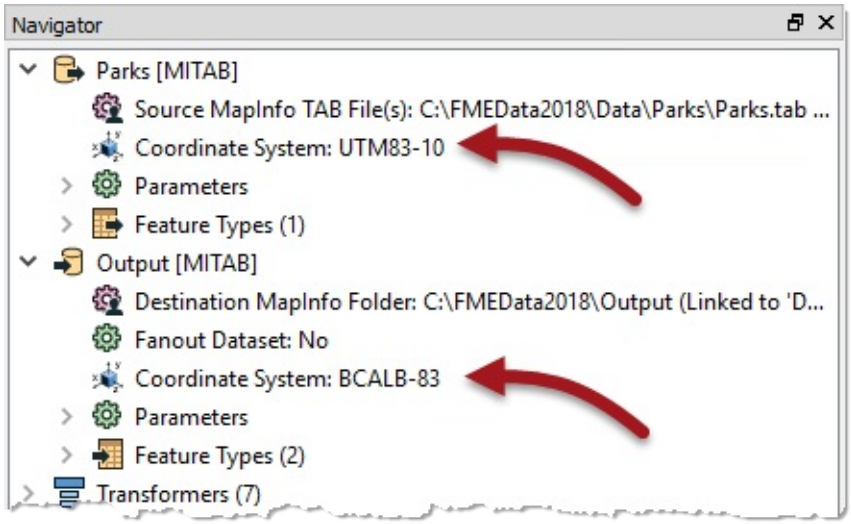
技巧
请记住，当读模块的坐标系参数定义为<未设置>时，FME将自动尝试从数据集本身确定正确的坐标系。 当源数据集采用存储坐标系信息的格式时（就像在此示例中一样），您可以安全地保留参数未设置。所以这一步并非真的有必要。 但是，当您希望重投影不存储坐标系信息的源数据时，必须设置此参数; 否则转换中会出错。

3）编辑目标坐标系
现在找到目标（写模块）数据集的坐标系设置。

同样，当前值应该是<not set>的默认值。

双击参数并输入坐标系名称BCALB-83，或通过从下拉列表底部选择“更多坐标系...”从坐标系统库中选择它。

Navigator窗口现在看起来像这样：



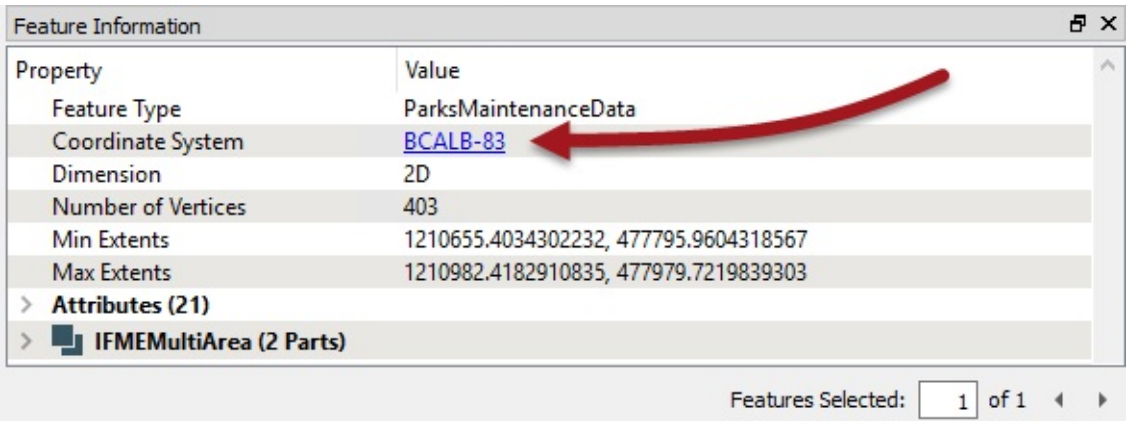
4）运行工作空间
保存，然后运行工作空间。

在日志文件中，您应该能够找到：

```
FME配置：读模块MITAB_1 [MITAB]的源坐标系设置为“UTM83-10”
FME配置：目标坐标系设置为“BCALB-83”
```

5）检查输出

打开新的重投影数据集并查询要素。“要素信息”窗口应报告数据现在位于BCALB-83中。（可选）单击该窗口中的坐标系名称; 将打开一个新对话框以显示所有坐标系参数。



技巧

如果在打开数据集时激活背景地图，则该数据集的内容将自动重投影到Spherical Mercator以匹配背景地图。如果您希望查看其自身坐标系中显示的数据，请在打开源数据集之前使用工具> FME选项关闭背景地图。

高级练习

为什么不使用Reprojector（或CSMapReprojector）转换器尝试这个练习，而不是在导航器窗口中使用读模块/写模块参数？转换器应放在工作空间的哪个位置，为什么这很重要？

恭喜

通过完成本练习，您已学会如何：

- 使用坐标系参数重投影空间数据
- 查询Data Inspector中的要素以检查坐标系信息

模块复习

本单元向您介绍了数据转换的概念，并解释了如何使用FME Workbench而不仅仅是快速转换。

你应该从这个模块中学到什么

以下是本次课程要学习的重点：

理论

- 一个模式描述了一个数据集的结构，包括它的要素类型，属性和几何形状。
- 模式编辑是编辑目标模式的行为，以更好地定义转换所需的内容。
- 将源模式加入目标的行为称为模式映射。两种模式之间的差异导致结构转换。
- 内容转换是在转换期间修改数据内容。在FME Workbench中，使用称为转换器的对象执行数据转换。
- 组可以使用转换器的参数组，通过选项来创建。
- 将数据拆分成多个流会创建多个副本而不是数据划分。将多个流组合在一起组合数据而不是合并它。

FME技能

- 编辑写模块模式的能力
- 通过手动和通过转换器将读模块映射到写模块模式来重构数据的能力
- 能够在Workbench中定位转换器并使用它们来转换数据内容
- 能够在参数编辑器或转换器对话框中设置转换器参数
- 使用“分组依据”选项定义功能组的功能
- 使用Workbench重新投影数据的能力

进一步阅读

为了进一步阅读，为什么不在我们的博客上浏览标记为**Transformation**的文章？

Q + A答案

以下是本章问题的答案。

Flibble先生说.....

使用FME定义的最常见格式转换是从Esri Shapefile到哪种格式？
1. Esri Geodatabase
2. AutoCAD DXF / DWG
3. Google KML
4. Esri Shapefile

是的！最常见的转换来自Esri Shapefile TO Esri Shapefile！它只是证明有多少人使用FME进行数据转换而不是格式转换。

Vector小姐说.....

哪三种颜色代表检查过，需要检查和取消设置变压器对象的参数？
1.蓝色，黄色，红色
2.绿色，黄色，红色
3.红色，绿色，蓝色
4.绿色，蓝色，黄色

Vector小姐说.....

如果我使用转换器来消除多边形边界中的不规则性（如自相交环），那么它是什么类型的转换？
1.属性的结构转换
2.几何的结构转换
3.属性的内容转换
4.内容几何的转换

Vector小姐说.....

您认为以下哪种转换器是“基于组”的？
1. StringFormatter
2. Clipper
3. Rotator
4. AttributeRounder

StringFormatter，Rotator和AttributeRounder一次可以在一个要素上运行。例如，您可以独立于任何其他功能旋转一个功能。但是，Clipper只能在一组要素上运行。“Group-By”参数是识别基于组的要素器的一个巨大线索！

练习：投票分析项目

练习**7**	投票分析项目
数据	选举映射（GML） 选举统计（Microsoft Excel）
总体目标	映射投票模式的统计
演示	数据转换
启动工作空间	C：\ FMEData2018 \工作空间\ DesktopBasic \转型-EX7-Begin.fmw
结束工作空间	C：\ FMEData2018 \ Workspaces \ DesktopBasic \ Transformation-Ex7-Complete.fmw C：\ FMEData2018 \ Workspaces \ DesktopBasic \ Transformation-Ex7-Complete-Advanced.fmw

在地面维护项目中，市政选举官员已经听说过您的技能，并要求帮助确定上次选举中投票率低的投票分区，或选民难以理解该过程的分区。

他请求您的帮助，并建议您在Google地球中显示结果，以便员工无需使用完整的GIS系统即可查看结果。

1）检查数据
启动FME Data Inspector并打开我们将使用的两个数据集：

读模块格式	**GML**（地理标记语言）
读模块数据集	C:\FMEData2018\Data\Elections\ElectionVoting.gml

读模块格式	**Microsoft Excel**
读模块数据集	C:\FMEData2018\Data\Elections\ElectionResults.xls

请注意，两个数据集都有一个Division属性，用于标识每个投票分区（区域）。Excel数据是非空间的，但具有一组其他投票属性：

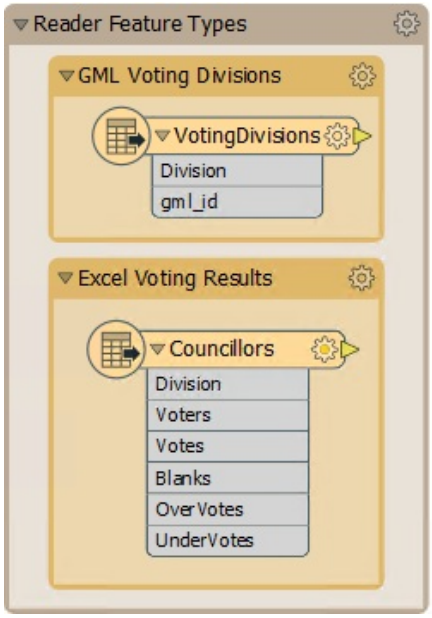
- 选民：登记选民人数
- 投票：投票的选民人数
- 空白：投空白票或废弃票的选民数量

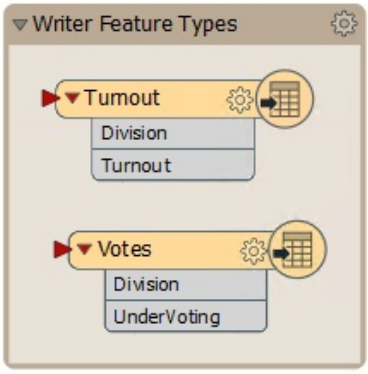
- **OverVotes**：投票选举太多候选人的选民人数
- **UnderVotes**：未投票数

OverVotes和UnderVotes属性指示了对投票过程的了解程度。每位选民最多可投票给10名候选人（30名）。

OverVotes是那些投票选出超过十名候选人的选民。UnderVotesUnderVotes是本来可以投出但没有投出的票数; 例如，选民只投了四个候选人而不是十个，有六个未投。

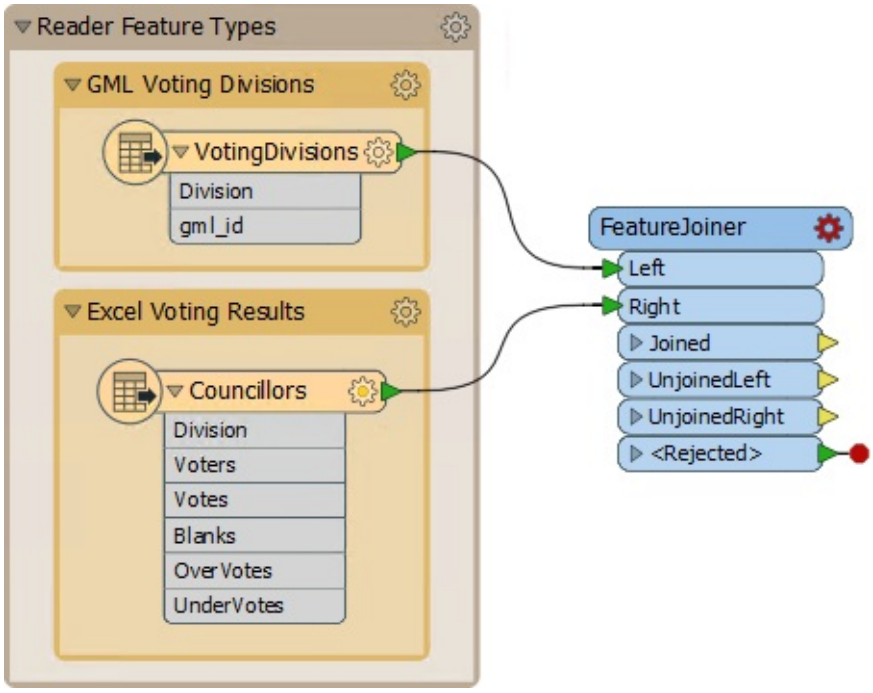
2）启动**Workbench**
启动Workbench并打开起始工作空间。它已经添加了读模块和作者来处理数据; 我们需要做的就是进行转换：





3）添加**FeatureJoiner**
第一个任务是将统计选举数据加入到实际要素中。我们将使用FeatureJoiner转换器来执行此操作。FeatureJoiner是一种连接或合并要素的方法。在这种情况下，我们将选举结果记录合并到选举边界要素上。

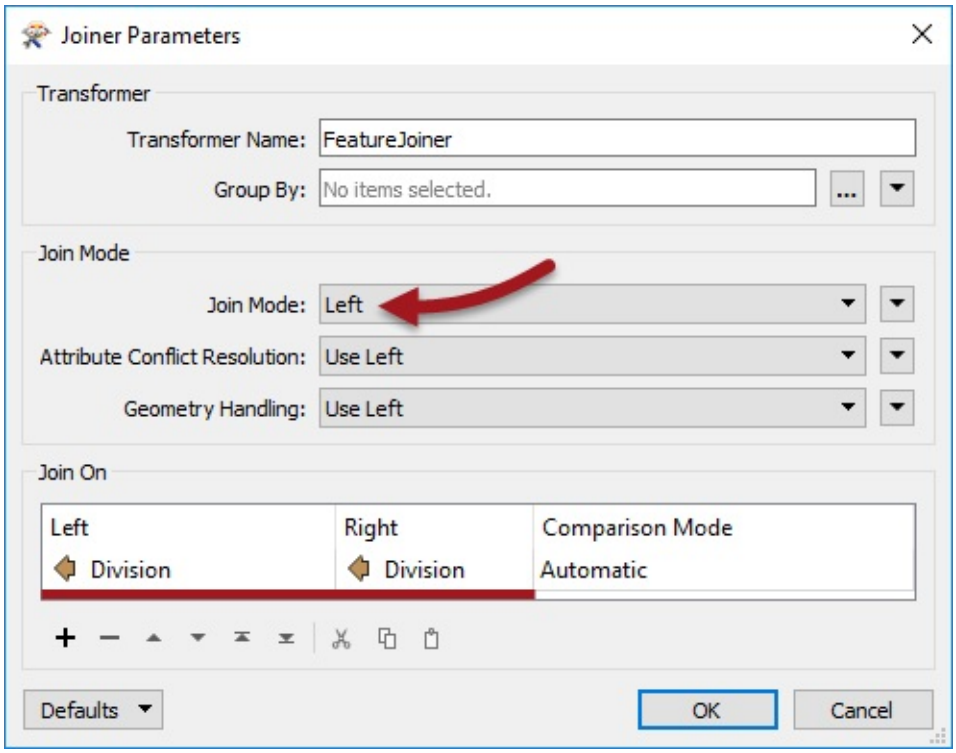
添加FeatureJoiner转换器。将VotingDivisions数据连接到Left端口，将Councilors（结果）数据连接到Right端口：



新东西
FeatureJoiner是FME2018的新转换器，旨在最终取代FeatureMerger。

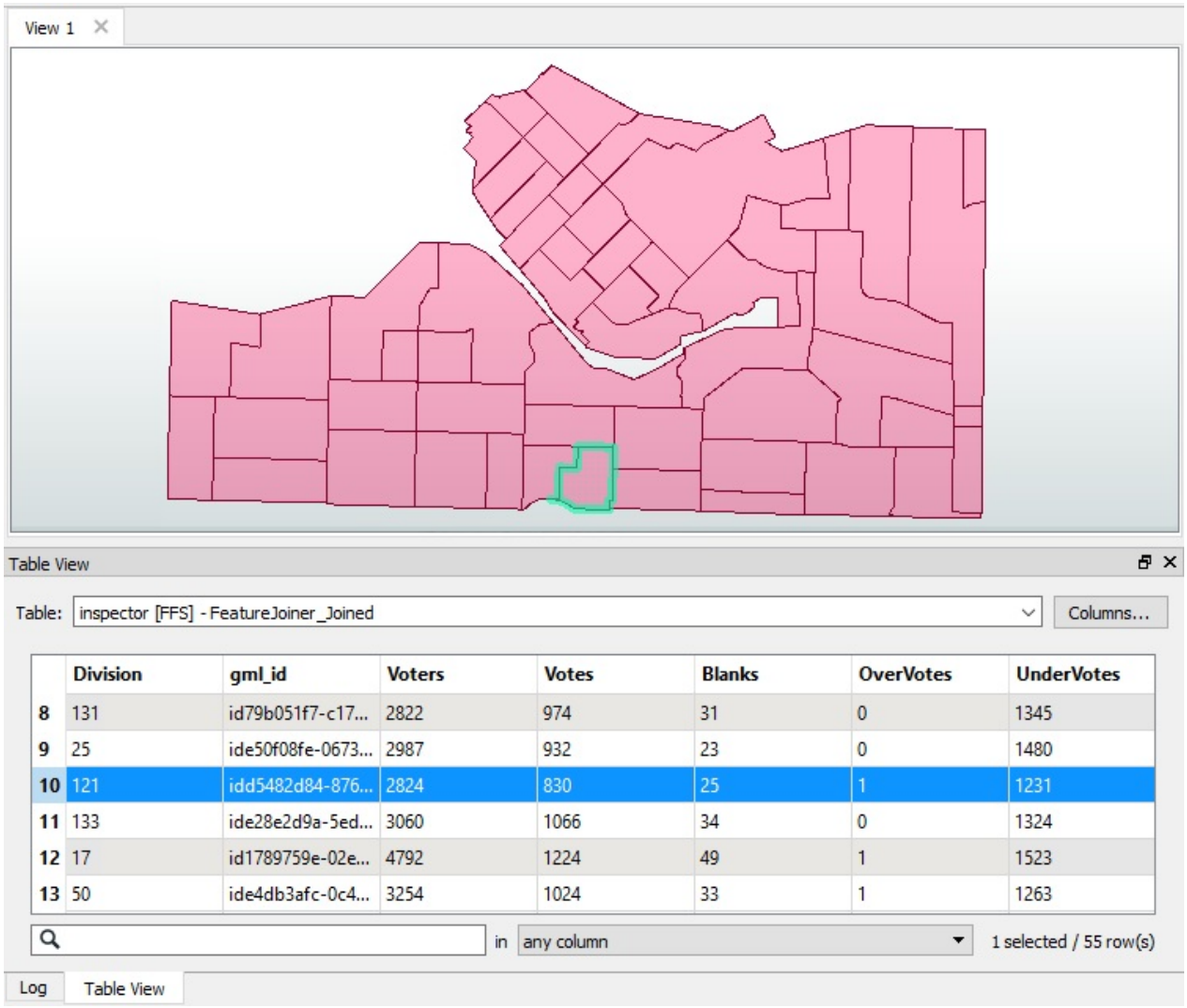
4）设置参数
查看FeatureJoiner参数。因为我们想要所有的投票分区要素，我们将进行左连接; 因此将Join Mode设置为Left。

对于“左”和“右”连接字段，单击该字段并从下拉列表中选择“Division”属性。此属性是我们加入数据的公共密钥：



5）添加**Inspector**转换器
在FeatureJoiner：Joined输出端口之后添加**Inspector**转换器。运行工作空间，忽略报告意外输入的任何警告或日志消息。

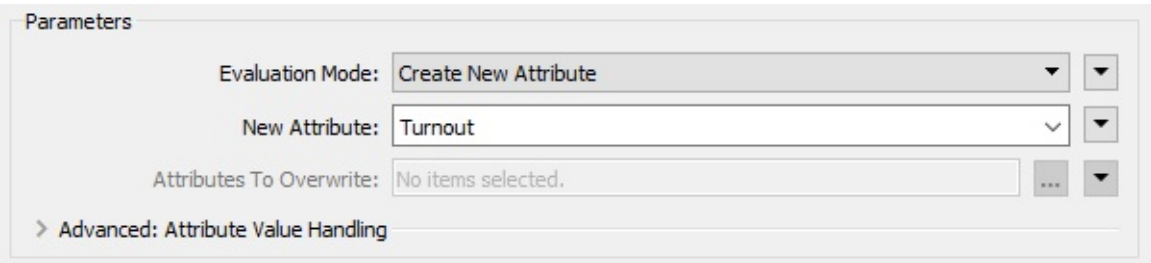
在FME Data Inspector中检查数据，以确保所有分区多边形现在包含从Excel电子表格复制的一组属性数据：



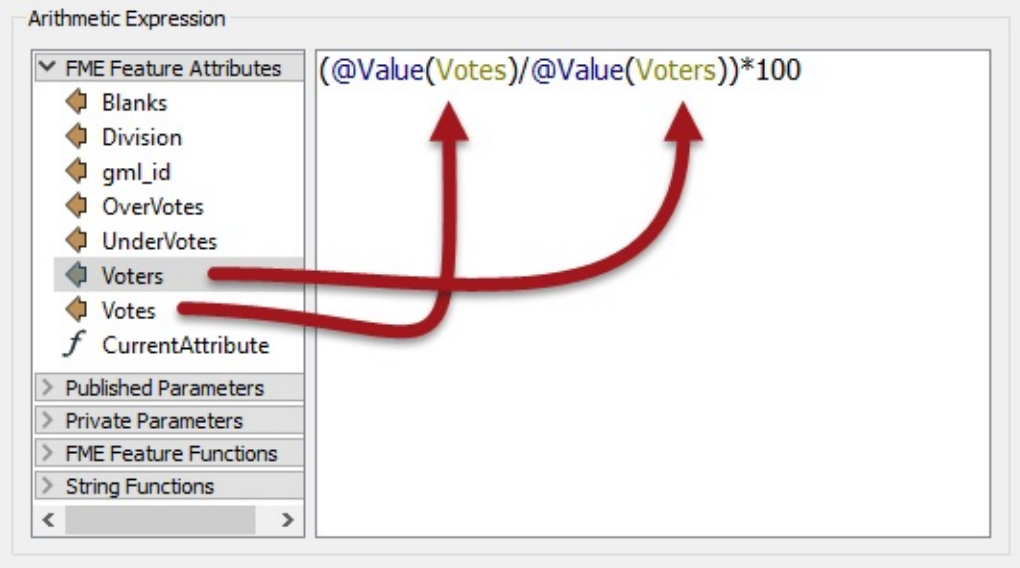
	Division	gml_id	Voters	Votes	Blanks	OverVotes	UnderVotes
8	131	id79b051f7-c17...	2822	974	31	0	1345
9	25	ide50f08fe-0673...	2987	932	23	0	1480
10	121	idd5482d84-876...	2824	830	25	1	1231
11	133	ide28e2d9a-5ed...	3060	1066	34	0	1324
12	17	id1789759e-02e...	4792	1224	49	1	1523
13	50	ide4db3afc-0c4...	3254	1024	33	1	1263

6）添加**ExpressionEvaluator**
现在我们有了需要的数字，我们就可以开始计算一些统计数据了。为此，我们首先使用ExpressionEvaluator转换器来计算每个分区的选民投票率百分比。

在FeatureJoiner之后放置一个ExpressionEvaluator转换器 - 将它连接到FeatureJoiner:Joined输出端口。查看转换器的参数。将New Attribute设置为Turnout（以匹配我们在目标模式上的内容）：



在表达式窗口中，将表达式设置为：




```
(@Value(Votes)/@Value(Voters))*100
```

您无需键入此内容 - 可以通过双击左侧列表中的该属性来获取@Value（投票）和@Value（选民）部分。

如果您愿意，可以重新连接Inspector并重新运行转换，以查看结果。

7）添加**ExpressionEvaluator**
使用类似的技术，添加第二个ExpressionEvaluator来计算每个选民的UnderVotes数量，并将其放在与输出模式匹配的属性中（例如UnderVoting）。表达式将是这样的：

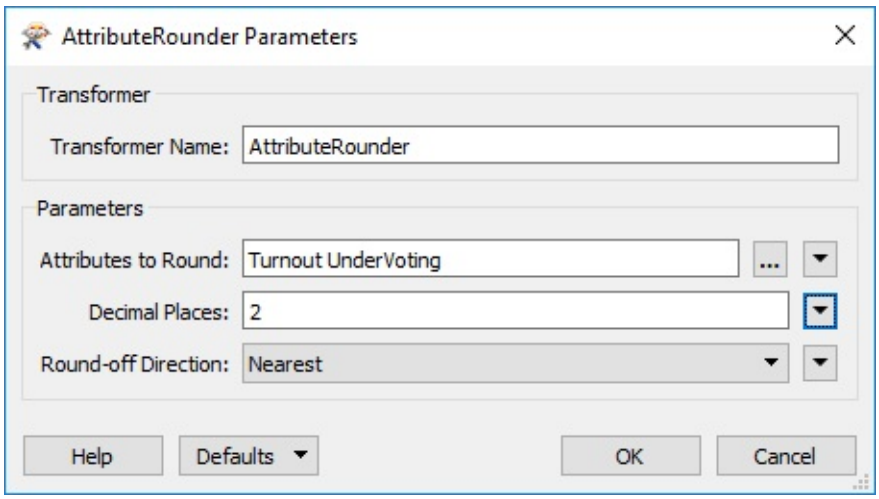
```
@Value(UnderVotes)/@Value(Voters)
```

注意： 这不是百分比，就像之前的计算一样。

随意在两个ExpressionEvaluator转换器周围添加书签。

8）添加**AttributeRounder**
将我们的统计数据计算到13位小数或更多位置有点过分了。我们应该截断这些数字。为此，在第二个ExpressionEvaluator之后放置一个AttributeRounder转换器。

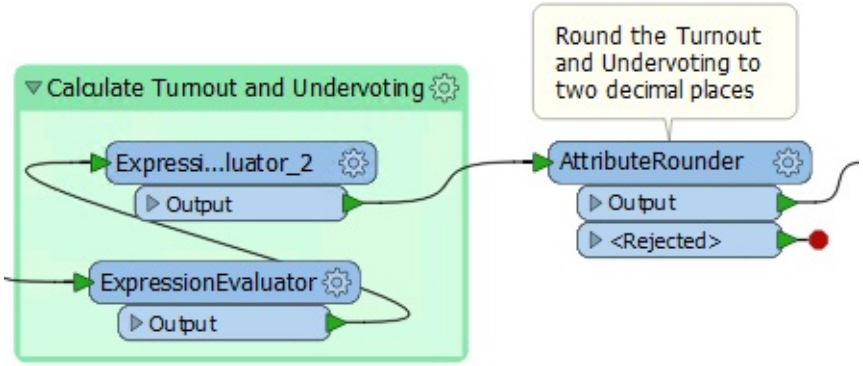
对于参数，在“要舍入的属性”下，选择新创建的Turnout和UnderVoting属性。将小数位数设置为2：



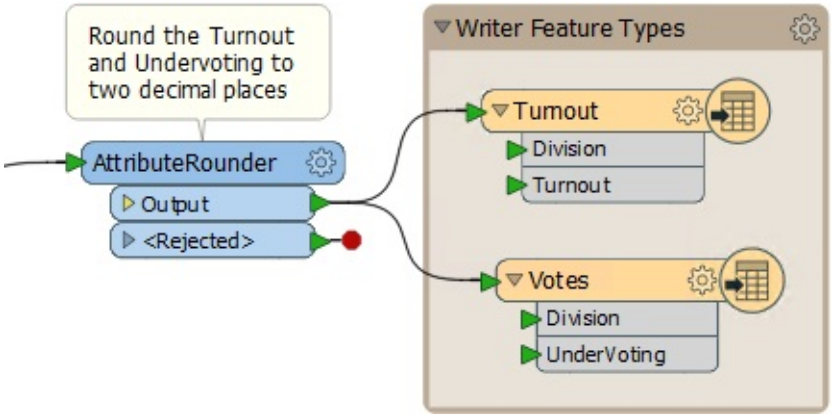
再次，如果您愿意，运行工作空间以检查结果。

9）添加注释
将标签添加到AttributeRounder是没有意义的，因此右键单击转换器并选择Attach Annotation选项。

这样做会为转换器添加标签。编辑内容，例如“将道岔和投票结果舍入到小数点后两位”：



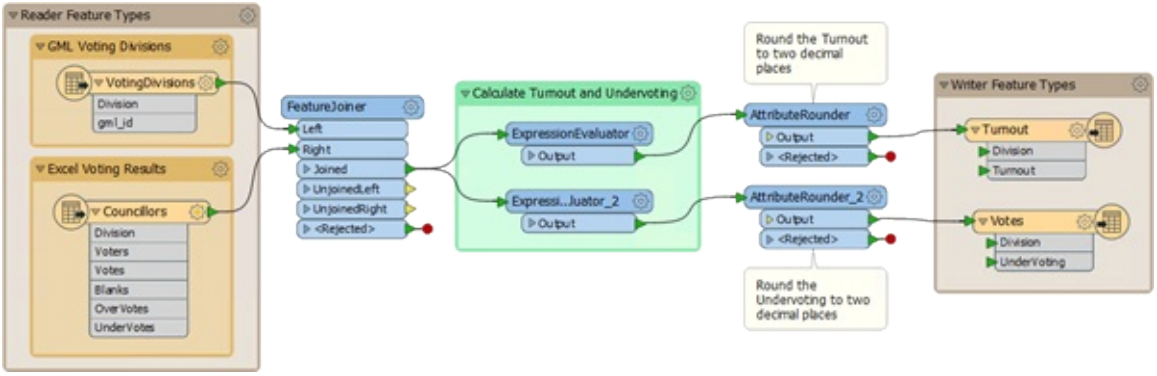
10）连接模式
最后一步让我们将AttributeRounder连接到输出模式。只需从AttributeRounder连接到两种写模块要素类型：



运行工作空间并检查Google地球中的输出，以证明它具有正确的属性并位于正确的位置。

技巧

所以这显然是一种并行数据流，但一直持续到最后一步。另一种布局是在FeatureJoiner之后拆分数据，如下所示： 输出没有区别，因此唯一的考虑因素是哪个更容易创建，哪个更好。使用的主要方法应该在两个方面都胜出。



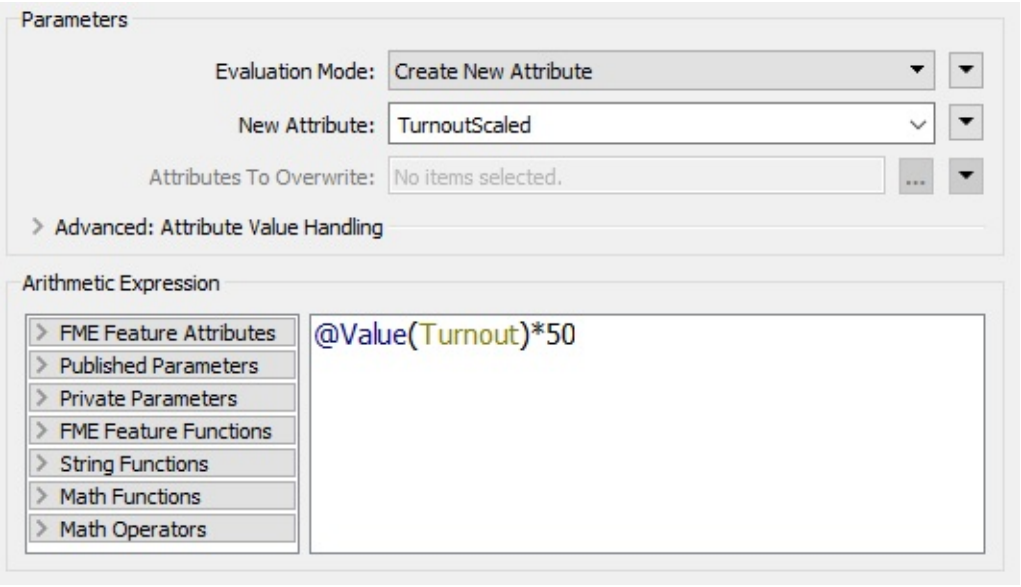
高级练习

该项目已完成，但输出非常简单。改善结果的外观会好得多，有几种方法可以用KML来完成。 我们可以根据他们的投票率/总票数简单地对投票部门进行着色，但更令人印象深刻的方法是使用三维块。 按照以下步骤在输出KML数据集中创建三维形状...

10）添加ExpressionEvaluator

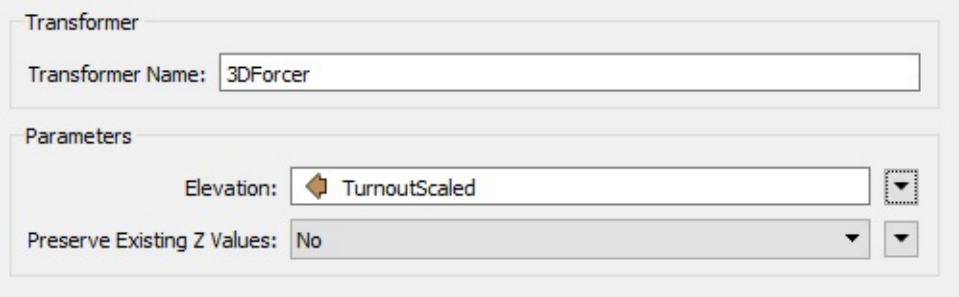
每个块的高度应该与该分区的道岔成比例。但是，为了看到差异，垂直比例需要夸大一些。

在AttributeRounder和Turnout要素类型之间放置一个ExpressionEvaluator。设置参数以将Turnout属性乘以值50.将其放入名为TurnoutScaled的新属性中。



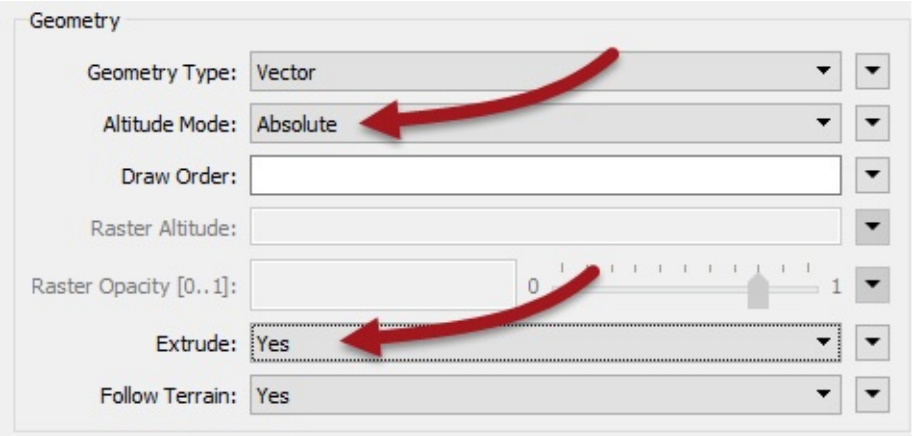
11）添加3DForcer

在新的ExpressionEvaluator之后添加一个3DForcer转换器。这会将要素提升到所需的高度。在参数对话框中，将高程设置为“属性值”>“TurnoutScaled”。

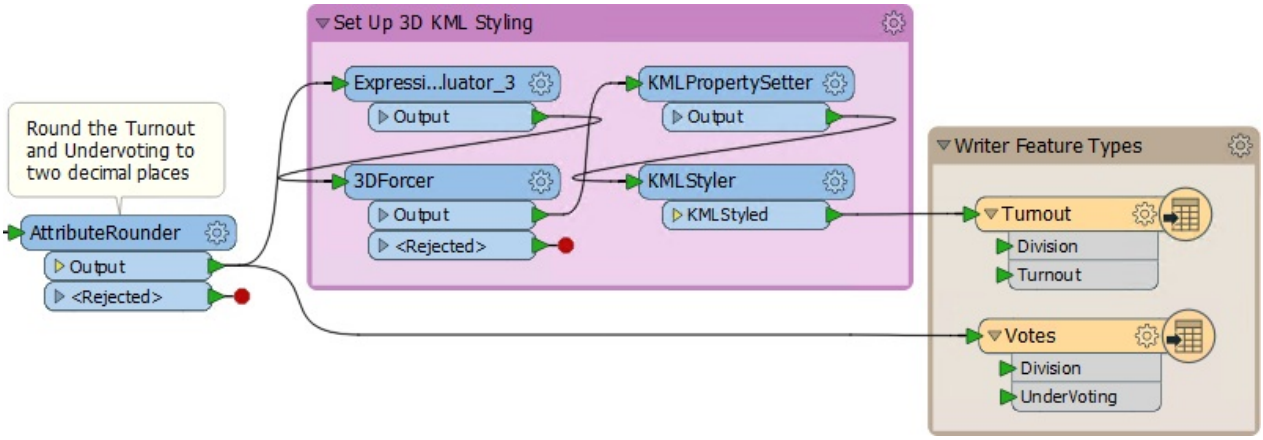


12）添加**KMLPropertySetter**
在3DForcer之后添加KMLPropertySetter转换器。该转换器允许我们在输出中设置3D块。设置几何东西参数如下：

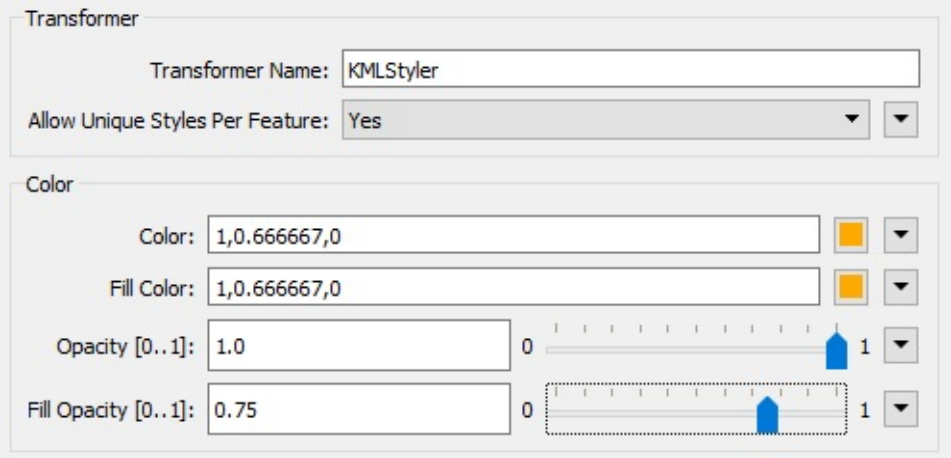
- 高度模式:绝对值
- 挤压：是的



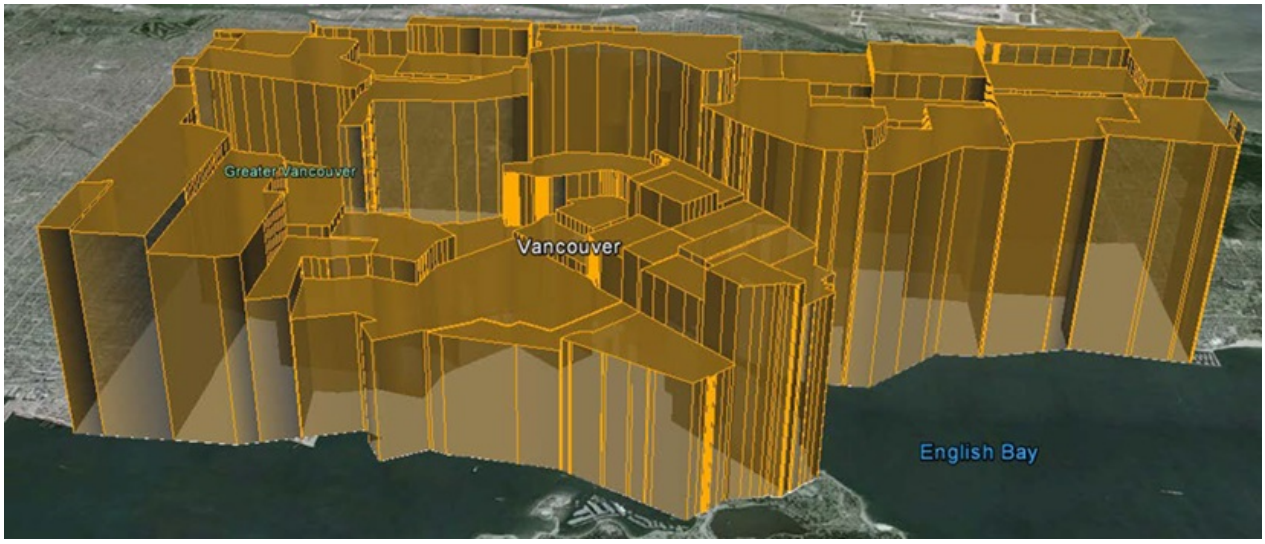
13）添加**KMLStyler** 最后，添加一个KMLStyler转换器。工作空间现在看起来像这样：



检查参数。选择颜色并填充要素的颜色。将填充不透明度增加到0.75左右。



保存并运行工作空间。在Google地球中，输出现在应如下所示：



这些3D块将向用户显示城市选民投票率高/低的情况。

如果您愿意，请重复这些步骤，为UnderVoting统计数据提供3D表示。

恭喜
通过完成本练习，您证明了您知道如何： • 使用转换器进行内容转换（ExpressionEvaluator，AttributeRounder，3DForcer） • 使用转换器参数创建与写模块模式匹配的属性 • 在单个工作空间中使用多个转换器流 您还学会了如何： • 使用公共密钥合并多个数据流（FeatureJoiner） • 使用FME的内置数学编辑器对话框 • 使用转换器为输出要素设置符号系统（样式）

工作空间设计

FME中的基本工作空间构建并严格按照ETL：Extract-Transform-Load的概念进行布局

但是，还有多种其他方法可用于构建更高级的工作空间，以及以独特方式通过工作空间引导数据流。

这些技术的一些示例用途可能是：

- 一次设计一个小的部分来设计大型的工作空间
- 在单个工作空间内读取多种格式的数据
- 在写入数据集之后执行操作
- 使用存储在Web服务上的数据
- 测试运行工作空间的各个部分

工作空间原型

在传统意义上，原型设计意味着创建一个不完整的应用程序，作为评估项目可行性的一种方法。

在这里，我们将扩展定义以表示如何逐步构建复杂的FME项目; 从一个空的工作空间开始，逐个构建它，以提供与最终规范相匹配的结果。

我们将介绍用于逐步构建工作空间的技术，以及如何处理转换器拒绝的数据。

增量发展

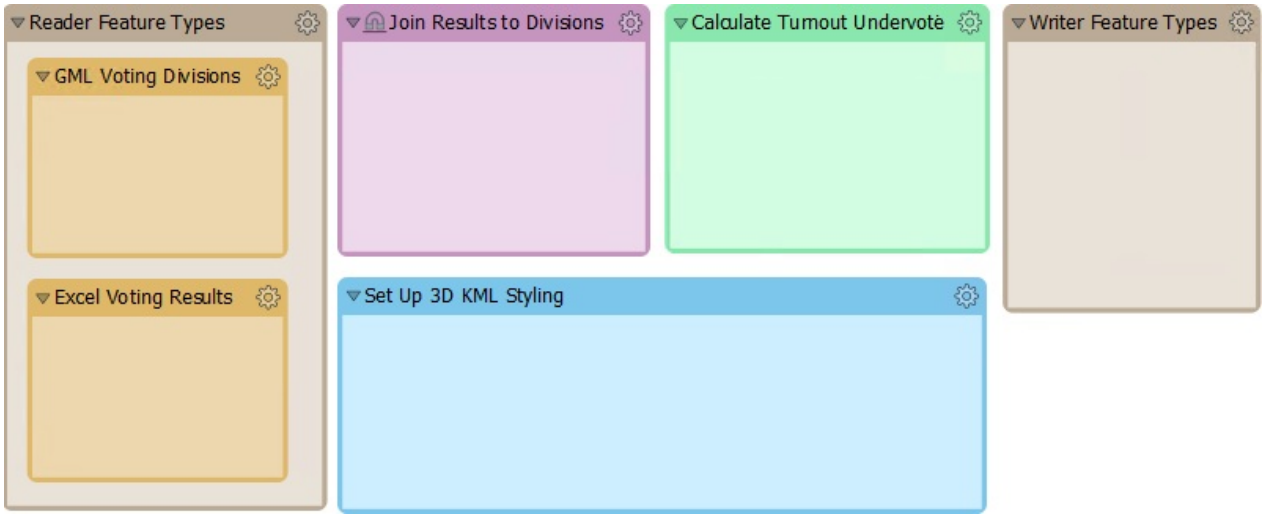
FME工作空间的关键开发技术是增量更新。

这种技术的步骤是：

- 把你的项目规划成一系列小的部分，每一部分都可以放在FME的书签里
- 在FME Workbench中设计并实现一个部分。理想情况下，它应该在3-10个转换器之间。
- 完成后立即测试每个部分。这样，您可以在早期识别问题，并更容易识别问题，因为在任何增量中只进行了一些更改。
- 重复此过程，保存工作空间并在添加新部分时对其进行测试

虽然3-10转换器的范围是任意数字，但是添加的转换器越多，识别任何问题的根源就越困难。超过十个转换器是您应该考虑将该过程切割成更小的部分的点。

在这里，作者通过将其作为一组书签放在画布上来计划他们的工作空间：



现在，作者可以一次完成并测试每个部分，同时牢记整体目标。

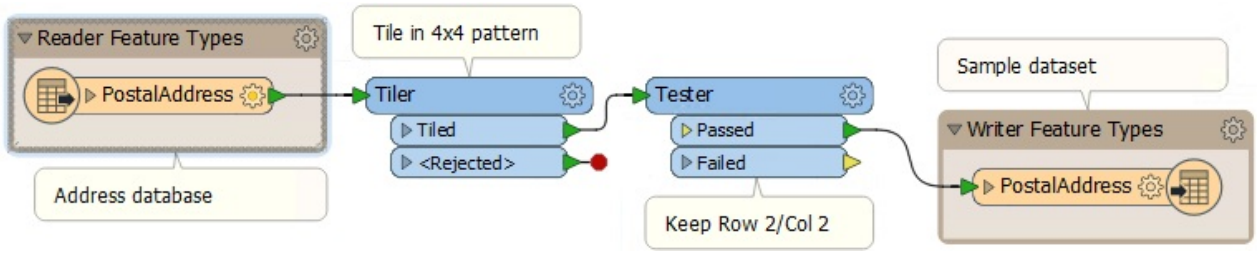
警告
开始开发工作空间并忘记保存工作空间是非常容易的!一旦第一次保存工作空间，FME就会保留恢复文件，但在此之前您将面临无法挽回的损失风险。

源数据

当FME项目庞大且复杂时，源数据可能也会很大且复杂。因此，当以小增量创建工作空间，依次测试每个部分时，最好避免使用整个数据集。

最好使用源数据样本进行测试。实际上，最好创建一个小的数据样本 - 从源中提取数据并写入FFS等中性格式 - 而不是为每次测试运行随机抽样数据。

采样对数据库特别有用，因为它还避免了等待网络流量和数据库响应的问题。



这里工作空间作者通过从数据库中读取，将其拆分为瓦片并仅将一个瓦片写入FFS格式来提取源数据的一部分。这一个瓦片可用于以代表整个源数据库表的方式对解决方案进行原型设计。

要使用的另一个转换器是Sampler，尽管它选择的要素在空间上不会相邻。

版本控制

对工作空间进行一组增量更改时，只能在单个工作空间文件上进行操作。但是，这有各种各样的问题：

- 无法轻松跟踪故障，因为没有记录已更改的内容以及何时更改
- 为不同的平台创建不同的版本并不容易
- 如果工作空间文件丢失或损坏，则整个项目将丢失

因此，最好保留版本化的工作空间，其中为每组修订保留不同的副本。可以在文件系统中手动执行此预防措施，也可以使用Git等存储库工具。

事实上，保留和版本化与FME项目相关的所有材料是一个好主意，包括：

- 工作空间文件
- Python文件
- 日志文件
- 来源数据集

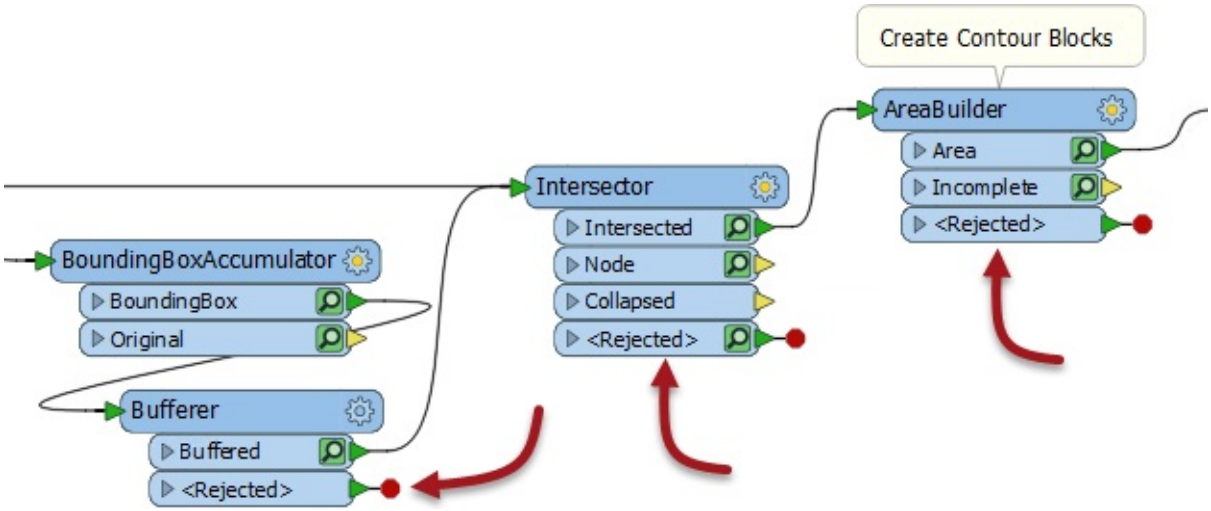
最好不要存储任何个人信息或包含密码的信息。此外，无需存储临时文件。

技巧
FME Server内置了版本控制系统。即使您没有FME Server许可证，您仍然可以将其安装用作工作空间和相关文件的存储库系统。

被拒绝的端口

任何工作流程的一个重要部分是处理未能处理的数据; 例如，将没有几何对象的要素发送到基于几何对象的转换器（如AreaBuilder）。

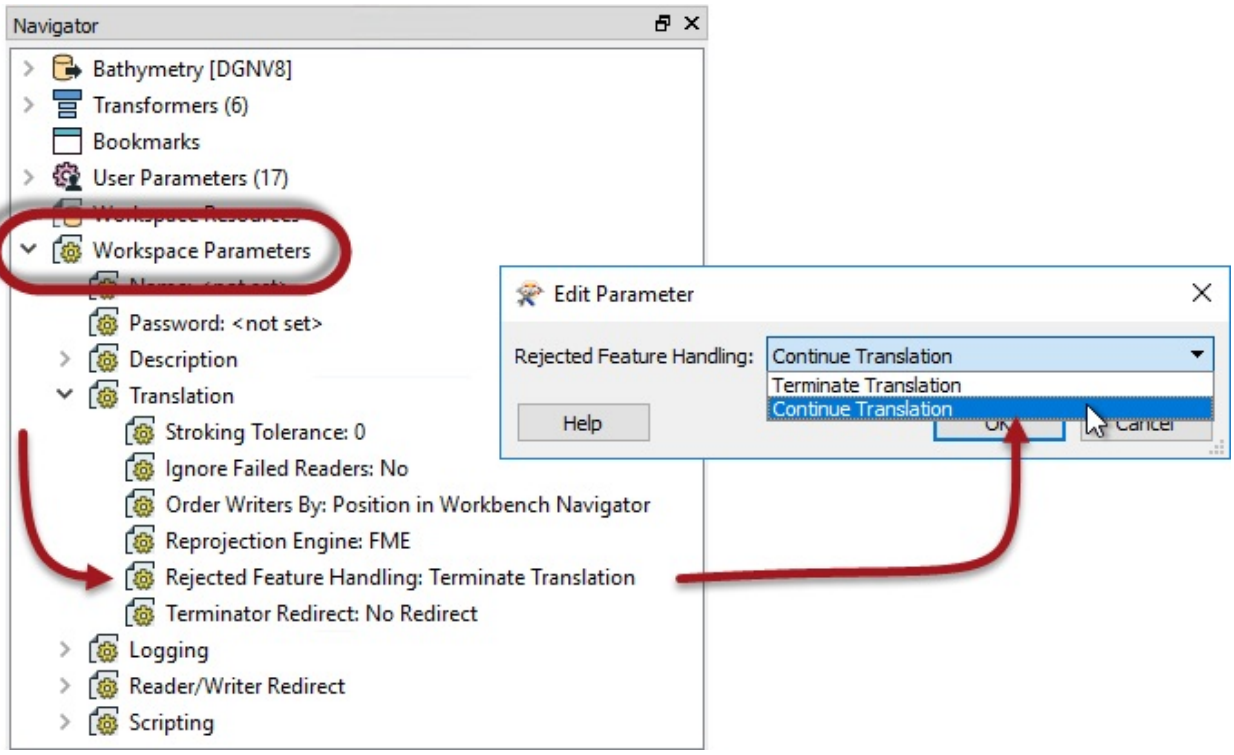
FME通过<Rejected>端口输出数据来处理此类故障，这些端口可在许多转换器上找到：



存在一个参数来控制<Rejected>端口的操作，并为工作空间作者选择要采取的操作。

被拒绝要素处理

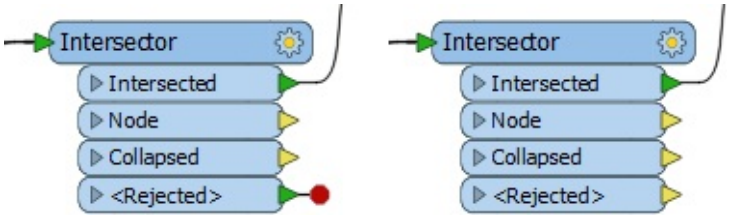
可以在“工作空间参数”下的“导航”窗口中找到控制被拒绝要素处理的参数：



这两个选项是终止转换和继续转换。

当参数设置为terminate时，通过<Rejected>端口退出的要素会导致转换停止。为了在视觉上表示这一点，<Rejected>端口上有一个红色标记。

当参数设置为continue时，无论退出<Rejected>端口的要素有多少，转换都将继续。在这种情况下，红色标记被删除：



在终止模式下，被拒绝的要素将被写入日志窗口，并显示错误消息：

```
The below feature caused the translation to be terminated
```

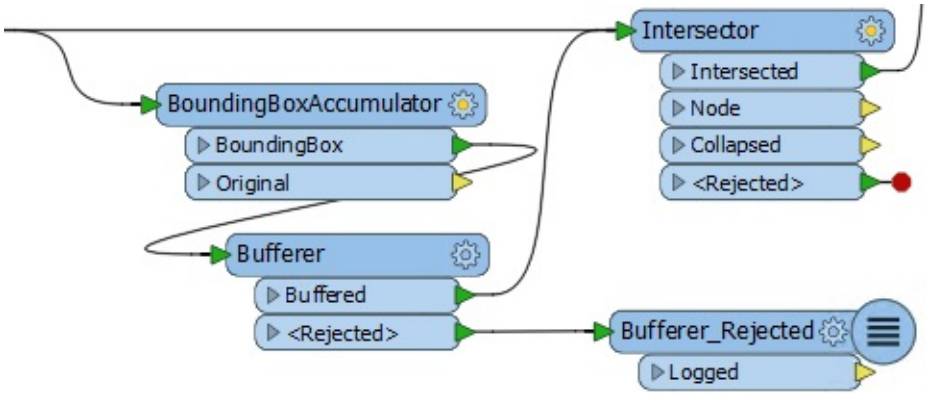
还会有一个与转换器有关的错误消息：

```
Intersector_<Rejected>(TeeFactory): Intersector_<Rejected>:
Termination Message: 'Intersector output a <Rejected> feature.
```

此错误很有用，因为它告诉作者哪个转换器出现故障。

混合模式

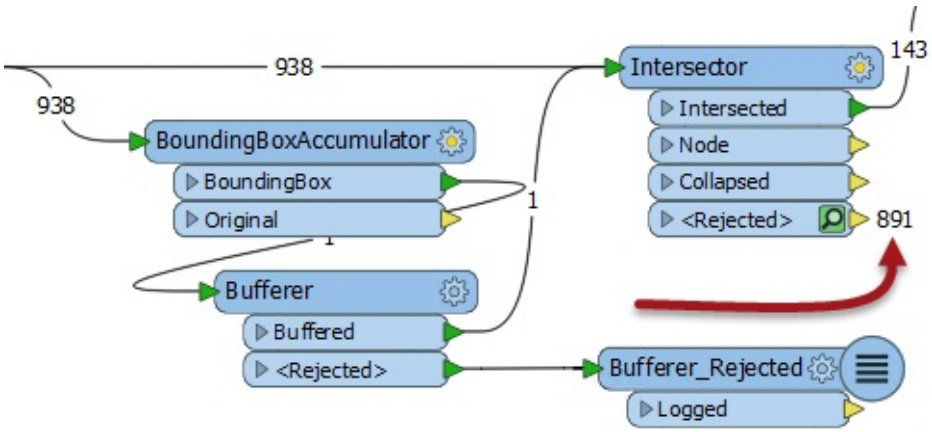
在终止模式下，如果<Rejected>端口连接到另一个对象，则被拒绝的要素不会导致转换终止：



简而言之，作者可以创建混合模式，其中一些转换器在拒绝某个要素（上面的交叉点）时终止，但其他转换器将以另一种方式处理该要素（Bufferer）。这样，作者可以尝试处理预期的被拒绝要素，但如果确实存在意外故障则停止转换。

要素计数和检查

在继续模式下，退出<Rejected>端口的要素将被计算并保存以供检查：

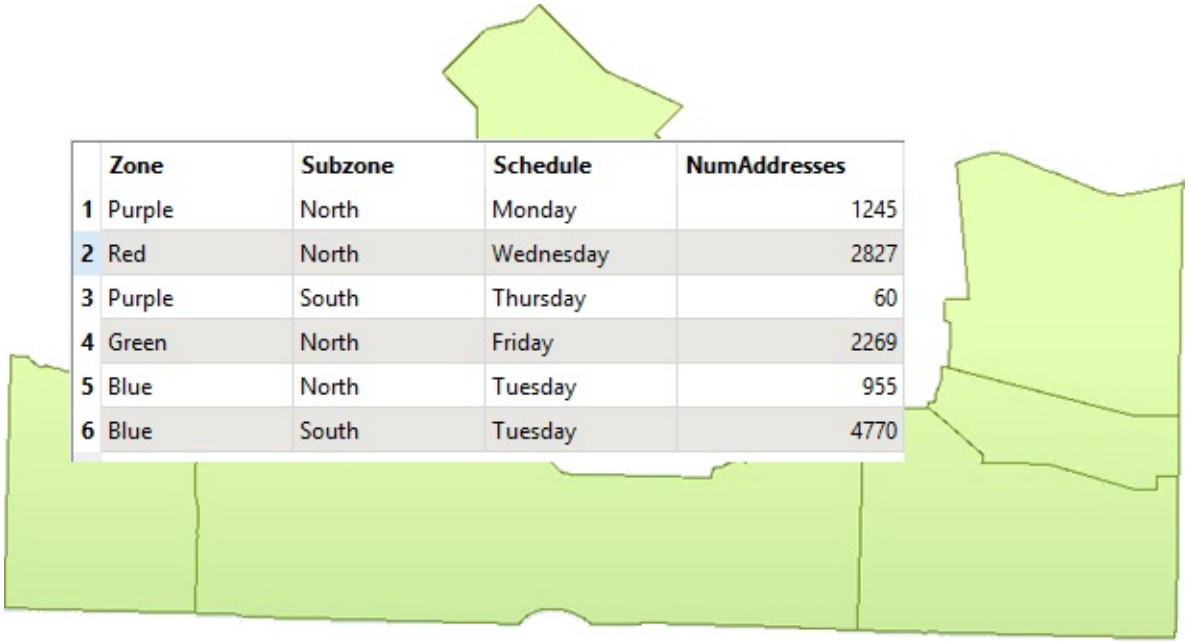


即使没有连接Logger 或其他转换器，也会保存要素以供检查。该数字告诉我们有多少要素被拒绝，并且可以单击绿色图标来检查数据。

练习：住宅垃圾收集区

练习**1**	住宅垃圾收集区
数据	地址（Esri地理数据库），区域（MapInfo TAB）
总体的目标	为住宅垃圾收集创建边界
演示	工作空间原型设计
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Design-Ex1-Begin.fmw
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Design-Ex1-Complete.fmw

城市维护部门有一个垃圾收集计划数据集，用于指定居民在特定的一天收集::



	Zone	Subzone	Schedule	NumAddresses
1	Purple	North	Monday	1245
2	Red	North	Wednesday	2827
3	Purple	South	Thursday	60
4	Green	North	Friday	2269
5	Blue	North	Tuesday	955
6	Blue	South	Tuesday	4770

然而，因为人口结构和分区的变化，他们决定应该绘制新的边界。

您的任务是使用FME创建新边界。您必须创建五个多边形，彼此相邻，并且每个多边形的居民数量大致相同。分析将基于城市的地址数据库。将根据其所属的区域类型估算每个地址的居民数量：

- 独户住宅：2名成人
- 两户住宅：4名成人
- 多户住宅：12名成人
- 综合开发区：8名成人
- 商业地产：1名成人

输出格式应为OGC GeoPackage。

要开发此工作空间，有必要考虑可能需要哪些不同的步骤。然后我们可以创建带有书签的部分，并在进行过程中填充它们。

1）规划工作空间
让我们一起规划这个工作空间。

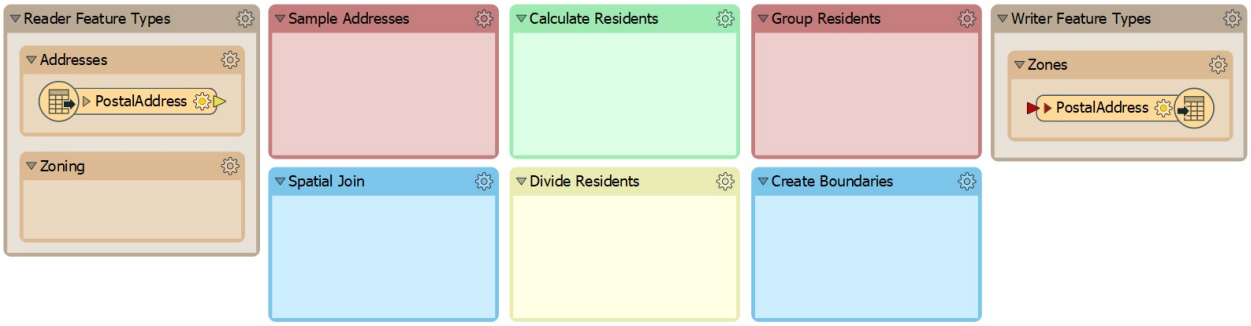
我们需要读取地址数据（或其示例）并将输出写入OGC GeoPackage。我们需要知道每个地址落入哪个区域类型，这需要分区数据和转换器来执行空间连接。

我们还需要根据区域类型创建居民计数，然后将居民划分为五个不同的区域。最后，我们需要将地址分组，并在它们周围形成边界形状。

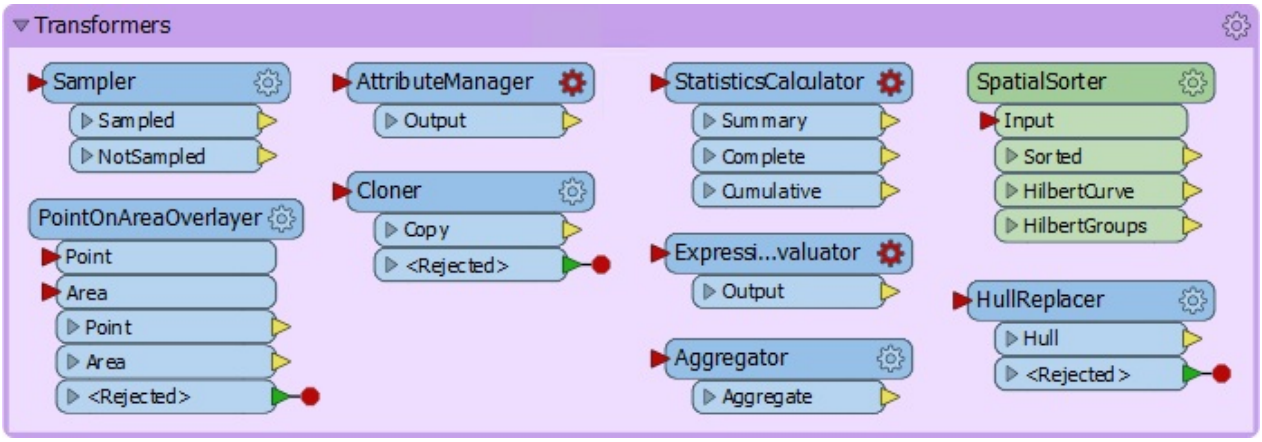
简而言之，我们需要这套行动：

- 读取/采样地址数据库
- 读取分区数据
- 创建空间连接
- 计算居民数量
- 将居民分为五组
- 将地址聚合到他们的组中
- 创建边界形状
- 写入OGC GeoPackage

因此，选择“文件”>“打开”，然后选择开始工作空间（C：\ FMEData2018 \ Workspaces \ DesktopBasic \ Design-Ex1-Begin.fmw）。它已经有一组书签来表示要执行的这些步骤，但是，到目前为止，我们无法确定哪些部分会更大，因此所有书签的大小都相同：



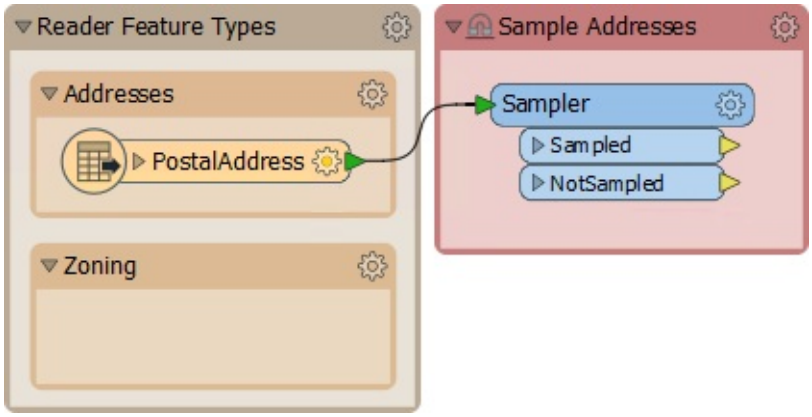
您还可以找到包含练习所需的所有转换器的书签：



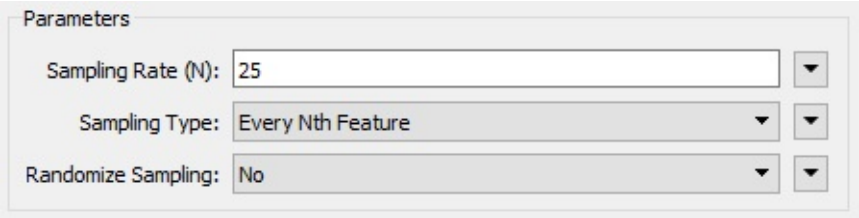
或者至少，这些是我们认为练习所需的转换器！

2）样本源数据
地址数据库中的要素比工作空间构建和测试需要的要多，所以让我们将其减少到更小的样本。

这里我们将使用Sampler转换器，而不是创建测试数据集。“转换器”书签中有一个采样器转换器，只需将该转换器移动到“示例地址”书签中，并将PostalAddress要素类型连接到它：



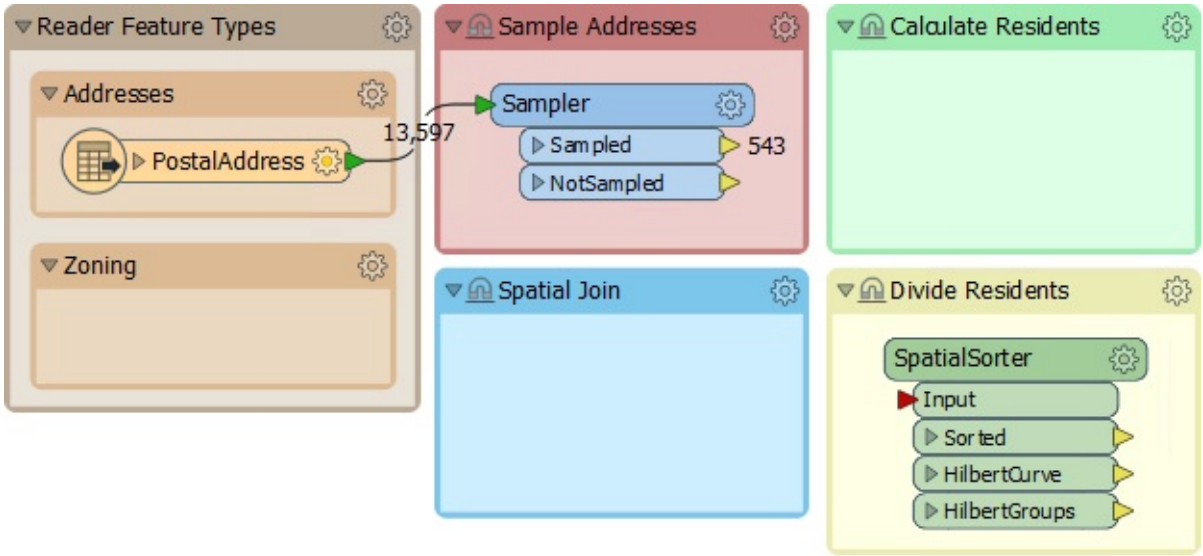
检查Sampler的参数。将其设置为每25个要素进行采样



添加一个Inspector转换器（到Sampler：Sampled输出端口）并运行工作空间以确保它正确地采样数据。在13,597个地址中，应该有543个选择使用。

3）将数据划分为组
在尝试将分区数据集添加到工作空间之前，让我们尝试从基本数据集创建组。我们可以使用FME Hub中的自定义转换器（称为SpatialSorter）来完成此操作。

因此，将SpatialSorter从“转换器”书签移动到“Divide Residents”书签：

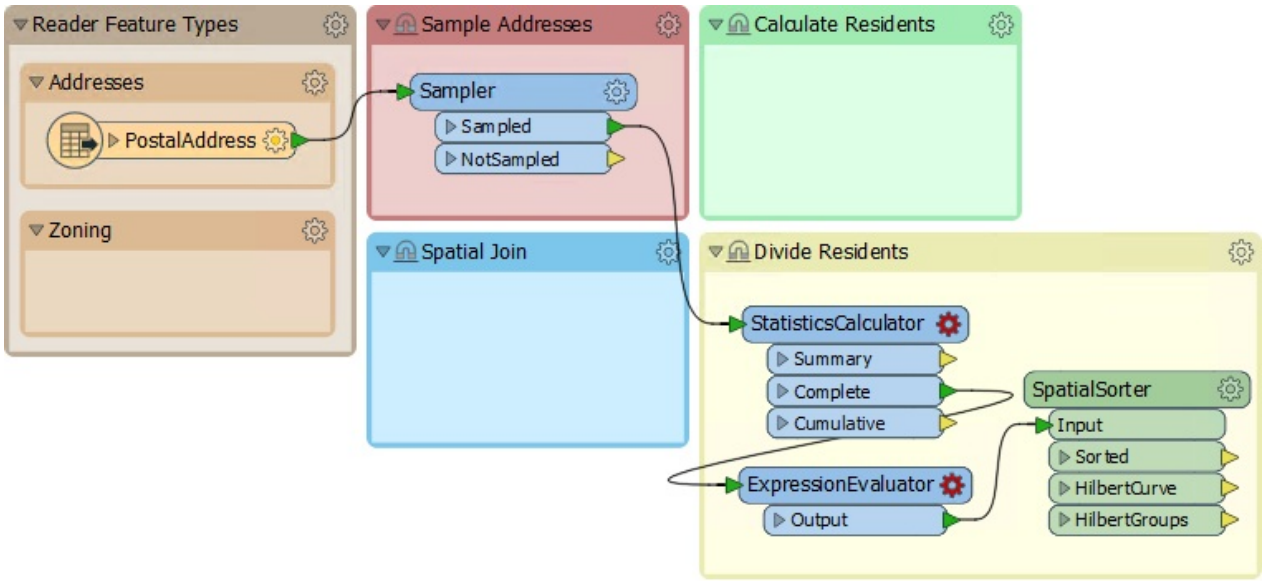


SpatialSorter在空间上对数据进行排序（因此在地理位置上的要素在排序的输出中更接近）并创建组。

检查此转换器的参数。请注意，group参数要求组大小，而不是组数。因此，我们需要计算分成五组时的地址数。

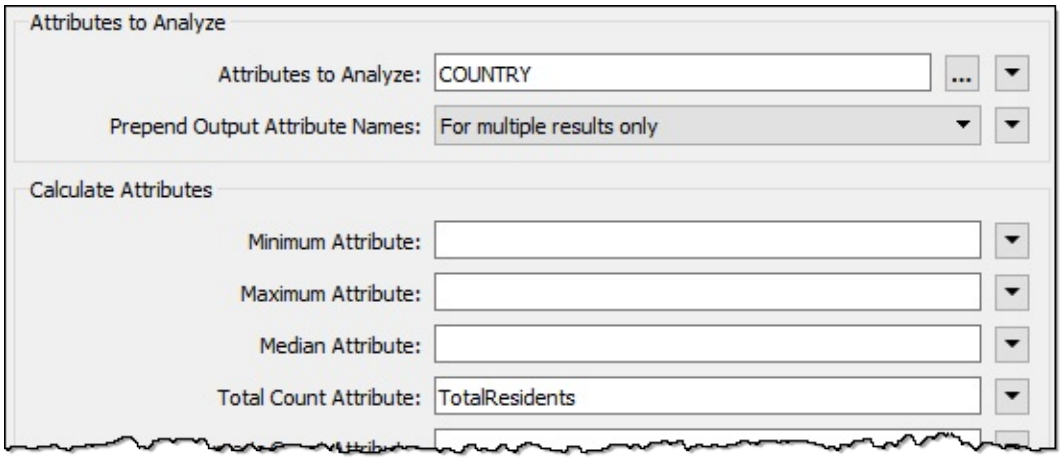
4）计算组大小
要计算每组的地址数，我们需要地址数，然后除以5。我们可以使用StatisticsCalculator和ExpressionEvaluator的组合来完成此操作。

因此，根据需要放大Divide Residents书签，并从“转换器”书签中移动StatisticsCalculator和ExpressionEvaluator转换器。将它们连接到Sampler：Sampled端口，如下所示：



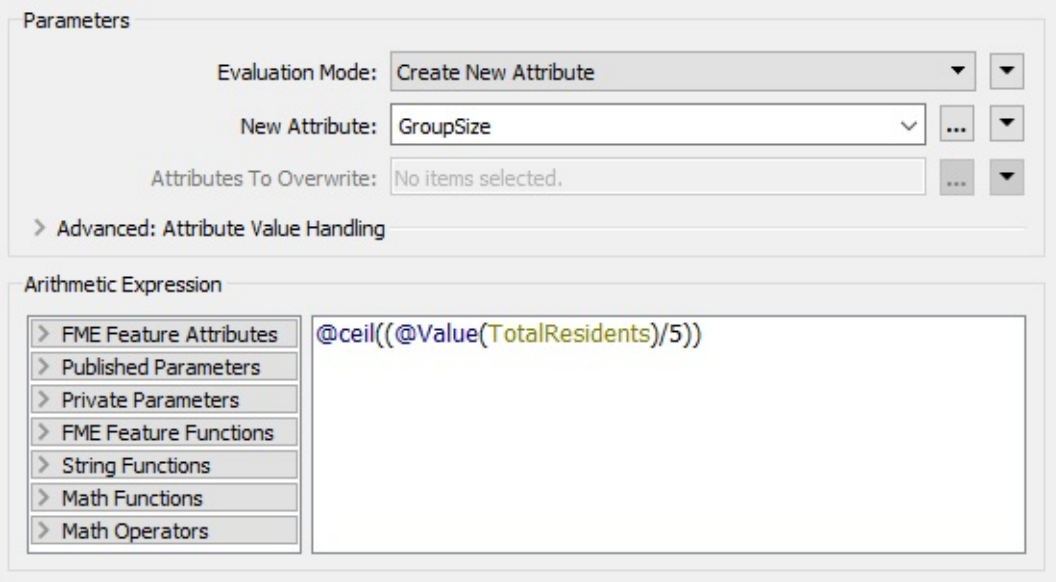
5）计算组大小
检查StatisticsCalculator的参数。此转换器将告诉我们有多少要素（总计数）。选择要分析的属性参数的属性。因为我们只想计算不创建真实统计的要素，所以它可以是您喜欢的任何属性。

在Total Count Attribute参数下，输入名称TotalResidents。删除其余的计算属性字段，以便不计算它们：



在ExpressionEvaluator中，在New Attribute参数中输入GroupSize。在算术表达式字段中输入表达式：

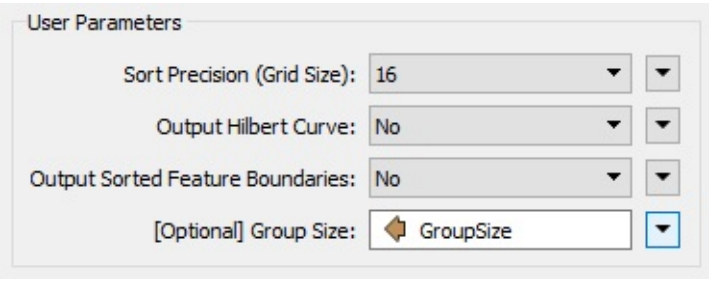
```
@ceil((@Value(TotalResidents)/5))
```



最简单的方法是双击ceil函数添加它，然后双击TotalResidents属性，最后手动添加/ 5部分。

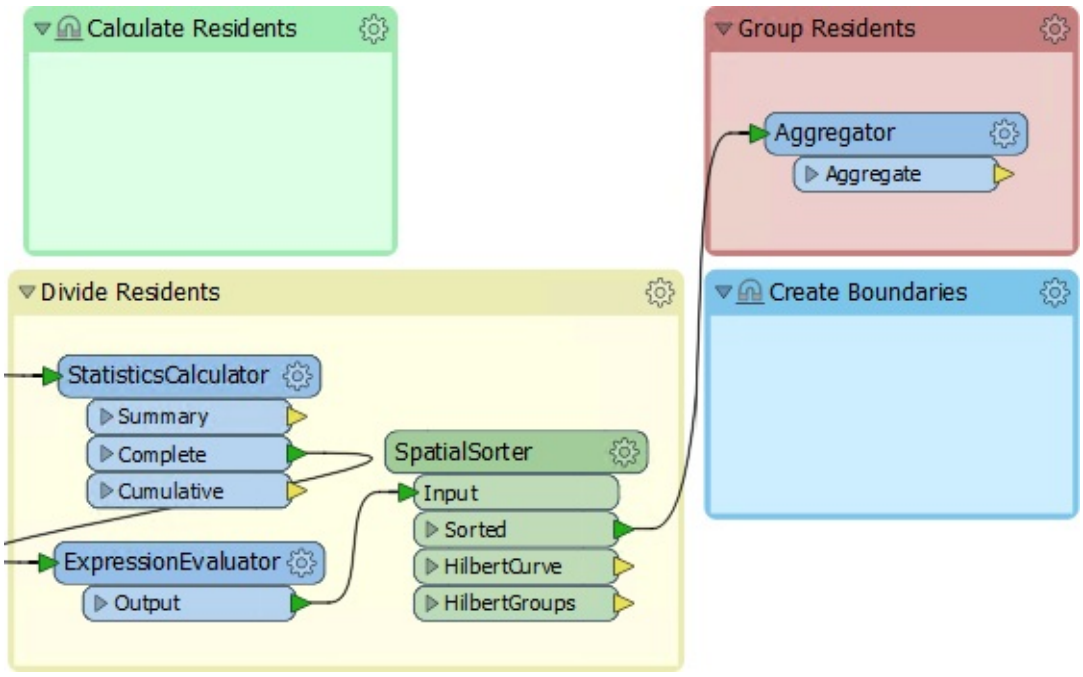
这个表达式将居民人数分为五组，四舍五入。舍入部分是必不可少的，它就是ceil函数的作用。将Inspector转换器添加到ExpressionEvaluator并运行转换以证明此部分有效。

6）分组居民
现在再次检查SpatialSorter的参数。我们现在将Grid Size保持在16; 这将给我们一个更粗略的结果，但在我们开发工作空间时会更快地运行。在Group Size参数下，单击下拉箭头并选择Attribute Value> GroupSize：

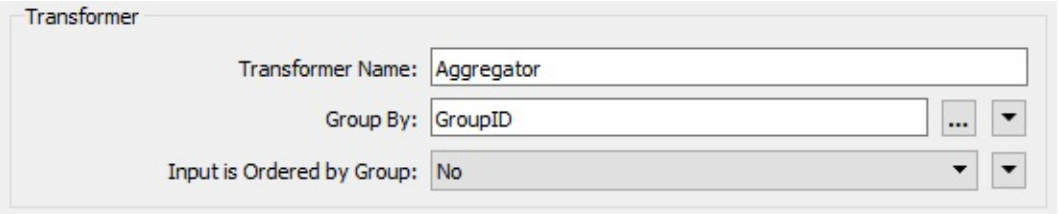


这会将组大小设置为刚刚计算的属性。

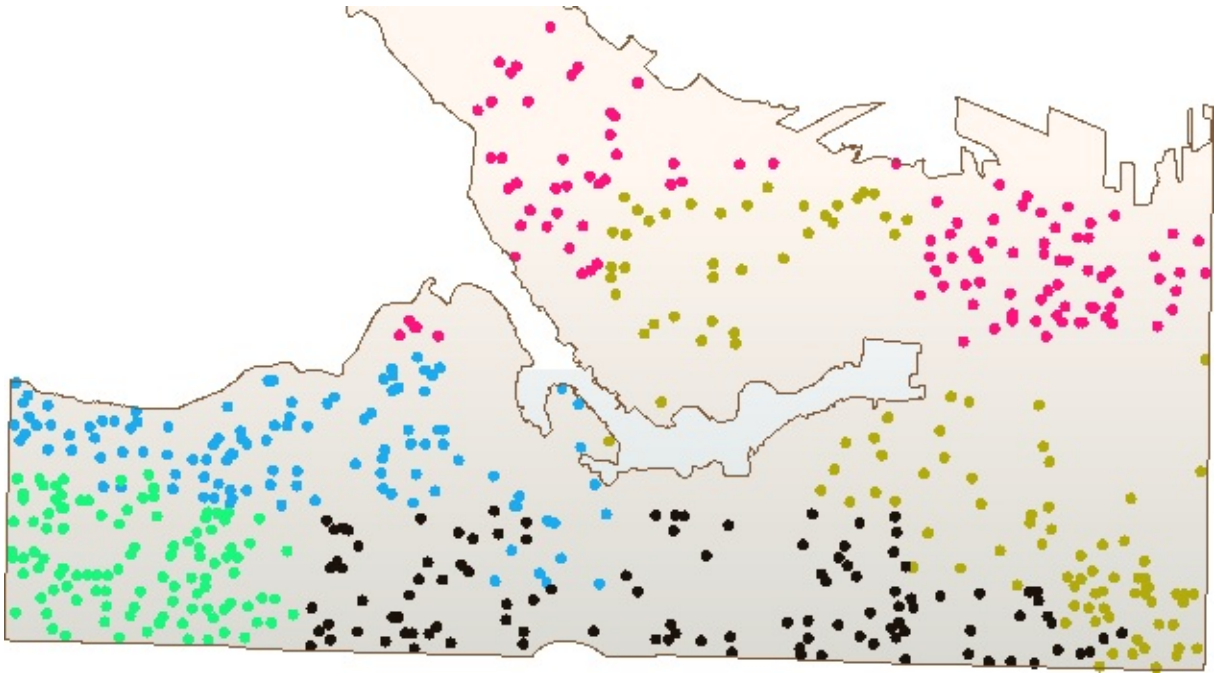
要创建地址组，请将Aggregator转换器移动到“Group Residents”书签，并将其连接到SpatialSorter：Sorted输出端口：



检查 Aggregator的参数。将Group By参数设置为GroupID属性（即，在SpatialSorter创建的组中聚合要素）：



连接Inspector（将Group By参数设置为GroupID也很有用）并运行转换。您应该会发现输出中有五组点聚合，每个聚合具有大致相同的点要素数：



保存工作空间，将其命名为GarbageCollection.fmw或类似名称。理想情况下，名称将包含版本号（如GarbageCollection-v1.fmw），但此练习不需要。

工作空间的下一步是添加区域划分数据，创建空间连接，并根据每个地址的分区类型计算每个属性中居住的居民数量。

恭喜
通过完成本练习，您已学会如何： ● 规划工作空间开发 ● 逐个部分创建工作空间 ● 将源数据限制为小样本 ● 使用FME Hub中的自定义转换器 ● 进行算术运算 ● 将数据聚合到组中 ● 使用版本号保存工作空间

读取和写入工作流

默认情况下，“生成工作空间”对话框使用单个读模块和单个写模块创建工作空间。但是，这并不意味着工作空间永远受限于此。此外，FME可以使用以下方式读/写数据：

- 多个读模块，多个写模块或两者兼而有之
- FeatureReader和/或FeatureWriter转换器
- 集成转换器，如DropboxConnector, Email, HTTPCaller等。

主要组件

工作空间是FME转换中的主要元素，负责存储转换定义。将工作空间视为转换所有要素的容器，存储在以下组件中：

读模块和写模块

一个读模块是转换中读取源数据集的组件的FME术语。同样，写模块是写入目标数据集的组件。

读模块和写模块由“导航”窗口中的条目表示。

要素类

要素类是描述记录子集的FME术语。此术语的常见替代说法是图层，表格，表单，要素类和对象类。例如，DWG文件中的每个层或Oracle数据库中的每个表都由FME中的要素类定义。

要素类由Workbench画布上显示的对象表示。

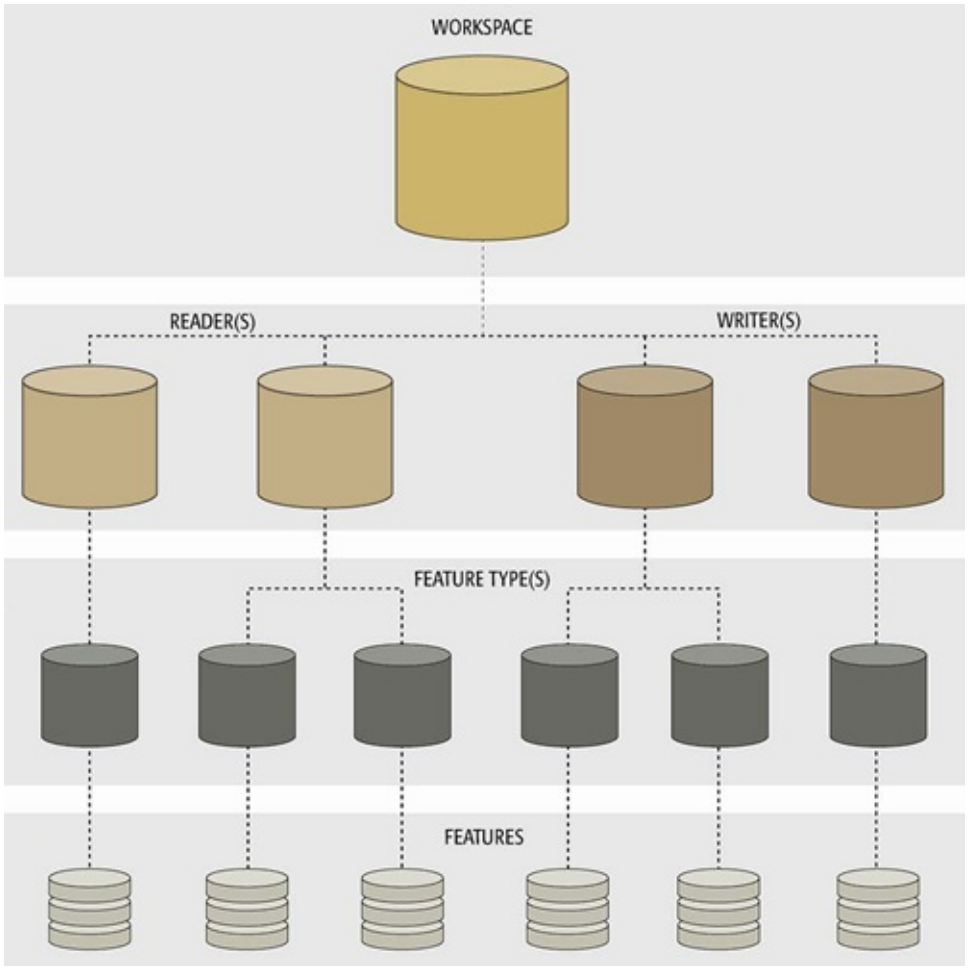
要素

要素是FME转换的最小单个组件。

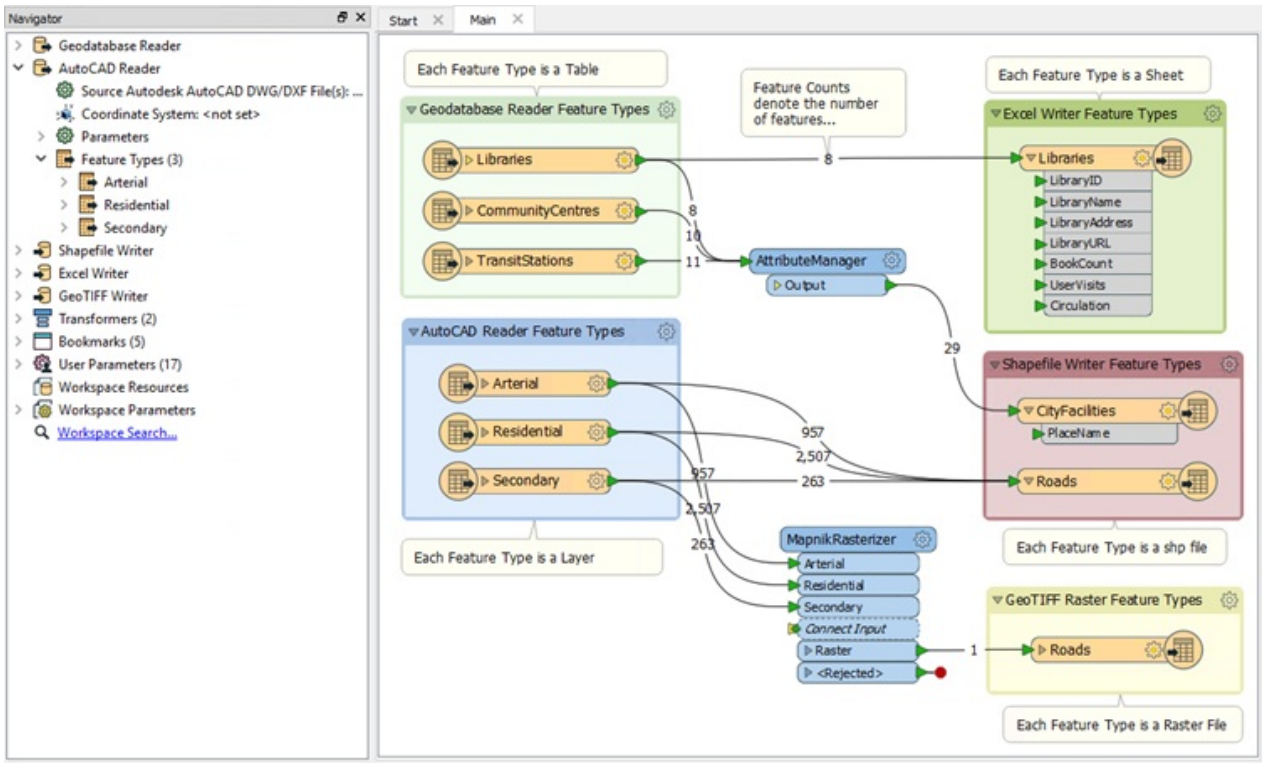
它们不是在工作空间中单独表示，除非已完成转换的要素计数。

关系

每个工作空间可以包含多个读模块和写模块，每个读模块和写模块可以具有多个要素类，具有多个要素。它们存在于如下所示的层次结构中：



具有多个读模块和写模块的工作空间可能如下所示：

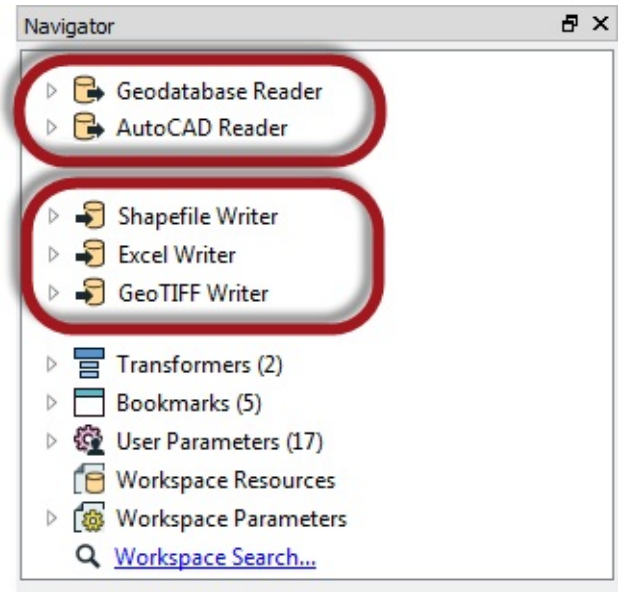


此工作空间有两个读模块（每个具有三种要素类）和三个写模块（具有一个，两个和一个要素类）。每个读模块都是一种不同的格式，每种格式的要素类都有不同的名称。

多个读模块和写模块

FME工作空间不限于任何特定数量的读模块或写模块; 读模块和写模块可以随时添加到工作空间，可以使用任意数量的格式，并且不需要相同数量的读模块和写模块。

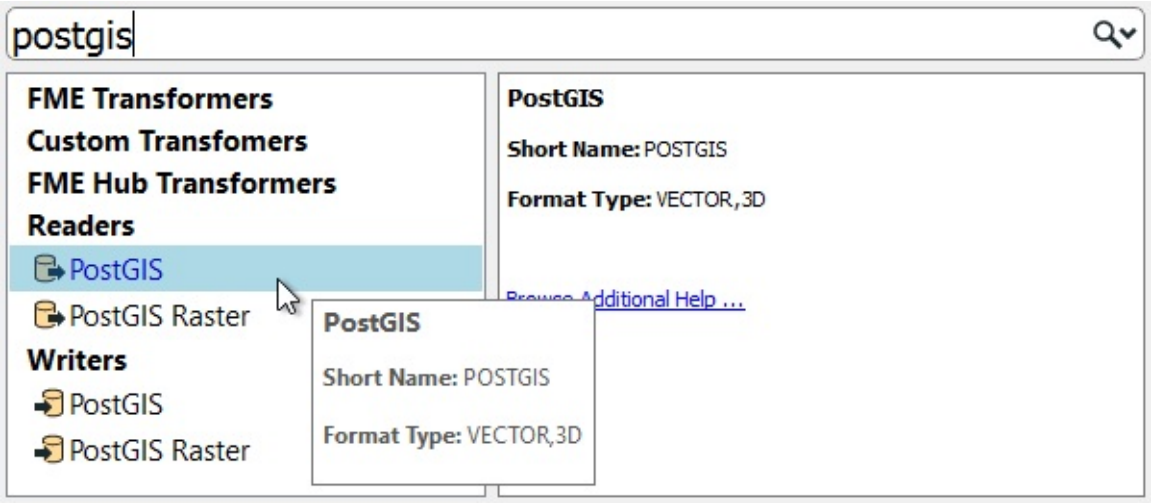
例如，“导航”窗口显示此工作空间包含两个读模块和三个不同数据类型和格式的写模块！



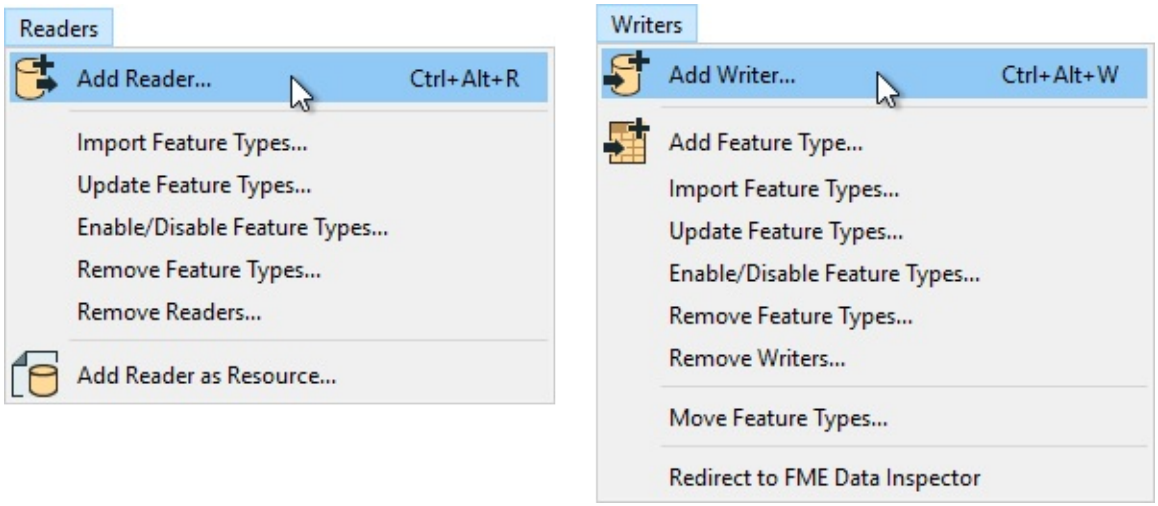
分析师女士说.....
重要的是要注意，读模块和写模块不会在Workbench画布上显示为对象。它们的要素类（层）可以，但读模块和写模块却没有。 相反，它们由“导航”窗口中的条目表示，如上面的屏幕截图所示。

添加读模块和写模块

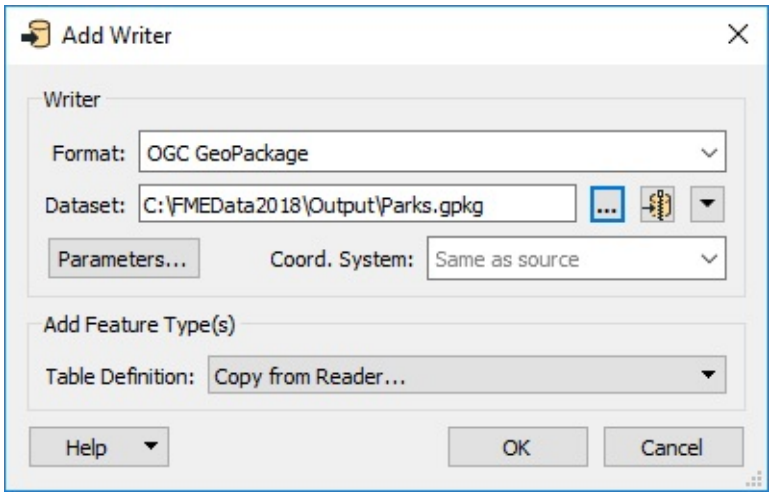
使用“快速添加”菜单将其他读模块或写模块添加到转换中：



...或者从菜单栏中选择“读模块”>“添加读模块”（写模块>添加写模块）：



此操作将打开一个对话框，类似于“生成工作空间”对话框，其中可以定义新读模块或写模块的参数：

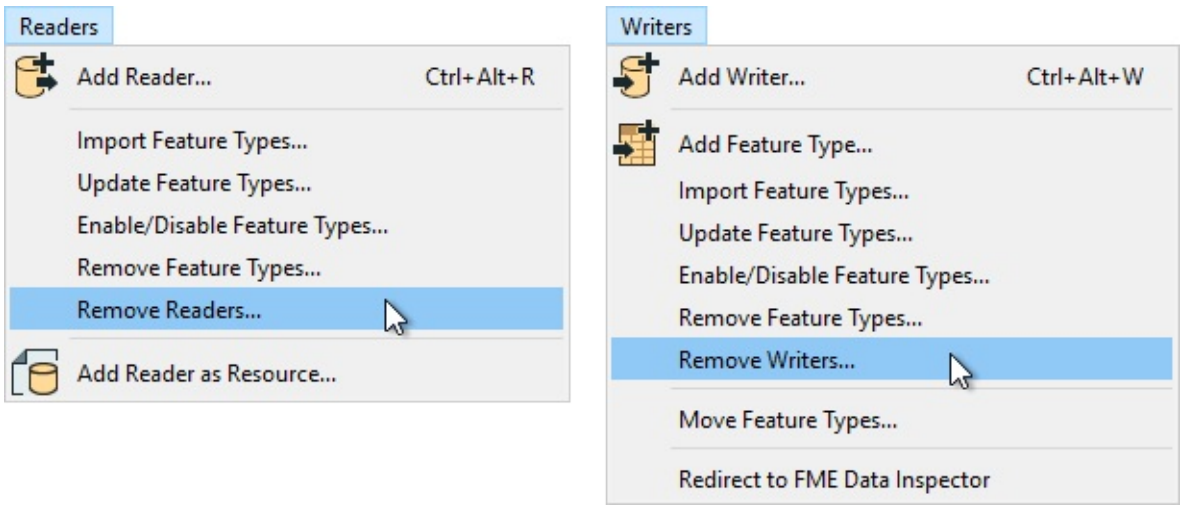


您可以通过这种方式添加任意数量的读模块和写模块。

技巧
还可以通过从文件系统资源管理器拖动数据集并将其放到Workbench画布上来添加读模块。

删除读模块或写模块

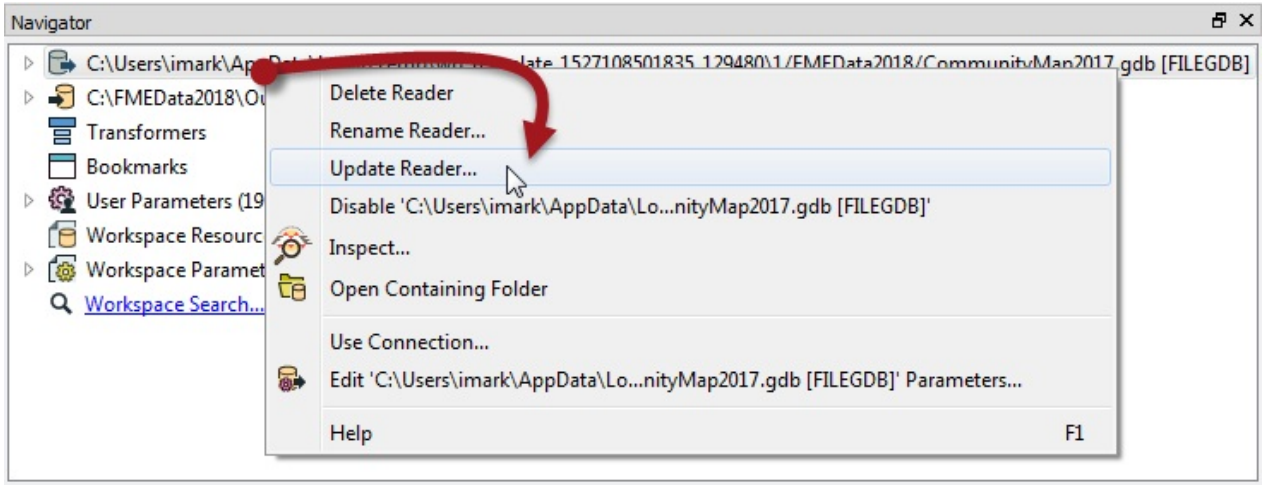
如果不再需要读模块或写模块，则可以使用菜单栏上的选项非常简单地删除它：



或者，可以在“导航”窗口中右键单击读模块/写模块，然后选择“删除”选项。

更新读模块或写模块

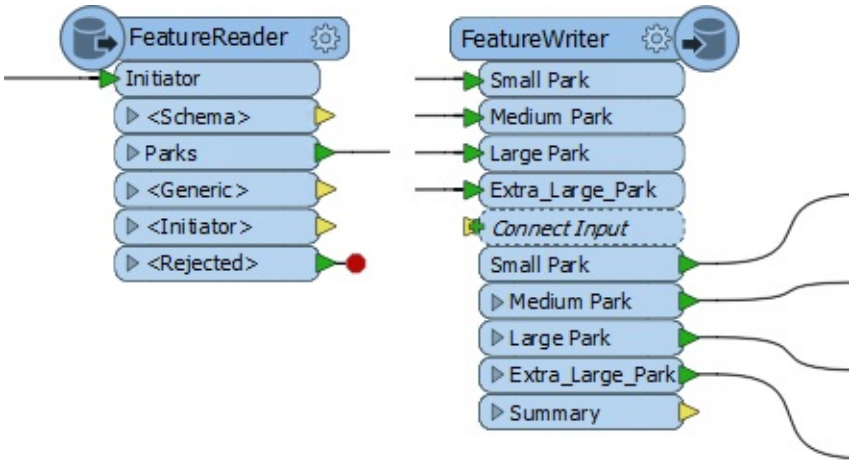
可以更新读模块和写模块，以便旧版工作空间具有较新版本的FME中可用的速度和功能。您可以通过右键单击“导航”窗口中的读模块/写模块并选择“更新”选项来更新转换器：



对于读模块，此工具提供更新读模块或更新正在读取的要素类型列表的选项。这样，如果源数据发生更改，则可以更新工作空间。更新要素类型的另一种方法是菜单栏上的“读模块”>“更新要素类型”。

FeatureReader和FeatureWriter

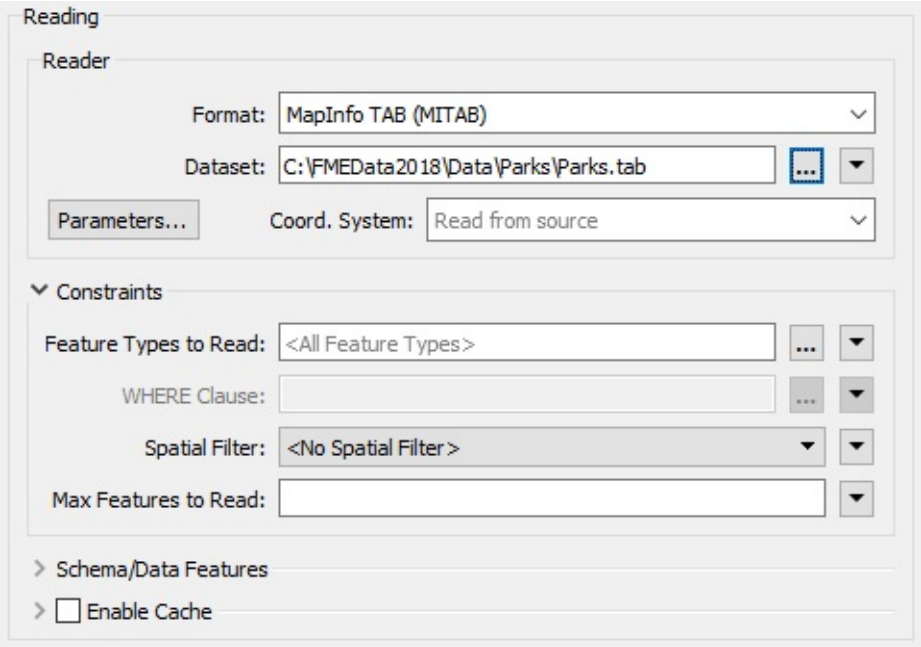
除了能够使用读模块读取数据外，FME还具有专门用于读写数据的转换器。这些是FeatureReader和FeatureWriter转换器。



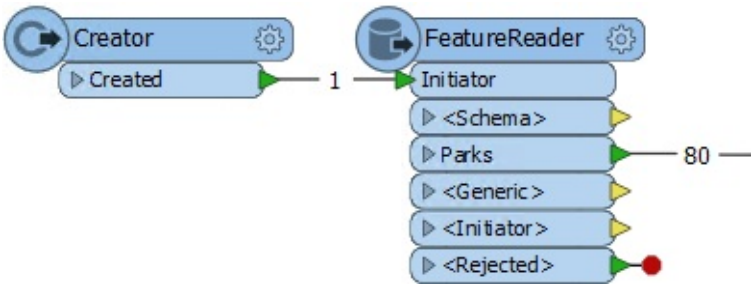
这些转换器的优点是它们具有输入端口（FeatureReader）和输出端口（FeatureWriter）。因此，如果读模块始终是工作空间中的第一个操作，并且写模块始终是最后一个，则FeatureReader和FeatureWriter可以在转换中的任何位置读取和写入数据。

FeatureReader

FeatureReader设置有参数以读取特定数据集：



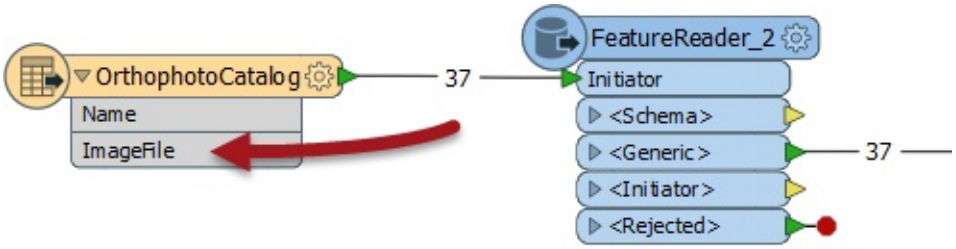
进入Initiator输入端口的任何要素都将导致数据被读取，就像这里Creator提供一个空要素来触发读取：

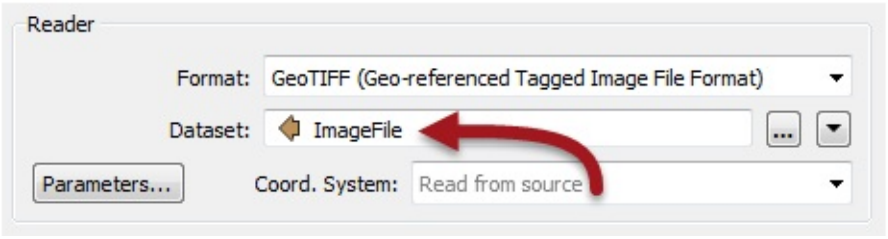


Creator创建单个要素，触发FeatureReader读取公园要素的数据集。如果出于某种原因，Creator创建了十个要素，那么数据将被读取十次，从而产生800个输出要素！

来自属性的数据集

FeatureReader的一个常见情况是提供数据集作为属性读取：





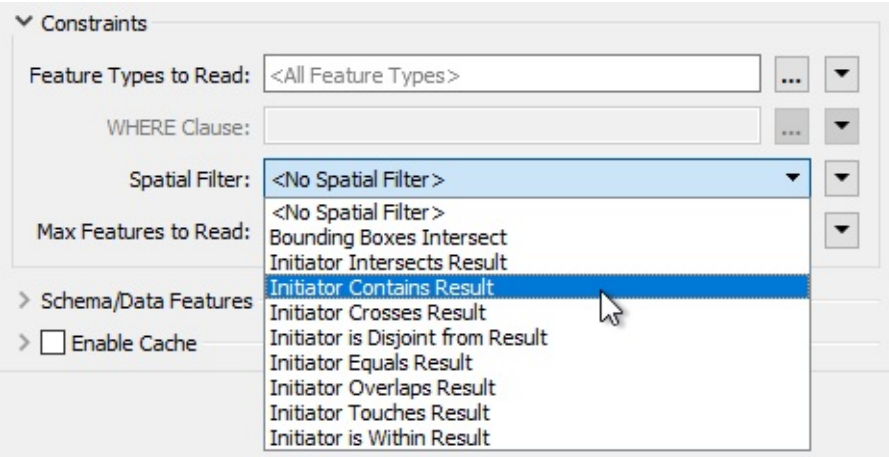
此示例包括reader和FeatureReader。工作空间读取包含正射影像数据集索引的Shapefile数据集。Shapefile中的每个要素都是一个多边形，表示正射影像边界，其属性指向包含该正射影像的GeoTIFF文件。

FeatureReader设置为使用该属性作为要读取的文件名。结果是从Shapefile中读取了37个要素，并且等效的37个GeoTIFF图像退出FeatureReader。

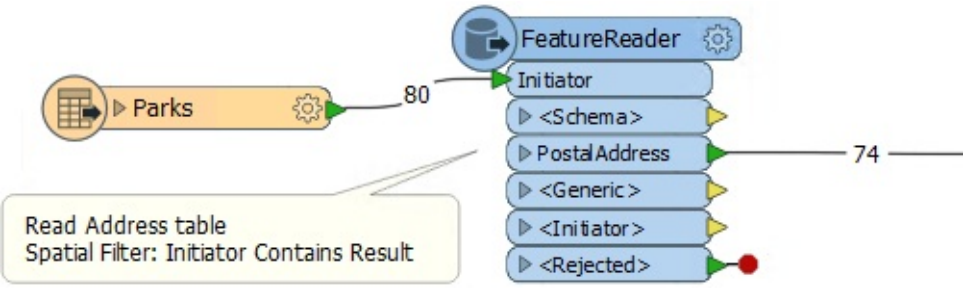
源要素不需要是空间的。例如，带有要读取的文件列表的Excel电子表格同样有效。

空间过滤器

FeatureReader中的关键参数为正在读取的数据设置空间过滤器：



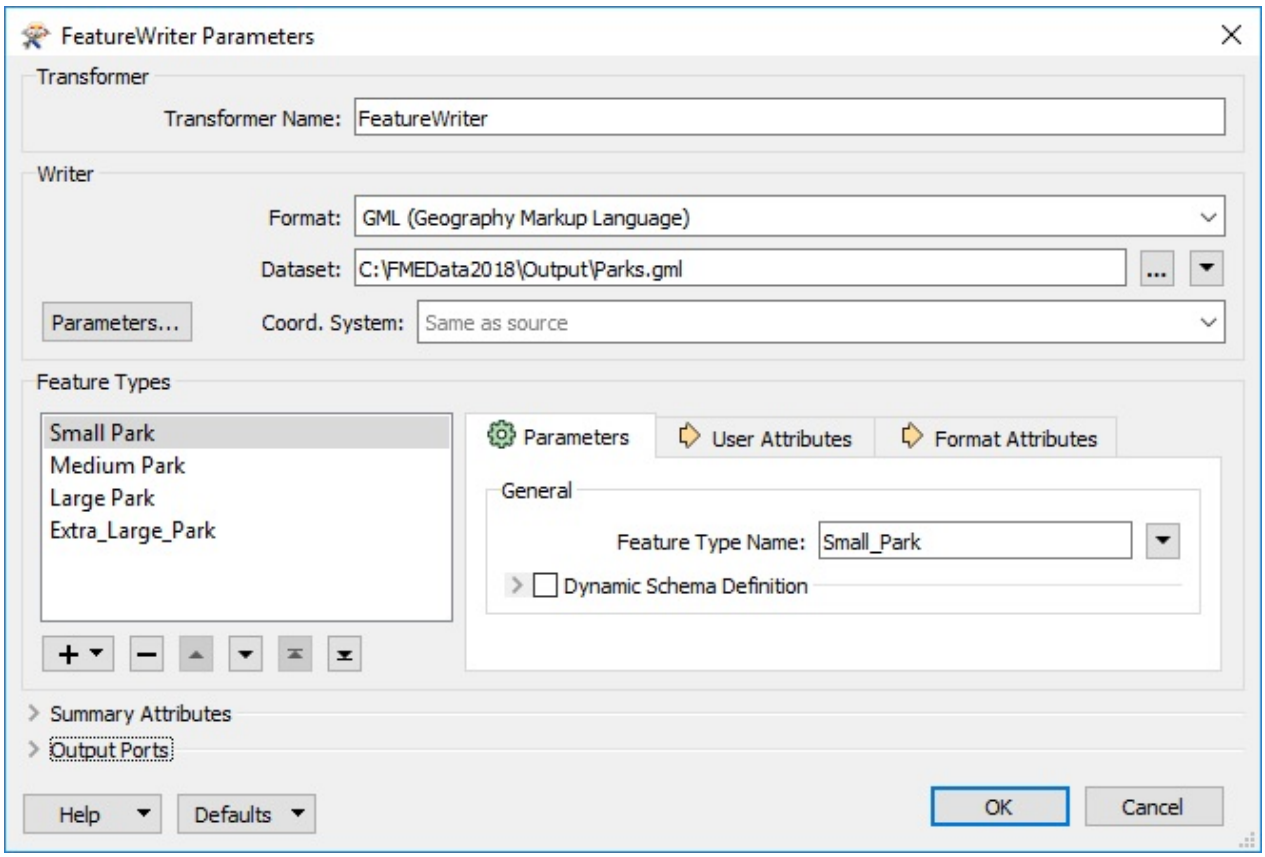
Initiator Contains Result过滤器（例如）表示如果它们的几何对象位于initiator要素的几何对象内，则要素会输出FeatureReader。例子如下：



公园数据集提供触发从数据库地址表读取的输入要素。应用空间过滤器，以便出现的唯一地址是落在公园内的地址。

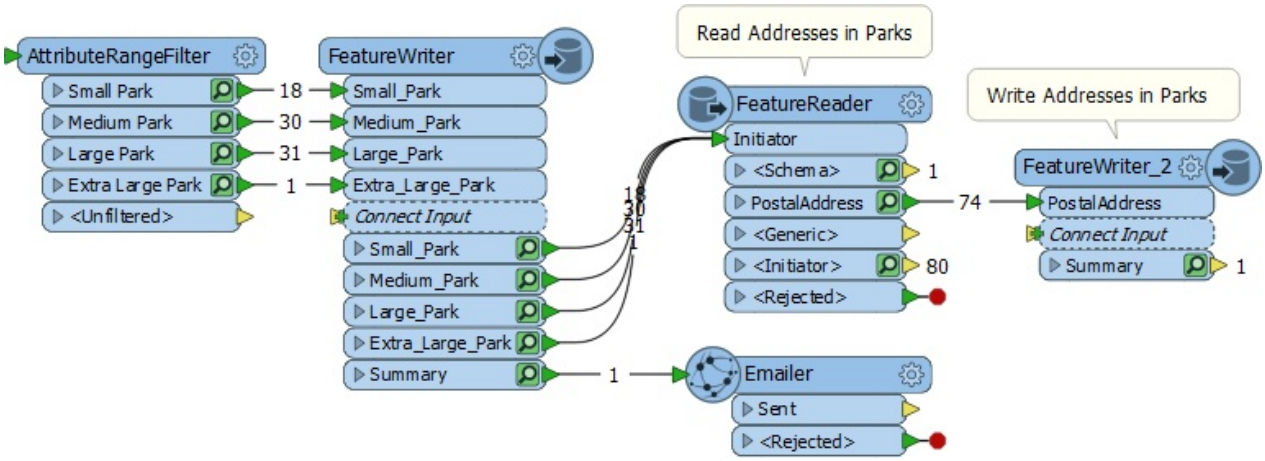
FeatureWriter

FeatureWriter使用参数设置以写入特定数据集：



该对话框允许定义要写入的格式和数据集，以及要写入的要素类型及其属性。简而言之，写模块所需的所有参数，设置和模式定义都显示在此单个对话框中。

可以在对话框本身中手动定义要素类型，也可以通过连接到*Connect Input*输入端口自动添加要素类型：

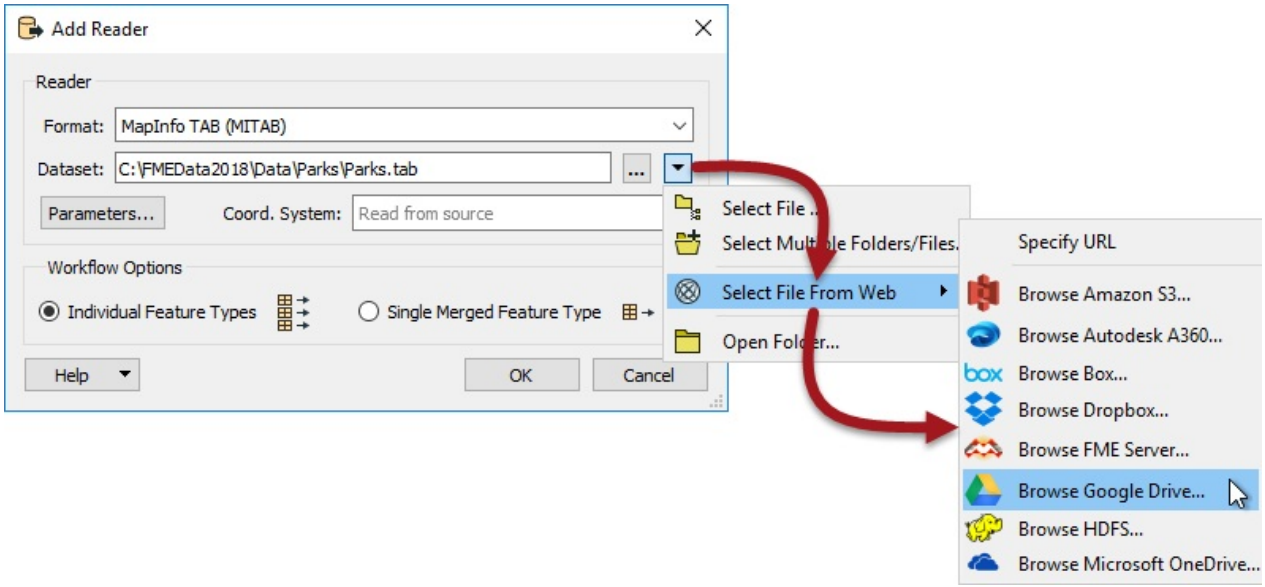


另请注意，FeatureWriter的一个重要部分是其出口可以连接到其他转换器以进行进一步处理。在上面的屏幕截图中，写入了公园数据，以及一个单独摘要要素用于触发Emailer转换器。 Emailer 将数据副本发送给用户。

然后，数据也被用作FeatureReader的输入，以读取落在公园内的所有地址，其结果也随后用FeatureWriter写入。

使用集成转换器进行读取/写入

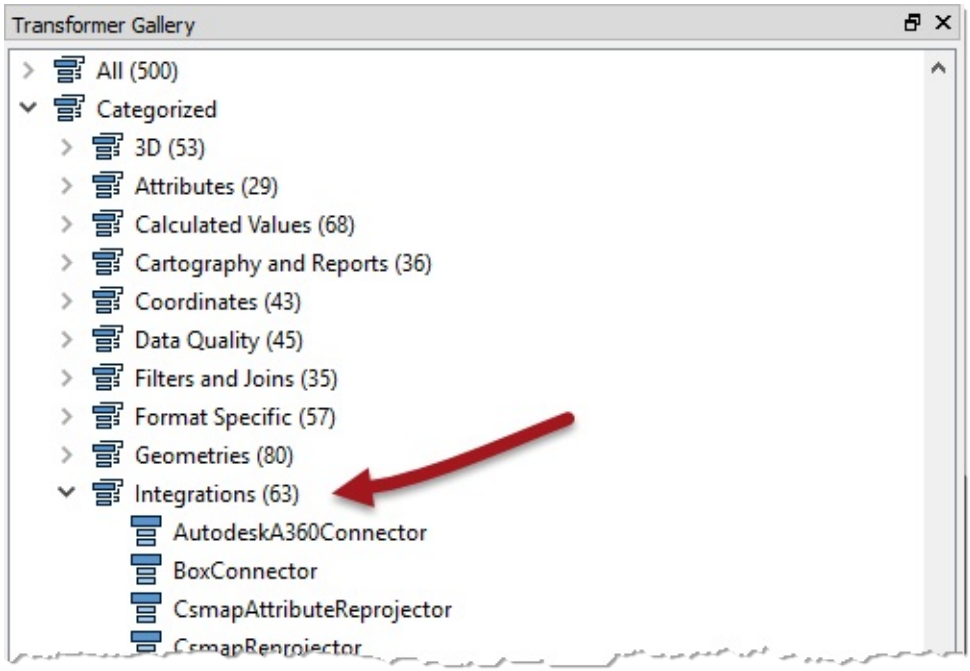
读模块和FeatureReader都可以在本地，数据库或使用Web服务读取数据：



但是，有时能够使用转换器读取（或写入）Web或其他集成服务是有用的。有各种原因，但关键的一个是能够自己读取文件，而不是从中读取要素; 例如，能够从Dropbox中提取JPEG文件并将其存储为属性（而不是将其作为实际的栅格要素读取）。

集成转换器

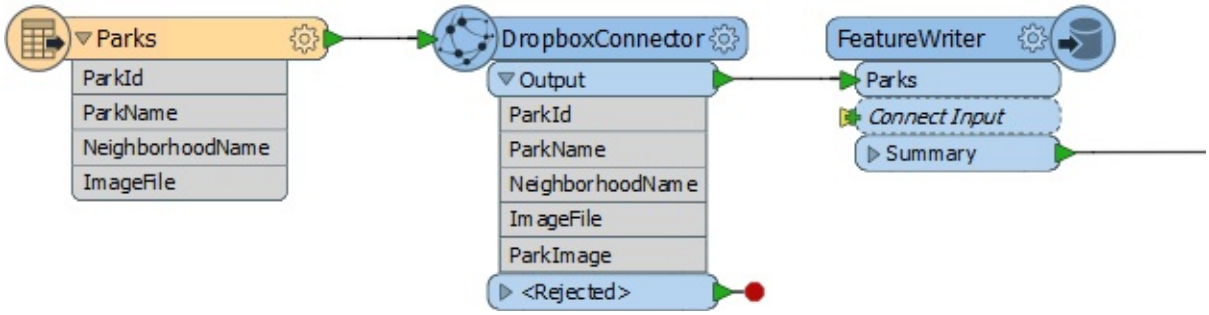
转换器库的集成类别中有多个转换器：



其中一些集成转换器用于读取，写入或复制文件，例如：

• **AutodeskA360Connector**	• **BoxConnector**
• DropboxConnector	• FMEServerResourceConnector
• GoogleDriveConnector	• OneDriveConnector
• S3Connector	

在此示例中，作者正在读取公园的数据集并将要素传递给DropboxConnector：



每个公园要素都具有存储在Dropbox中的JPEG图像的名称。DropboxConnector转换器设置为读取该图像并将其作为属性（ParkImage）存储在该要素上。

然后将这些要素发送到PostGIS数据库 - 使用FeatureWriter转换器 - 其中ParkImage属性被写入二进制（bytea）列。

技巧
与大多数Connector转换器一样，DropboxConnector也可以将内容作为文件上传，而不是从Dropbox中获取文件。

练习：住宅垃圾收集区

练习 **2**	住宅垃圾收集区
数据	地址（Esri地理数据库），区域（MapInfo TAB）
总体目标	为住宅垃圾收集创建边界
演示	读模块和写模块
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Design-Ex2-Begin.fmw
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Design-Ex2-Complete.fmw C:\FMEData2018\Workspaces\DesktopBasic\Design-Ex2-Complete-Advanced.fmw

在这里，我们继续一个重新定义垃圾收集计划的项目。

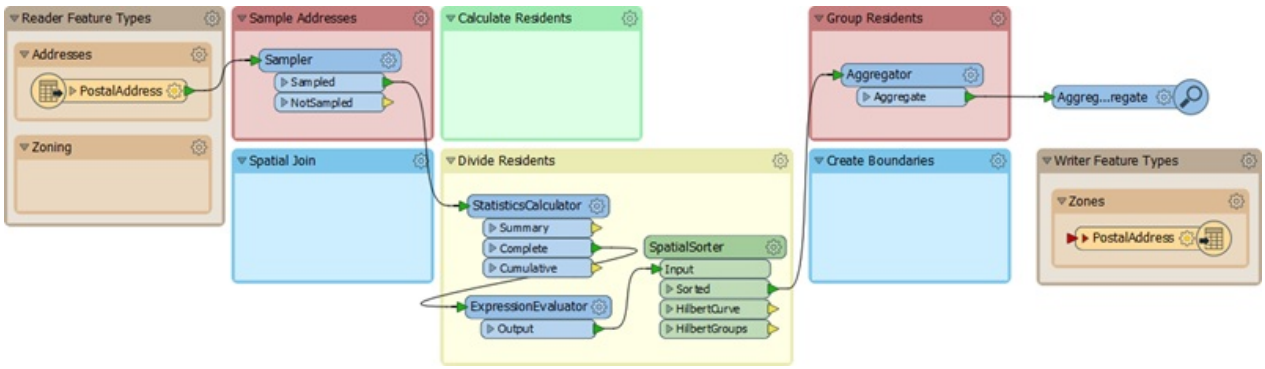
在第一个练习中，我们使用各种变换器将地址划分为五个不同的组。现在的任务是通过估算每个地址的居民数量来根据其所属的区域类型来改进该工作：

- 独户住宅：2名成人
- 两户住宅：4名成人
- 多户住宅：12名成人
- 综合开发区：8名成人
- 商业地产：1名成人

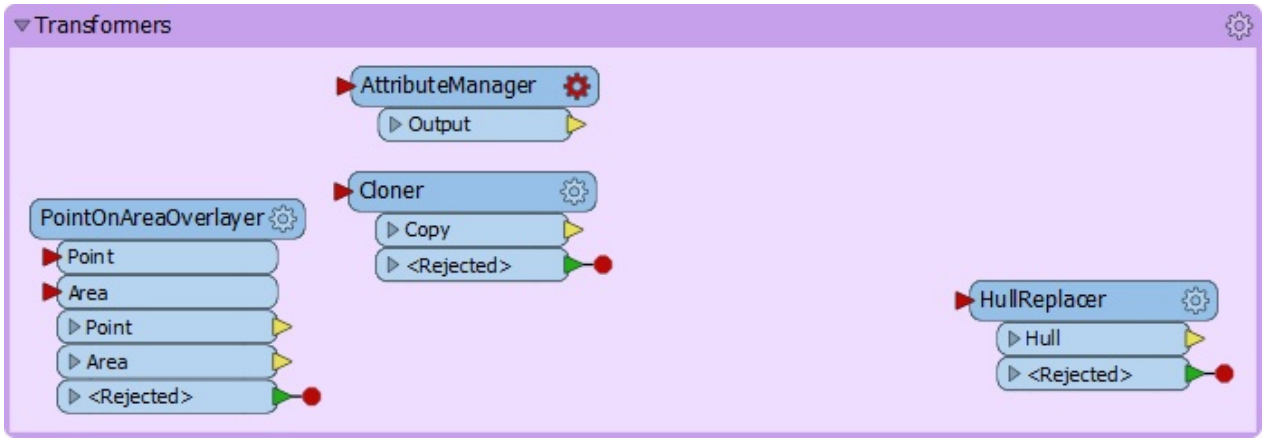
1）打开工作空间
从上一个练习中打开工作空间。

如果您在其名称中为该工作空间指定了版本号，则应使用新版本号创建工作空间的副本。例如，如果您将其保存到GarbageCollection-v1.fmw，则创建一个名为GarbageCollection-v2.fmw的副本并打开它进行编辑。

或者，您可以打开工作空间C：\ FMEData2018 \ Workspaces \ DesktopBasic \ Design-Ex2-Begin.fmw：



“转换器”书签中的其余转换器是：

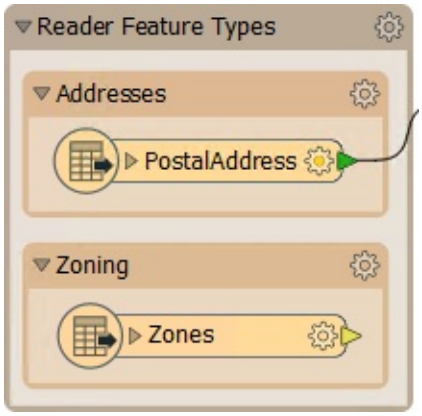


2）添加读模块
这里的第一项任务是确定每个地址属于哪个计划区域。我们需要读取分区数据并执行空间连接。要以不同格式读取新数据数据集，需要新的读模块。

因此，从菜单栏中选择“读模块”>“添加读模块”。出现提示时输入以下参数：

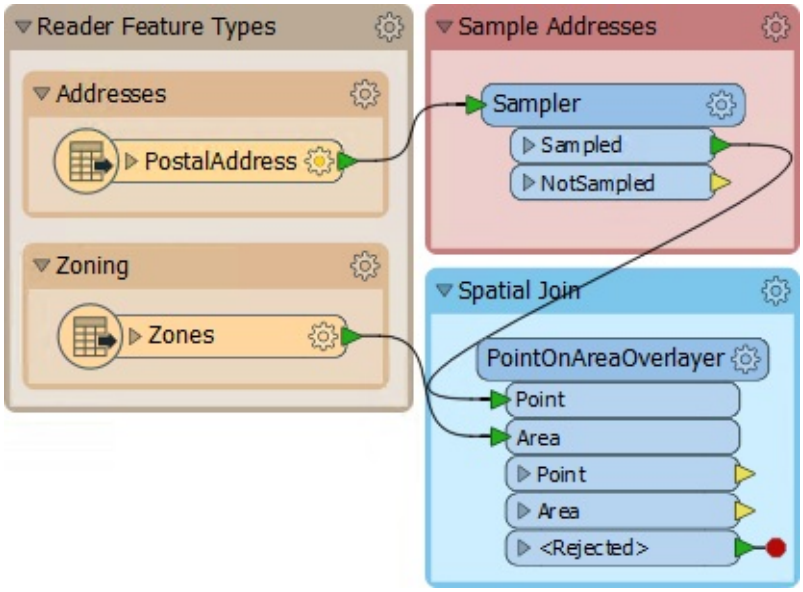
读模块格式	**MapInfo TAB（MITAB）**
读模块数据集	C:\FMEData2018\Data\Zoning\Zones.tab

读模块将添加到“导航”窗口，并将要素类型添加到画布。将要素类型移动到区域划分书签：

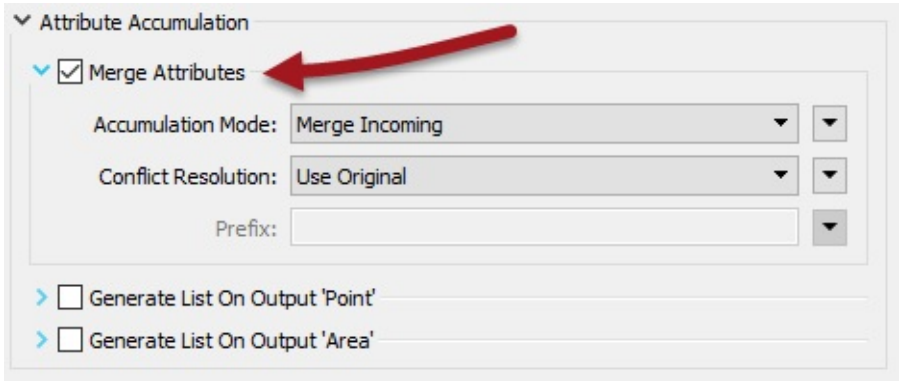


3）创建空间连接
为了执行空间连接，我们将使用PointOnAreaOverlayer转换器; 这是一种称为Point-in-Polygon的连接。

因此，将PointOnAreaOverlayer转换器从“转换器”书签移动到“空间连接”书签。将新添加的分区数据连接到Area端口，将Sampler的输出连接到Point端口：



检查参数并对Merge Attributes进行复选标记：



这个转换器是我们第一次使用具有实时<Rejected>端口的转换器。现在，我们将留下它来终止转换，因为在测试期间我们想要了解导致转换器故障的任何事情。

将Inspector附加到Point输出端口并运行转换。叠加和属性合并应该使每个地址都被赋予区域名称和类别。

4）计算居民数
下一步是根据分区类型设置居住在某个地址的居民数量。

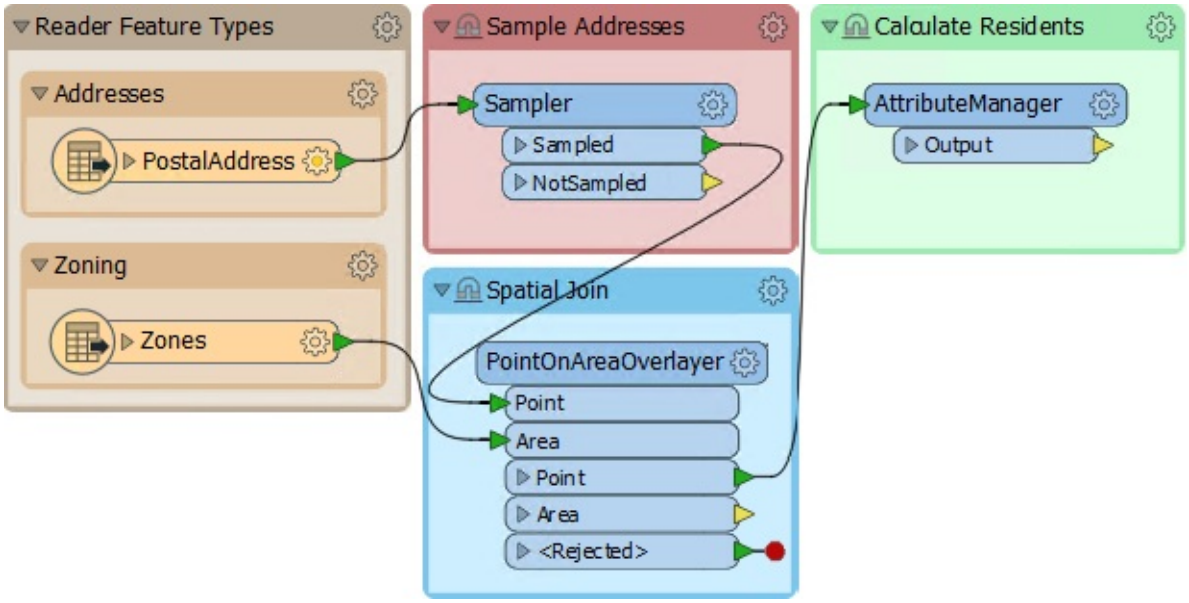
我们知道：

区域开始由	区域类型	居民
RS	独户住宅	2
RT	两户住宅	4
RM	多户住宅	12
CD	综合开发区	8
C	商业地产	1

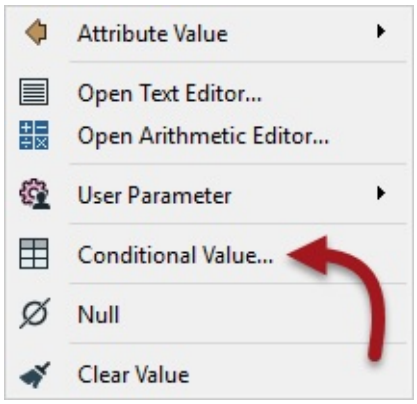
例如，区域RS-1，RS-2，RS-3都是独户住宅区域，并且我们假设每个地址总共有两个成年人。这个假设使它稍微复杂一些，因为我们需要使用“开头”字符串比较来匹配区域类型。

可以使用带有条件值的AttributeManager来完成此匹配。

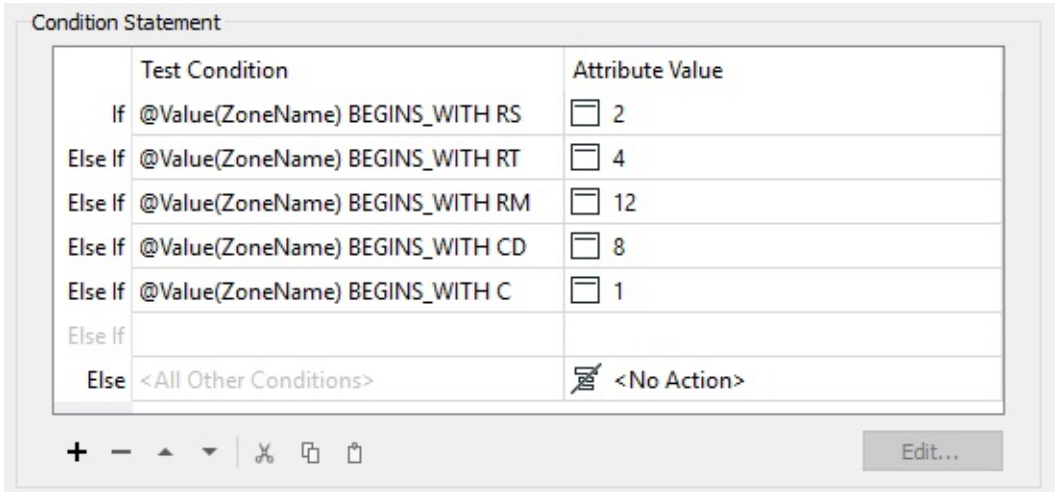
这一步稍微复杂，但幸运的是，“转换器”书签中的AttributeManager已经为此目的而设置。因此，将AttributeManager移动到“Calculate Residents”书签并将其连接到PointOnAreaOverlayer：Point输出端口：



如果您对条件值的外观感兴趣，请打开AttributeManager的参数对话框，然后单击Persons属性的Attribute Value字段中的下拉箭头。选择条件值：



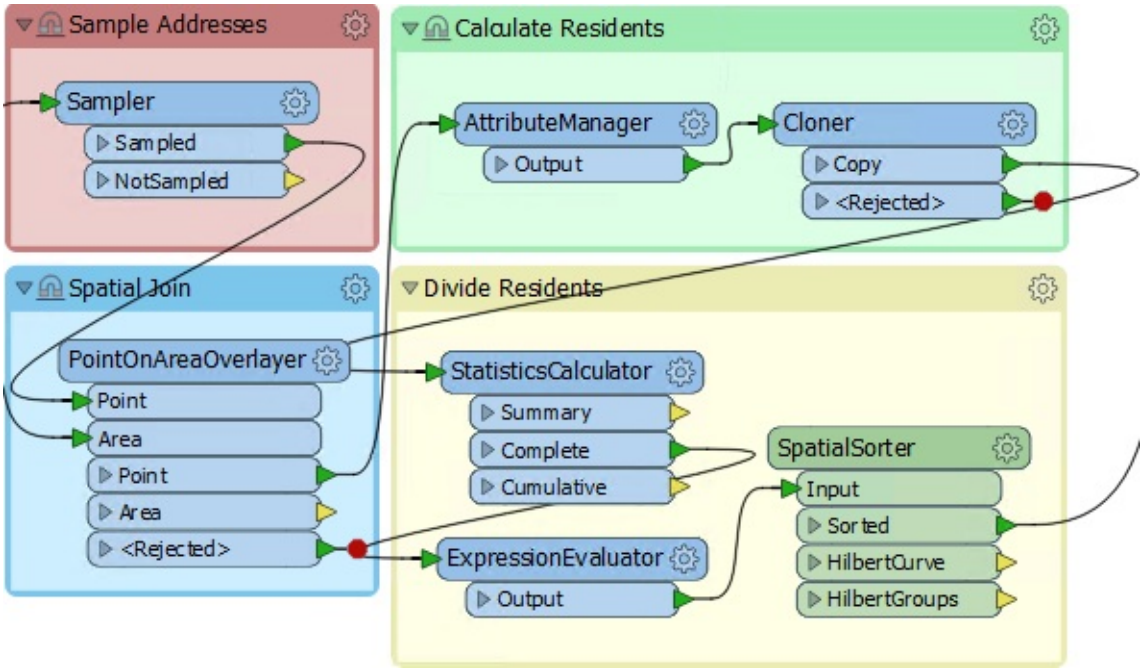
这样做会打开一个类似Tester的对话框，其中包含测试每个区域类型的多个条件，以及一个将它们设置为的属性值：



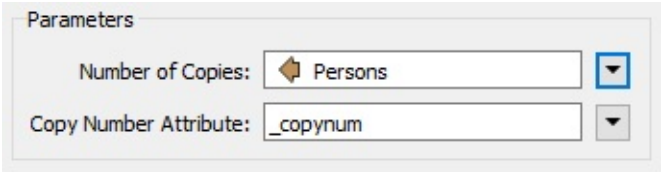
5）创建居民
我们现在知道（或已经近似）每个地址的居民数量。但是，我们必须以影响输出的方式使用该数字。最简单的方法是创建每个地址的多个副本，每个地址一个。

例如，对于有八个居民的地址，我们将创建八个地址点。

我们可以使用Cloner转换器非常简单地完成这项工作。因此，将Cloner转换器添加到Calculate Residents书签中。将AttributeManager连接到其输入，并将其输出连接到StatisticsCalculator：



检查Cloner参数。对于Number of Copies参数，单击下拉箭头并选择Attribute Value> Persons：



这样做会创建原始地址的*<Persons>*副本（请注意，转换器也不会输出原始地址，因此输出为*<Persons>*要素，而不是*<Persons> +1*）。

6）运行转换
确保Inspector仍然连接到 Aggregator转换器并运行转换。转换将失败并显示错误消息：

```
Cloner_&lt;Rejected&gt;: Termination Message: 'Cloner output a &lt;Rejected&gt; featur
e.'
```

转换失败，因为没有居民（例如，工业）的地址没有人员属性，并且被Cloner转换器拒绝。<Rejected>端口仍设置为终止转换，因此我们收到此错误。

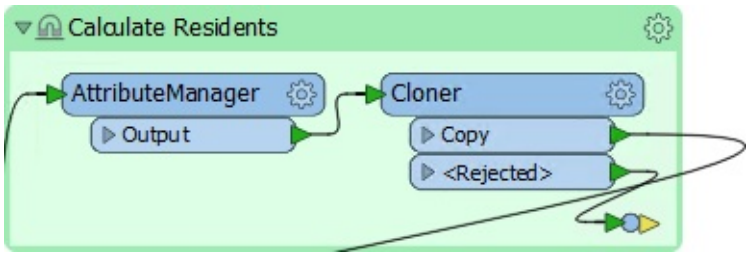
处理这个有多种选择。我们可以：

1. 更改工作空间参数拒绝要素处理以继续转换
2. 添加转换器以处理Cloner的被拒绝要素
3. 设置条件值为给出0的值，而不是完全不包含值

设置条件值将是直接处理问题的最佳解决方案。但是可能存在被拒绝数据的其他原因，我们希望在没有终止转换的情况下处理这些问题。

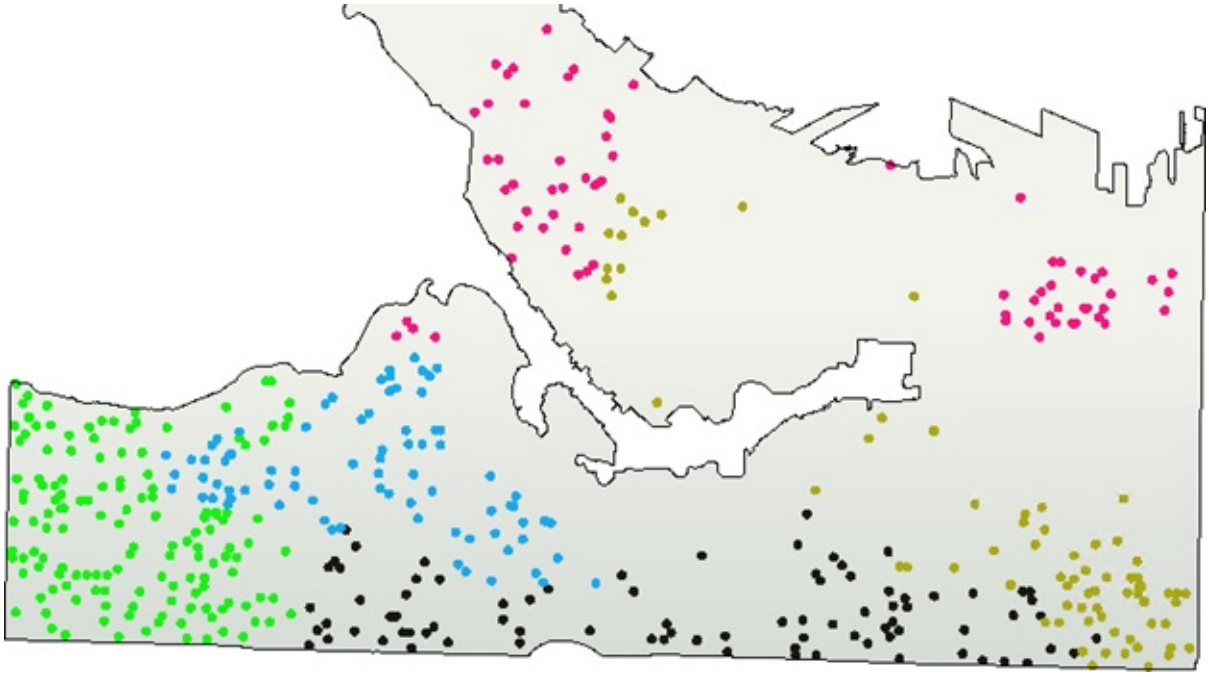
设置Rejected Feature Handling参数意味着所有<Rejected>端口都会忽略被拒绝的输出。此设置在制作工作空间中可能很有用，但在测试中，我们可能希望终止转换，以便我们立即意识到问题。

所以对我们来说，更好的解决方案是在Cloner <Rejected>端口添加一个转换器。我们不需要检查或记录这些要素，因为我们知道它们将存在。因此，将<Rejected>端口连接到一个名为Junction的小型转换器：



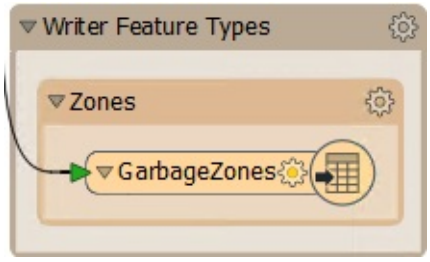
这个Junction将处理被拒绝的输出，但会安静地丢弃它，而不会有更多的麻烦。

重新运行转换。输出应该是五组点要素，但是与上一练习结束时的图案不同：

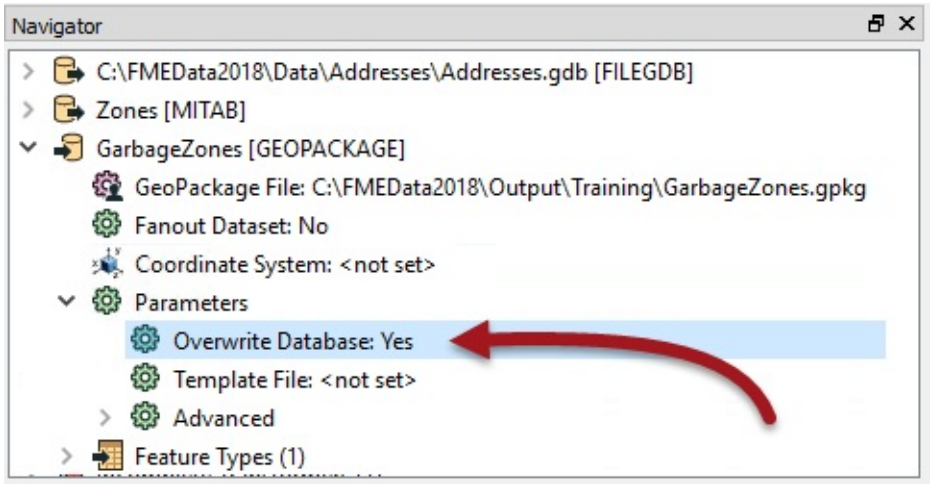


7）写入输出
现在写入一些输出。最简单的方法是将Aggregator输出连接到PostalAddress输出要素类型并重新运行工作空间。

但是，重命名输出要素类型并删除其所有属性也很有用，因为它们来自读模块数据集，因此不适用于此处。因此，打开写模块要素类型参数对话框。在“参数”选项卡中，将要素类型重命名为GarbageZones。在“用户属性”选项卡中，删除正在写入的所有属性：



此外，每次运行工作空间时，更改GeoPackage写模块参数Overwrite Database以覆盖数据库：

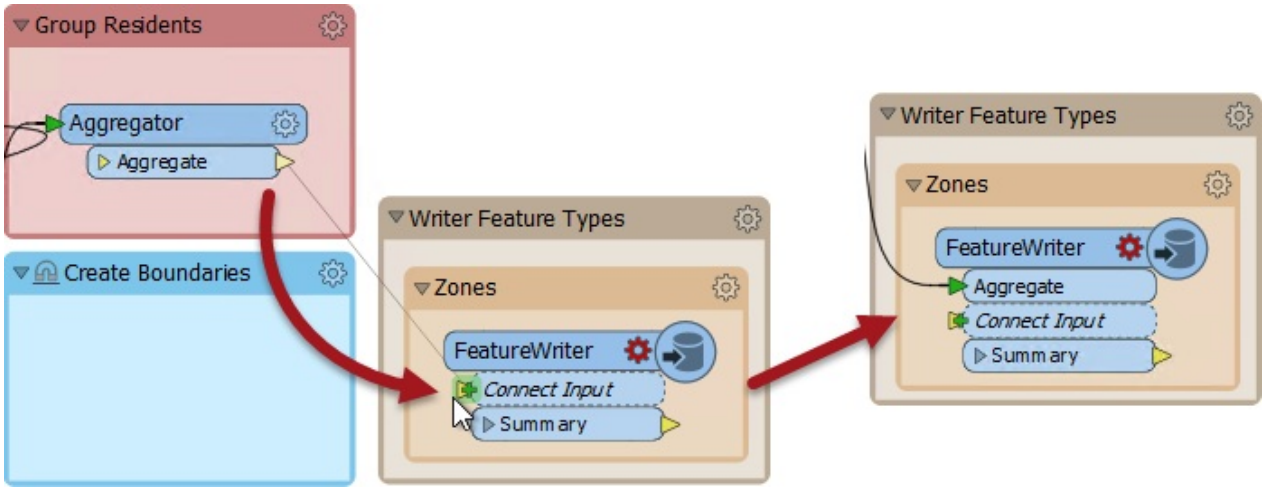


这样我们就不会在同一个数据集中积累越来越多的结果。

高级练习
工作空间工作正常，但假设我们希望自动将输出数据集通过电子邮件发送给某人。要做到这一点，需要Emailer转换器。但是，电子邮件必须在写入数据后出现，因此首先我们必须用FeatureWriter转换器替换写模块。

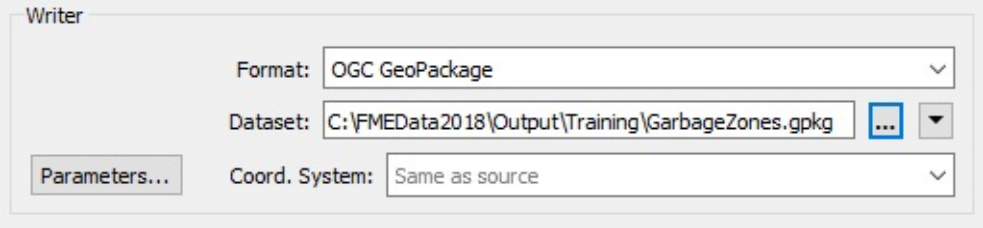
8）版本控制工作空间
如果要对工作空间进行版本控制，这可能是保存新版本的一个好点。然后，如果FeatureWriter不起作用，我们可以轻松地恢复为简单的写模块。

9）添加**FeatureWriter**
添加FeatureWriter转换器代替写模块要素类型，并将Aggregator连接到它：

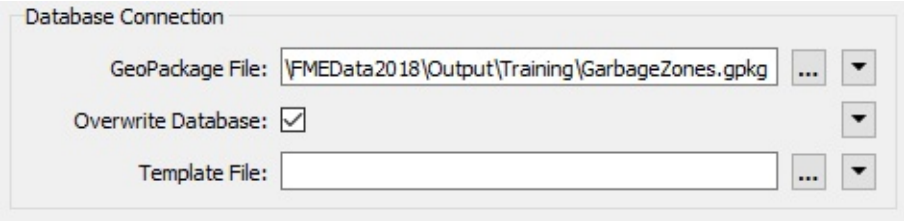


这样做会导致添加Aggregate端口。此端口将成为输出中的默认要素类型名称。

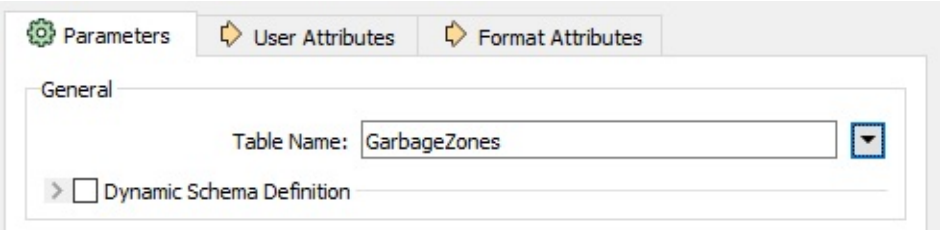
10）设置**FeatureWriter**
检查FeatureWriter参数。将写模块格式和数据集参数设置为已定义的GeoPackage输出数据集：



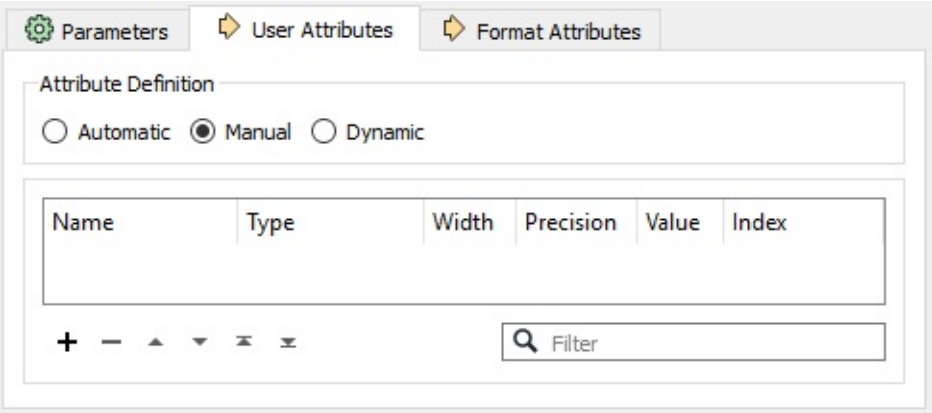
单击“参数”按钮，并确保将参数设置为“覆盖数据库”：



在Feature Type定义下，将输出表重命名为GarbageZones：



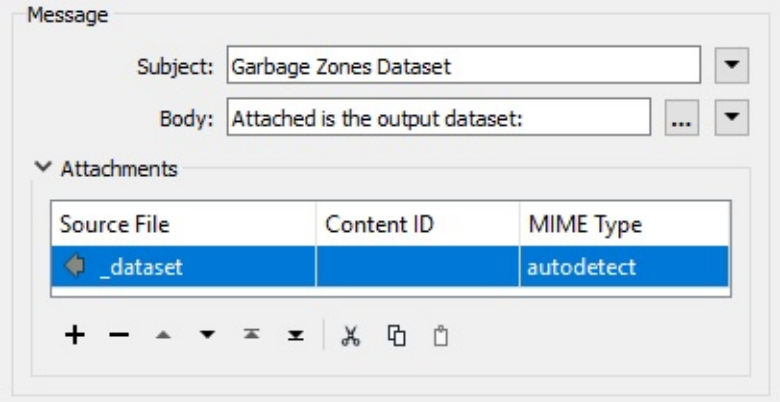
然后单击“用户属性”选项卡，将“定义”更改为“手动”，并删除所有属性定义（不需要它们）：



接受更改。现在，FeatureWriter设置为写入数据，您可以在“导航”窗口中找到GeoPackage写模块并将其删除。

11）添加集成
如果您愿意，并且如果您有可用的帐户，请将集成转换器（如Emailer或DropboxConnector）添加到FeatureWriter的Summary输出端口。

请注意，Summary要素有一个名为_dataset的属性，用于定义写入的数据集。它可以在Emailer中使用（例如）来定义要附加到电子邮件的文件：



现在，当您运行工作空间时，数据将被写入，然后通过电子邮件发送给某人。

警告
与我们使用FeatureWriter写入数据的方式相同，我们可以使用FeatureReader来读取数据。这样做的好处是空间连接内建于流程中。 但是，我们必须考虑这对我们是否有用。当前工作空间在执行空间连接之前对源数据进行采样。FeatureReader会在采样之前进行空间连接，因此可能会更慢。出于这个原因，我们暂时不会这样做。

恭喜
通过完成本练习，您已学会如何： • 将读模块添加到工作空间 • 执行多边形点空间连接 • 在AttributeManager转换器中设置条件值 • 使用Cloner转换器创建多个数据副本 • 管理被拒绝的要素 • 用FeatureWriter替换写模块

工作空间测试技术

在增量开发工作区或对结构化工作区进行编辑时，将进行大量测试。每次进行更改时，作者都希望测试更改，并且此更改/测试过程会一次又一次地重复。

为了有效地执行此过程，FME Workbench中提供了许多工具。

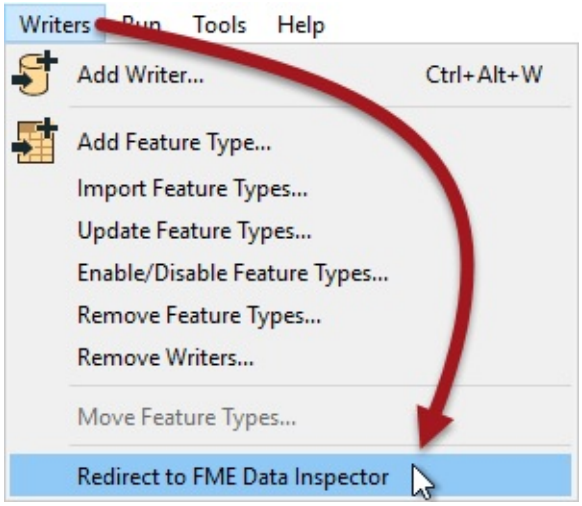
第一组工具用于在数据发生变化时检查数据。第二组工具用于定义需要测试工作空间的哪些部分。

综合检查

在开发和测试工作空间时，有必要定期检查生成的数据。有多种方法可以将数据直接发送到数据检查窗口。

重定向到Inspector

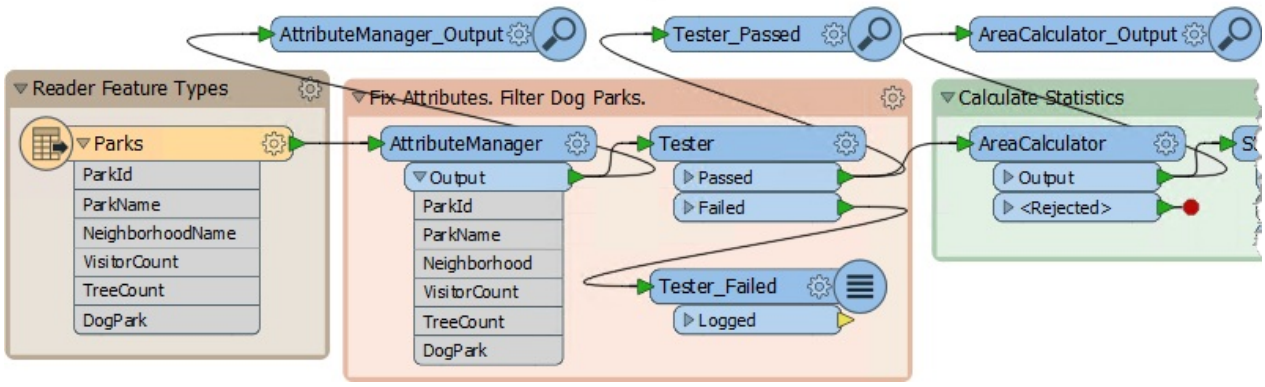
在写入之前查看输出的最简单方法是使用Redirect to FME Data Inspector选项：



在这种情况下，所有数据都将发送到Data Inspector而不是写入; 没有产生输出，并且可以在写入之前检查数据是否存在缺陷。

Inspector转换器

在大多数情况下，有必要在工作空间的某些点检查数据 - 而不是在最后 - 这可以通过Inspecto转换器来完成：

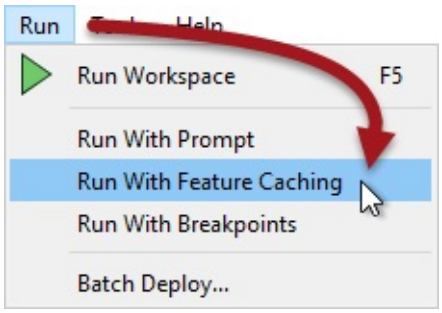


例如，作者将Inspector连接到三个不同的转换器。

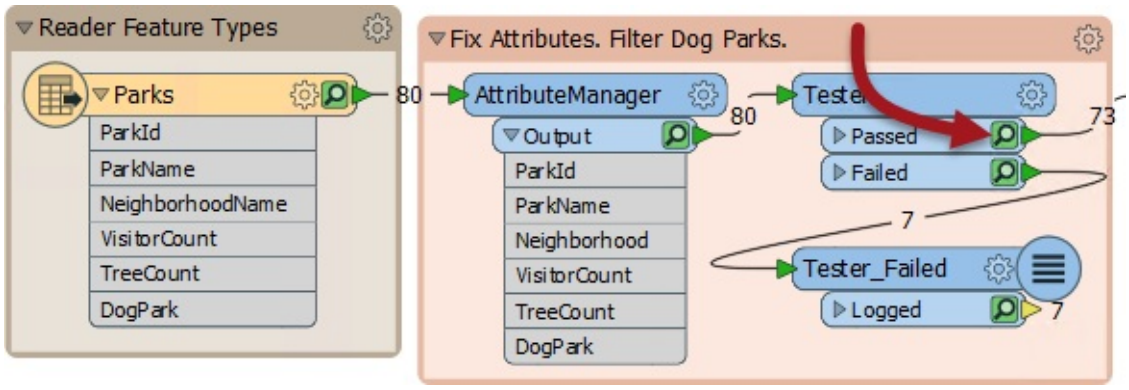
要素缓存

有时，能够在转换的任何步骤检查数据都很重要。在每一步添加Inspector转换器都很烦人，因此FME可以选择自动缓存数据。

使用运行>在菜单栏上运行要素缓存激活此行为：



启用此选项后，FME会在运行工作空间时在转换的每个步骤生成缓存：

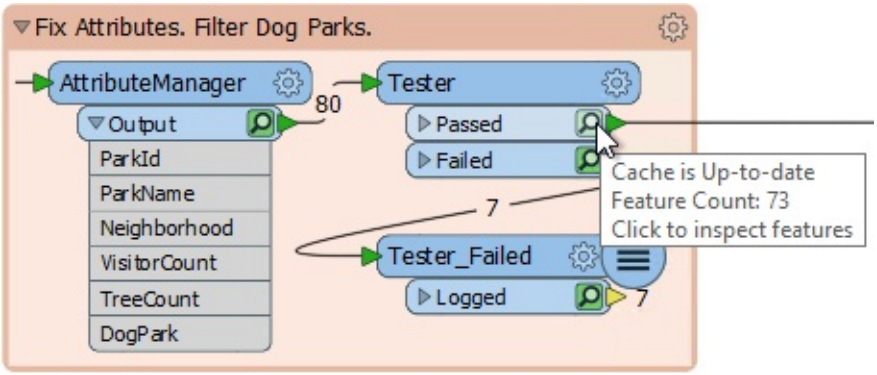


缓存由每个对象上的小图标表示。在上面的屏幕截图中，缓存是绿色的，但它们可以更改为黄色或红色，具体取决于数据的新鲜程度。

新知识
使用数据缓存运行与在先前版本的FME中使用完全检查运行基本相同。它已在FME2018中重命名，以更好地匹配利用这些缓存的新要素。

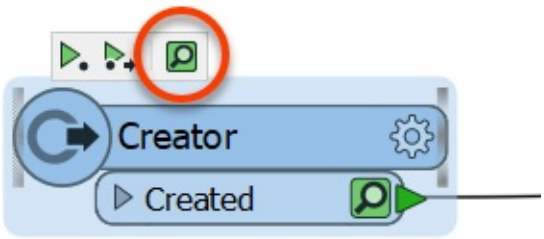
检查缓存数据

可以通过单击特定对象上的图标来检查缓存数据。



2018.1更新

在2018.1中，您可以使用画布上对象上方弹出图标中显示的新“检查缓存要素”按钮立即检



查缓存数据：

技巧

设置“使用数据缓存运行”比手动添加Inspecto转换器更快。但是，请注意缓存数据显然会导致转换速度变慢，并使用系统资源（如磁盘空间）。 数据缓存在开发工作空间时非常有用，但在将工作空间投入生产之前应该关闭。

部分运行

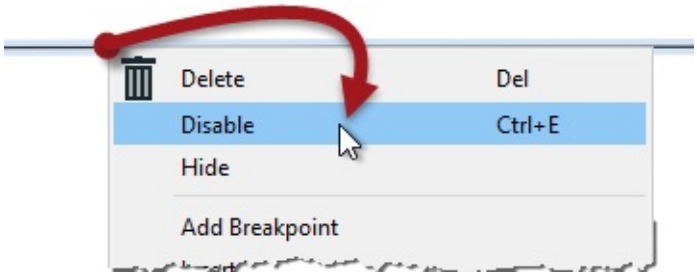
部分运行是指仅执行工作空间的一个部分。一种方法是禁用画布中的对象以仅运行某些已启用的部分。另一种方法是使用名为“部分运行”的工具，当启用缓存运行工作空间时，该工具由弹出选项表示。

您使用的技术取决于工作空间的大小，以及您需要运行多少工作空间。您可以使用一种技术或另一种技术 - 或者您可以同时使用两种技术！

禁用的对象

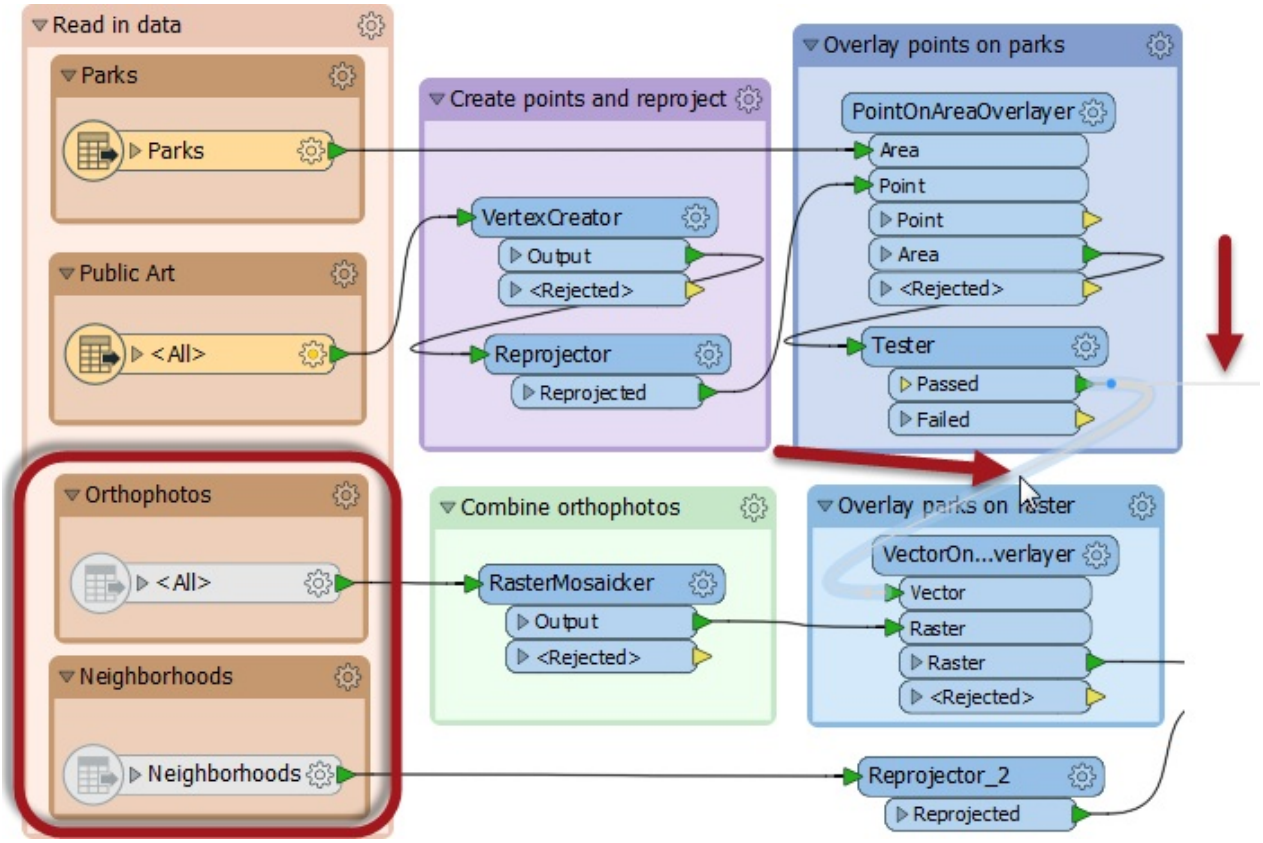
如果设计正确，大型工作空间应由小的部分组成。通过禁用与所有其他组件的连接，可以隔离部分（或部分中的一部分）以进行测试。

通过右键单击对象（连接或要素类型）并选择“禁用”（或选择它并使用快捷键Ctrl + E），可以禁用对象（连接或要素类型）：



禁用的连接不起作用，就像它被删除一样，并且没有任何要素可以通过。可以对其他画布对象（例如转换器和要素类型）执行相同的禁用。甚至可以通过“导航”窗口禁用读模块/写模块。

这里作者禁用了两个连接（都来自Tester：Passed端口）和两个要素类型：



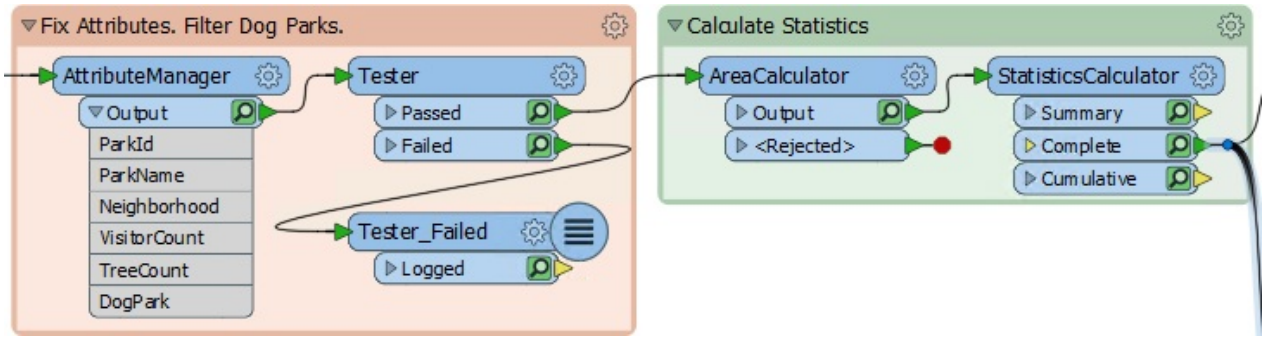
通过该设置，工作空间的顶部将一直运行直到（并包括）Tester。底部根本不会运行。禁用的要素类型不会显示任何数据，也不会从测试程序传递任何数据。

启用缓存后，作者可以检查部分工作空间而无需运行整个转换。（如此例）当禁用的部分占用了整个处理时间的大部分时，此要素是一个显着优势。

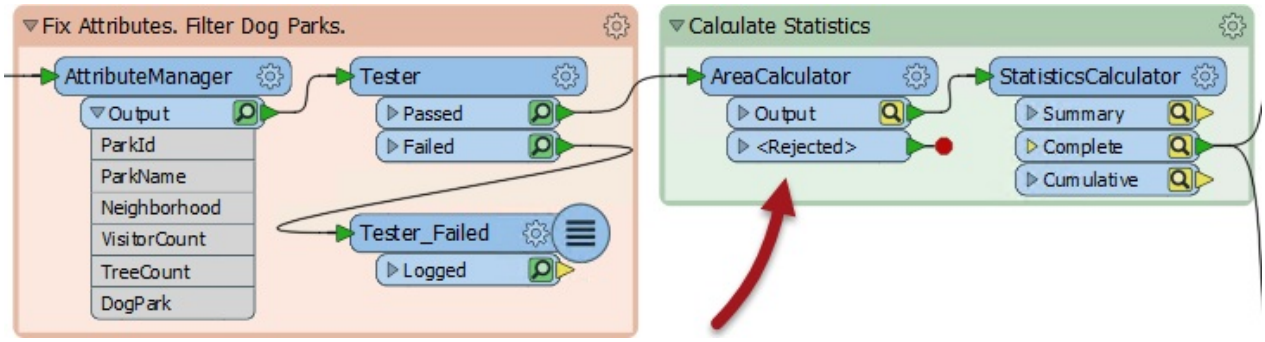
部分运行

启用缓存后，运行转换会导致数据缓存在工作空间的每个部分。在后续运行中，可以使用这些缓存，而不必重新运行工作空间的整个部分。

例如，在此处运行工作空间并启用了缓存：

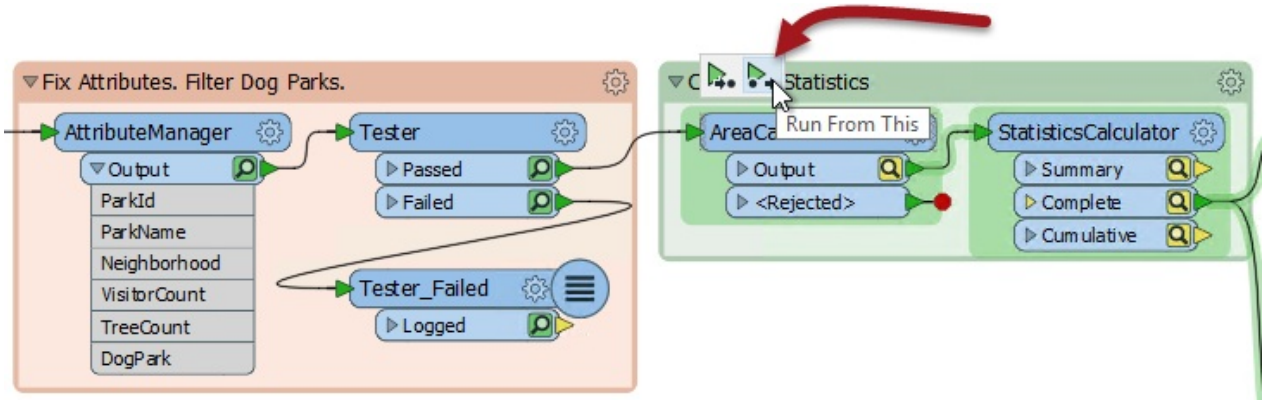


现在作者对AreaCalculator参数进行了更改：



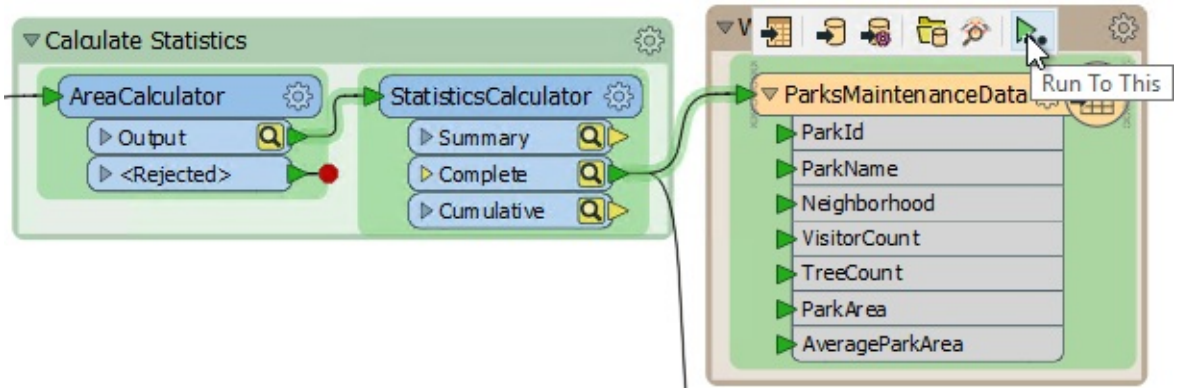
请注意，缓存会在AreaCalculator和后续转换器上更改颜色（变为黄色）。这种颜色表示缓存是旧的; 他们的数据内容不再与工作空间产生的内容相匹配。

要获得新结果，作者必须重新运行工作空间。但是，他们不必重新运行整个工作空间; 他们可以在变化点启动工作空间 - AreaCalculator：



Run From This（从此运行）导致工作空间仅从该点运行，使用缓存到此时为止的数据。注意如何将鼠标悬停在该选项上会导致所有"下游"转换器突出显示。仅有它们是将要运行的。这使转换更快。

另一个选项是*Run To This*（运行到此）。作者可以在写模块要素类型上使用该选项，并获得相同的效果：



...但请注意StatisticsCalculator中的第二个分支如何不突出显示。它不会运行。这表明如何避免运行特定的工作空间部分，就像禁用该连接一样。

技巧
部分运行对于避免从源头重新读取数据特别有用; 特别是当数据来自缓慢的远程位置（如Web服务）时。 此外，当缓存作为模板保存时，可以使用工作空间保存缓存。这意味着可以使用前一个会话中的缓存甚至是由另一个作者重新运行工作空间！

警告
部分运行与变量（VariableSetter / Retriever转换器）不兼容。

练习：住宅垃圾收集区

练习**3**	住宅垃圾收集区
数据	地址（Esri地理数据库），区域（MapInfo TAB）
总体目标	为住宅垃圾收集创建边界
演示	读模块和写模块
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Design-Ex3-Begin.fmw
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Design-Ex3-Complete.fmw

在这里，我们继续一个重新定义垃圾收集计划的项目。

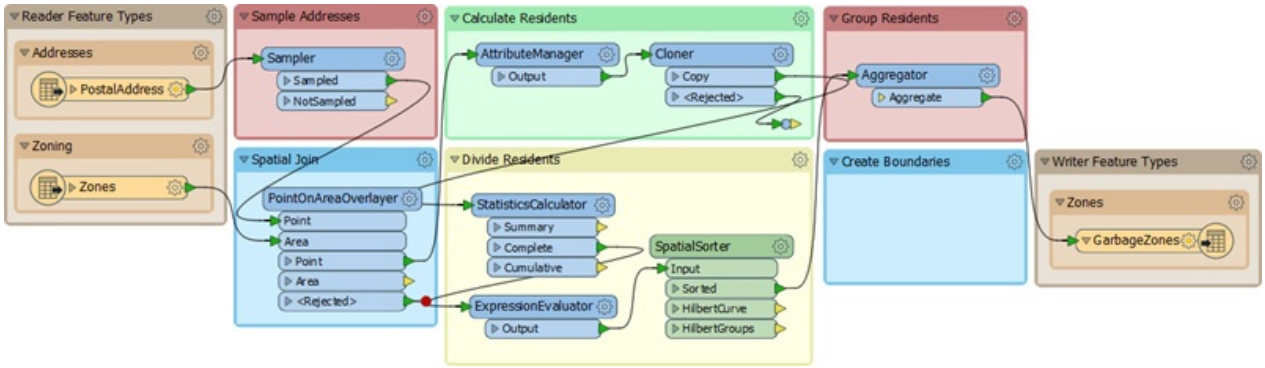
根据分区类型，在前两个练习中，我们使用各种转换器将地址划分为五个不同的组。然后我们将数据写入Geopackage并（可选）用FeatureReader / FeatureWriter转换器替换读写模块。

现在的任务是用多边形边界替换点要素组。

1）打开工作空间副本
从上一个练习中打开工作空间。

如果您在其名称中为该工作空间指定了版本号，则应使用新版本号创建工作空间的副本。例如，如果您将其保存到GarbageCollection-v2.fmw，则创建一个名为GarbageCollection-v3.fmw的副本并打开它进行编辑。

或者，您可以打开工作空间C：\ FMEData2018 \ Workspaces \ DesktopBasic \ Design-Ex3-Begin.fmw：

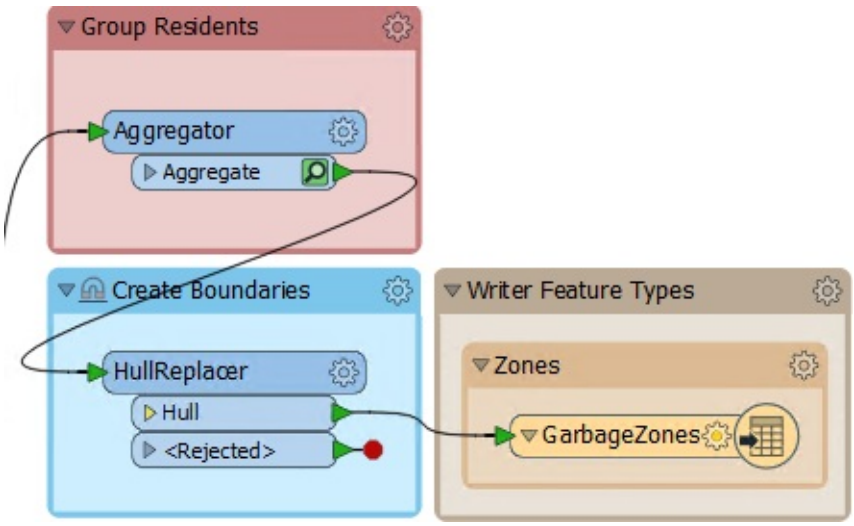


2）运行工作空间
使用菜单栏上的“运行”>“使用要素缓存运行”启用要素缓存。

现在重新运行工作空间。在创建和填充缓存时，该过程将比标准转换时间稍长一些。记下所花的时间（我的机器上为1分7秒）。

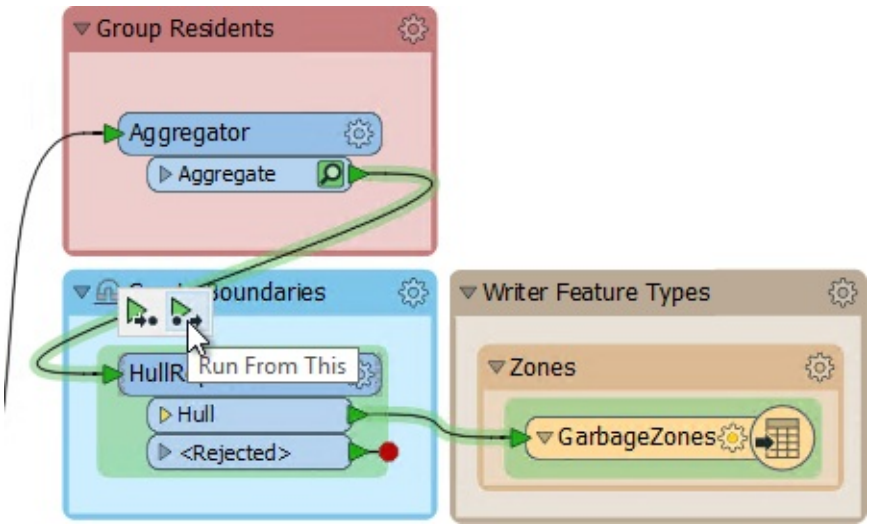
完成后，单击一些缓存以证明可以直接在FME Data Inspector中检查数据。

3）添加**HullReplacer**
将HullReplacer转换器从“转换器”书签移动到“创建边界”书签。在Aggregator和FeatureWriter之间进行连接：



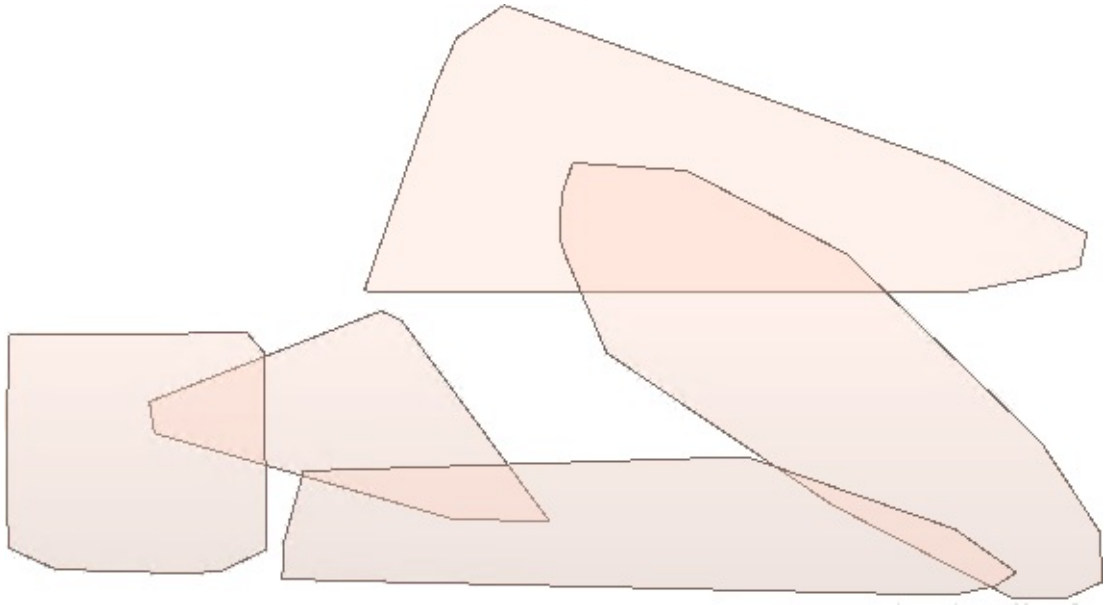
注意HullReplacer如何没有缓存，因为它是新放置的。

4）重新运行工作空间
现在让我们重新运行工作空间。但是，不要运行整个转换，而是单击HullReplacer转换器，然后在弹出的图标上单击“从此运行：



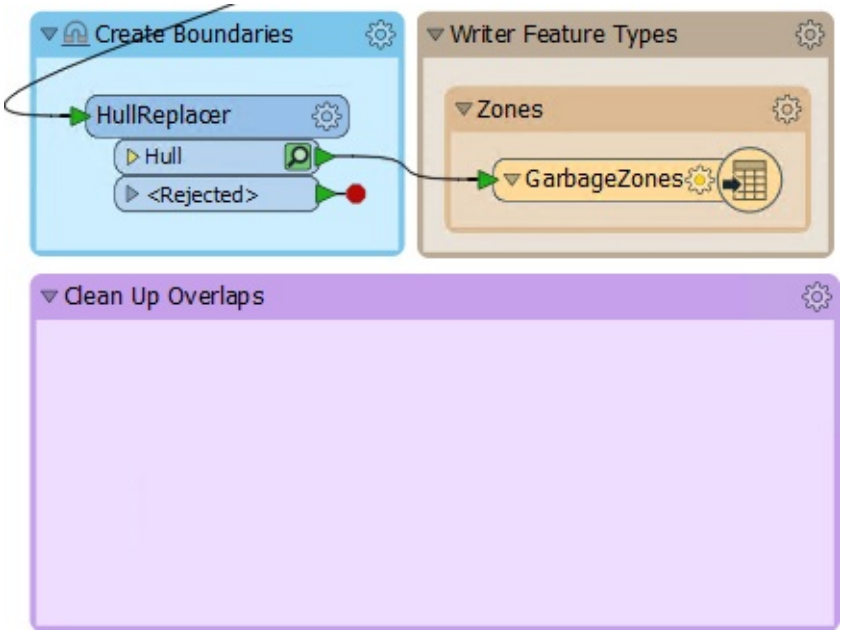
请注意悬停如何显示工作空间的哪些部分将运行。这次的转换应该花费不到一秒，因为工作空间的先前部分的结果已经存储在高速缓存中。

输出现在包含多边形，以证明转换要素正常：

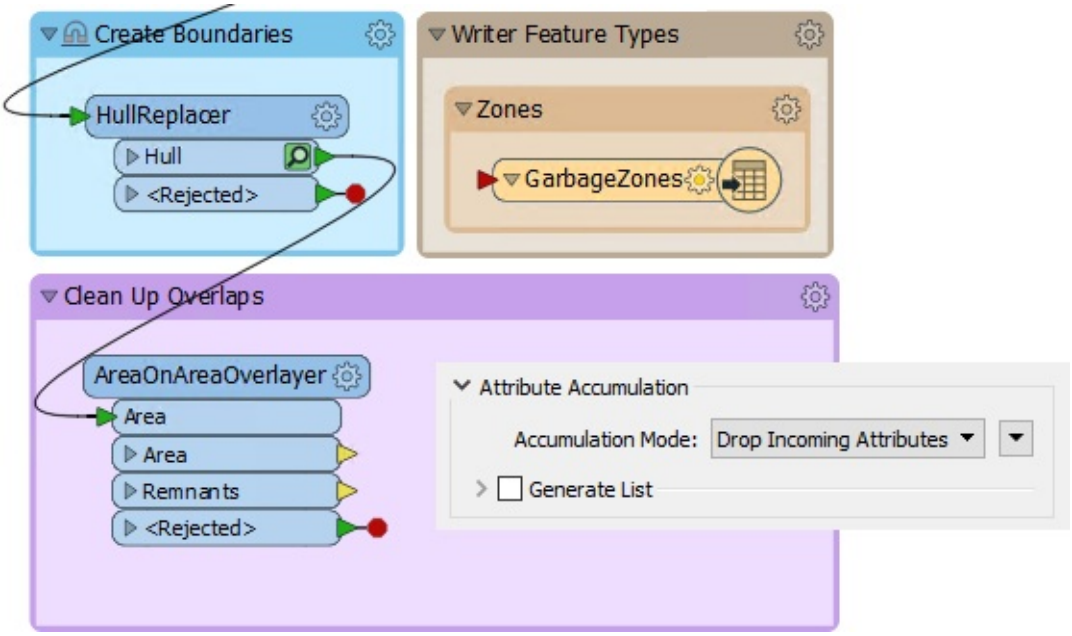


5）清理重叠部分
输出的问题是所有多边形在某种程度上重叠。这需要修复，以便没有重叠。更重要的是，我们应该通过查看哪个组包含其大部分地址来检查重叠属于哪个区域。

因为这是意料之外的，我们还没有预留工作空间的区域。添加新书签（或移动当前为空的转换器书签）并将其命名为Clean Up Overlaps：



6）添加**AreaOnAreaOverlayer**
可以使用AreaOnAreaOverlayer转换器对重叠部分进行切割，因此，将其中一个添加到新书签中，连接到HullReplacer转换器。检查参数并将**Attribute Accumulation Mode**设置为*Drop Incoming Attributes*。



单击AreaOnAreaOverlayer并选择Run From Here重新运行转换。无需添加Inspector转换器，因为只需单击缓存图标即可检查数据。

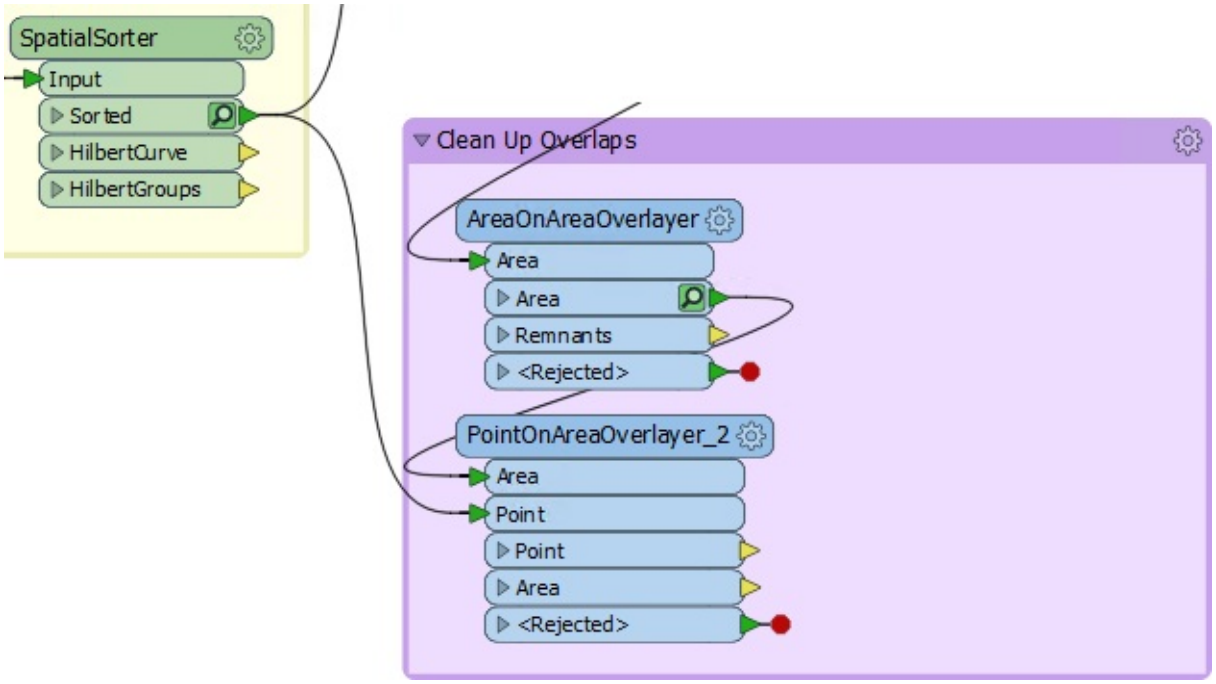
7）添加**PointOnAreaOverlayer**

重叠现在是单独的要素，但我们还不知道它们应分配到哪个区域。应该是地址最多的那个; 例如，如果重叠包含来自组1的31个地址和来自组2的52个地址，则应将其分配给组2多边形。

我们可以通过使用PointOnAreaOverlayer来开始。这个转换器将让我们创建一个重叠包含的地址列表。

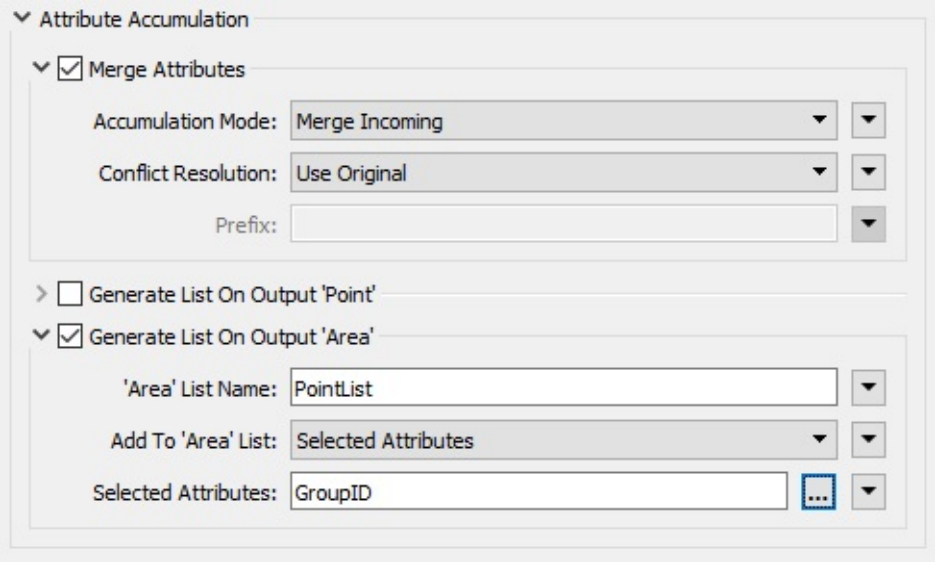
所以添加一个PointOnAreaOverlayer转换器。面要素将是AreaOnAreaOverlayer的输出。

点要素应该是地址的副本。获得这些的最简单方法是从SpatialSorter建立第二个连接：



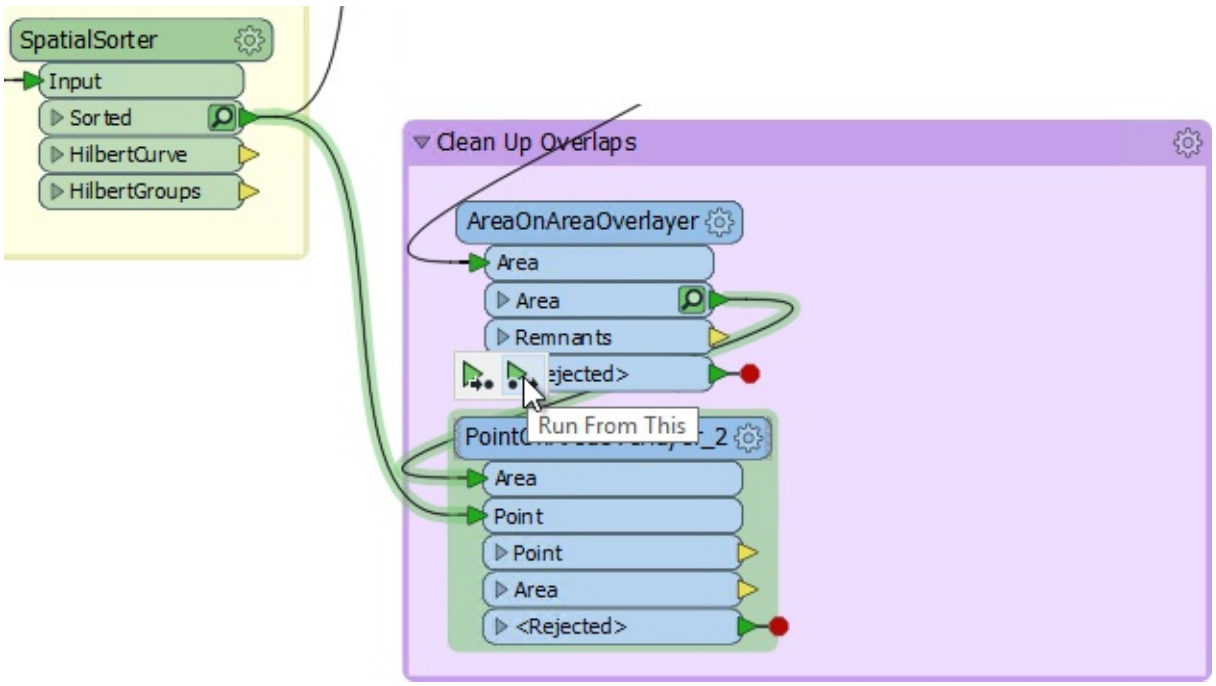
检查参数。在“Attribute Accumulation”下，设置以下参数：

合并属性**(Merge Attributes)**	**Yes**
在输出'面'上生成列表(Generate List on Output 'Area')	Yes
'面'列表名称('Area'List Name)	PointList
选定的属性(Selected Attributes)	GroupID



这样做将创建FME列表属性。列表属性是具有多个值的属性。此列表将包含落在多边形内的点要素（及其GroupID值）的记录。

通过在新的PointOnAreaOverlayer上运行工作空间来确认这是否正常工作。注意转换如何从两个缓存中提取数据; AreaOnAreaOverlayer和SpatialSorter转换器：

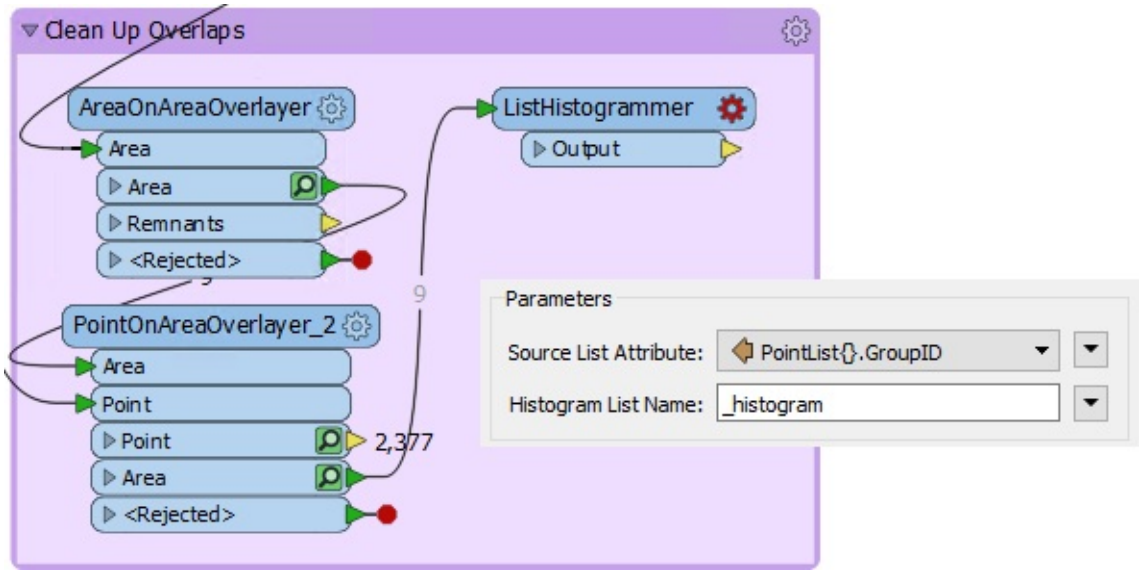


检查面要素。查询要素将显示（在“要素信息”窗口中）列表属性及其所有值。

8）添加**ListHistogrammer**
为了计算重叠上每个列表的最频繁GroupID，我们将使用ListHistogrammer转换器。

技巧
这不是常用的转换器，所以如果您不了解它，或者您担心FME中可用的大量转换器，请不要担心。您将通过练习了解有关这些转换器的更多信息。目前，使用部分运行的能力更为重要。

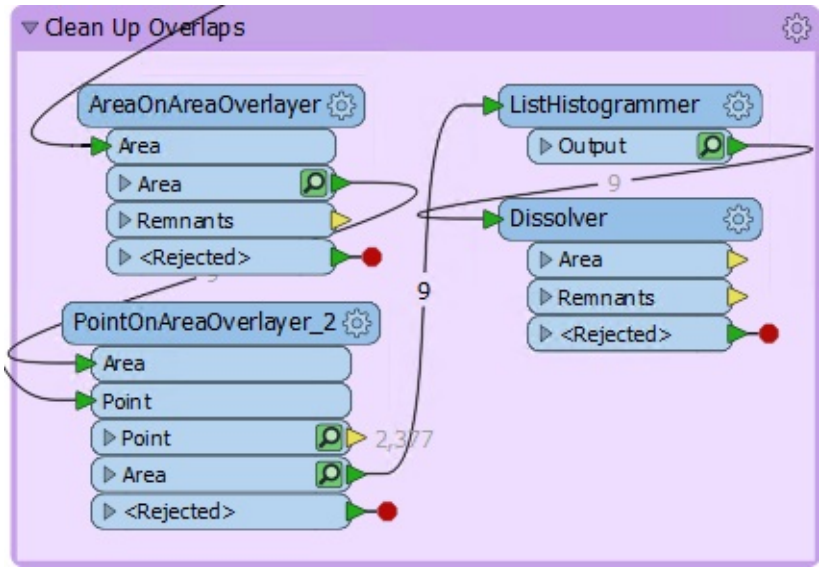
将ListHistogrammer转换器连接到PointOnAreaOverlayer:Area输出端口。检查参数并选择PointList{}.GroupID作为源属性：



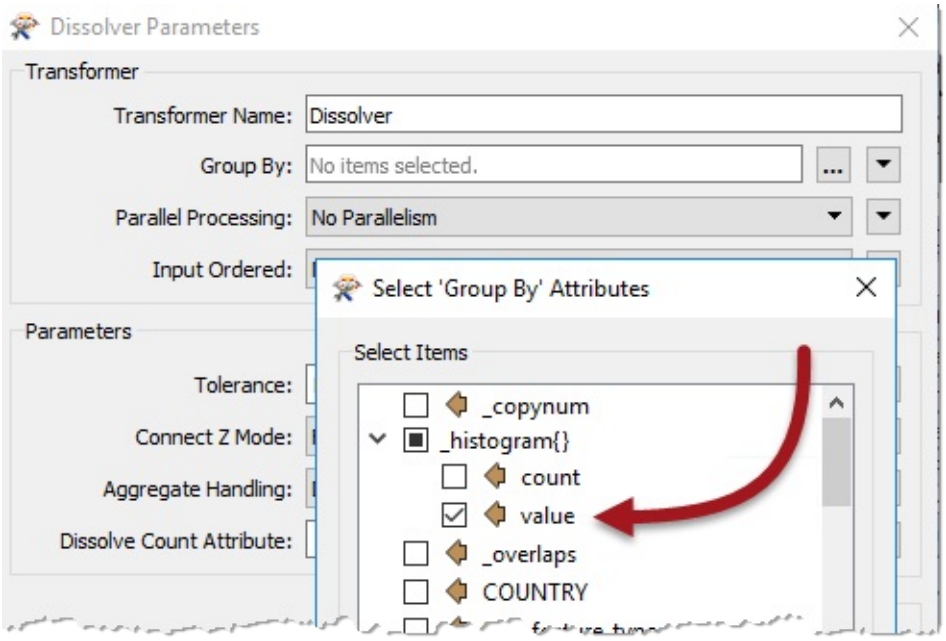
重新运行工作空间（从ListHistogrammer）并检查结果。请注意，创建了一个新的列表属性;在列表顶部有最频繁GroupID的不同GroupID值的数量列表。所以我们只需要使用该GroupID来合并区域。

8）添加**Dissolver**

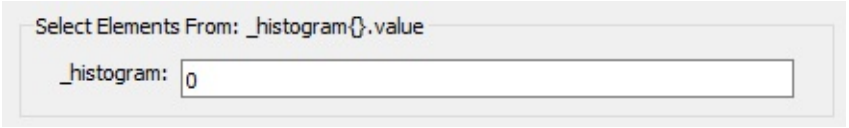
最后添加Dissolver转换器以将这些要素合并在一起。将Dissolver连接到ListHistogrammer输出端口：



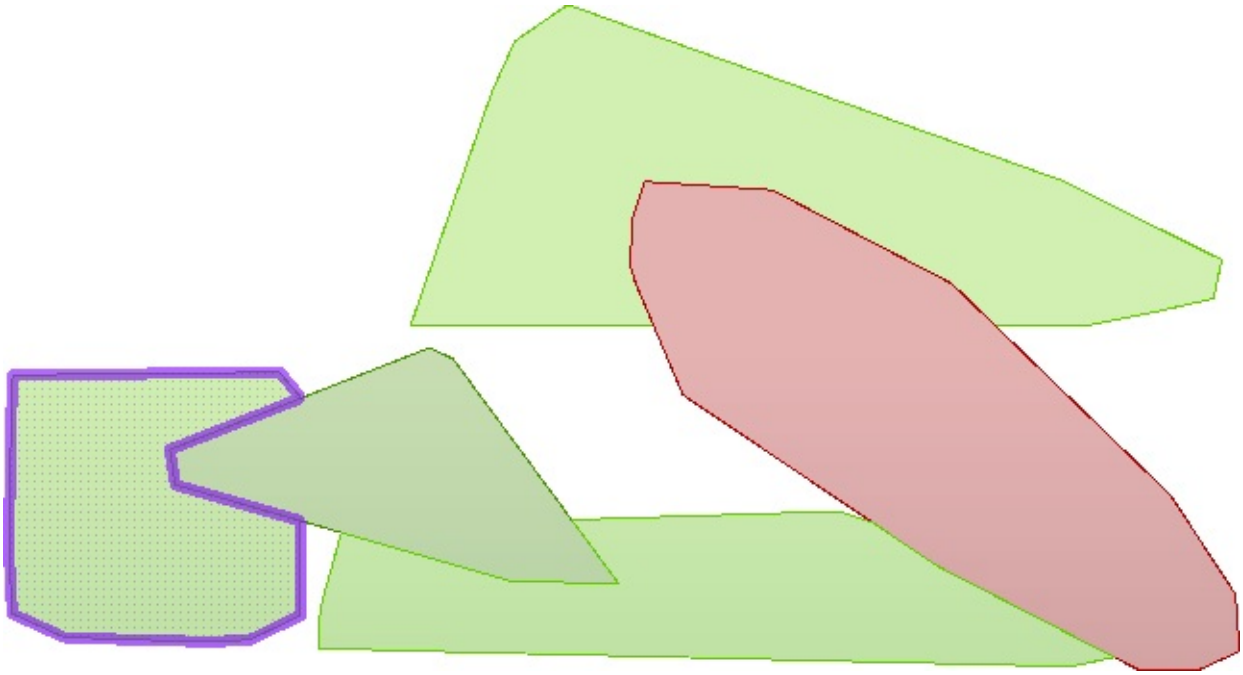
检查参数。在Group By下，选择属性_histogram.value：



系统会提示您输入值; 这是我们想要的列表中的哪个项目。我们想要第一个元素，因为它具有最多的值，因此该字段应该设置为零（默认情况下将是这样）：

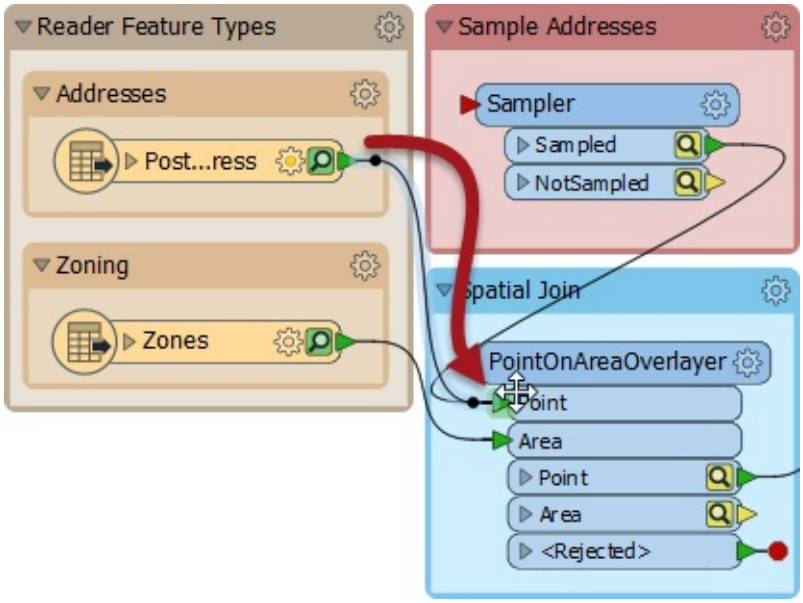


从Dissolver重新运行工作空间并检查输出：



我们现在有五个多边形要素来表示垃圾收集区域，每个区域的居民数量大致相同。将Dissolver:Area端口连接到写模块要素类型，此工作空间已完成.....差不多。

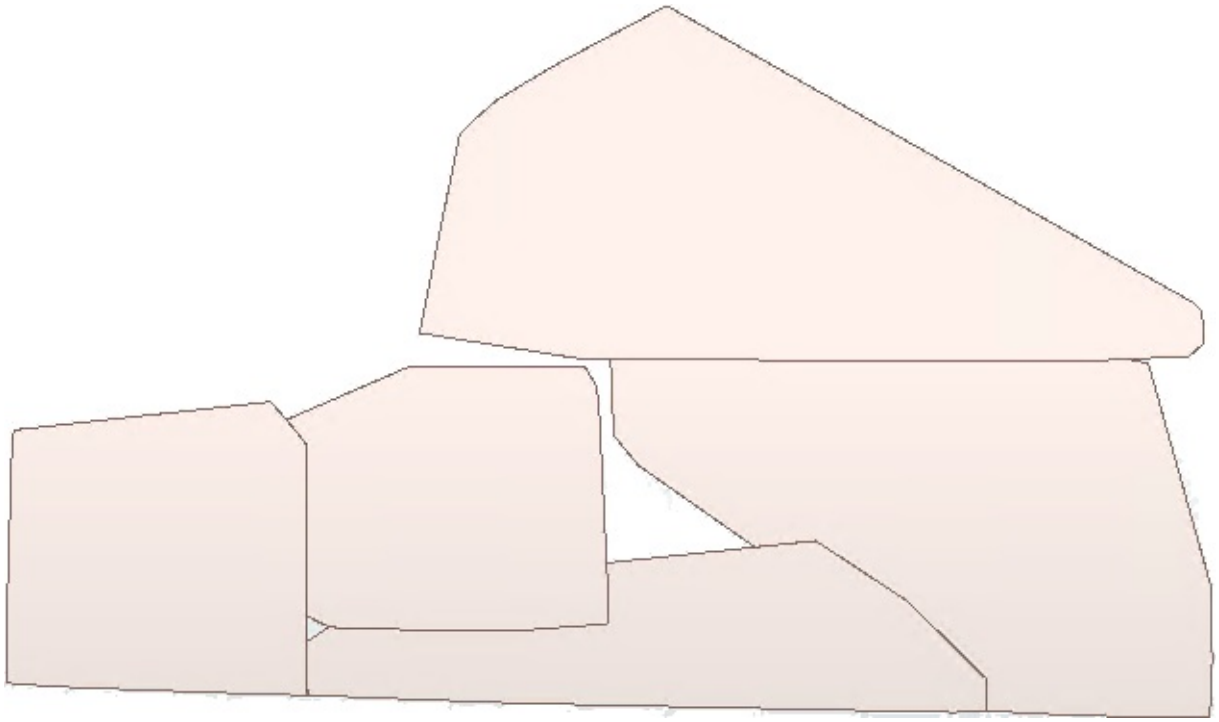
9）删除**Sampler**
要完成项目，让我们来在完整数据集上运行工作空间。您可以通过删除Sampler转换器（最简单的解决方案）或将PostalAddress-Sampler连接替换为PostalAddress-PointOnAreaOverlayer连接来执行此操作：



到目前为止，我们已将SpatialSorter的网格大小参数设置为16（即不太精确）以提高性能。现在我们已经接近完成，我们应该将网格大小增加到256（即更精确）。

出于时间和性能的原因，请关闭“使用缓存运行”选项，然后重新运行转换。

正如预期的那样，结果看起来会有所不同，现在我们正在使用完整的数据集：



恭喜

通过完成本练习，您已学会如何：

- 启用并使用要素缓存
- 使用新的转换器：HullReplacer，ListHistogrammer，Dissolver
- 通过禁用与转换器的连接来绕过转换器

模块复习

本节课旨在帮助您了解如何将转换组合在一起，创建转换的最佳方式，如何以不同方式读取数据以及如何测试工作空间的不同部分。

你应该从这个模块中学到什么

以下是本次课程要学习的重点：

理论

- 工作空间由读模块，写模块和要素类型组成。它们存在于一个层次结构中。
- 增量开发是一种构建工作空间的方法，一次一个部分。
- 工作空间可以包含任意数量的读模块和写模块 - 或者根本不包含任何读模块和写模块
- FeatureReader和FeatureWriter转换器在工作空间中的任何位置读取和写入数据
- 整体检查是一组从FME Workbench中检查数据的功能
- 部分运行允许工作空间的指定部分单独运行

FME技能

- 能够一次创建一个工作空间
- 处理工作空间中被拒绝数据的能力
- 能够将新读模块和写模块添加到工作空间
- 使用FeatureReader和FeatureWriter转换器的能力
- 从Web服务读取数据和向Web服务写入数据的能力
- 使用FME Workbench中的功能检查数据的能力
- 运行工作空间的选定部分的能力

Q + A答案

本章节没有问题

实用转换器使用

转换器是FME能力的核心，就像一个能够将原材料转化为成品的各种机器的车间。主要应用程序称为FME Workbench并非毫无意义！

为了能够有效地使用FME，能够找到合适的转换器是一项重要的技能，并且它有助于许多最常用的转换器具有共同的主题。

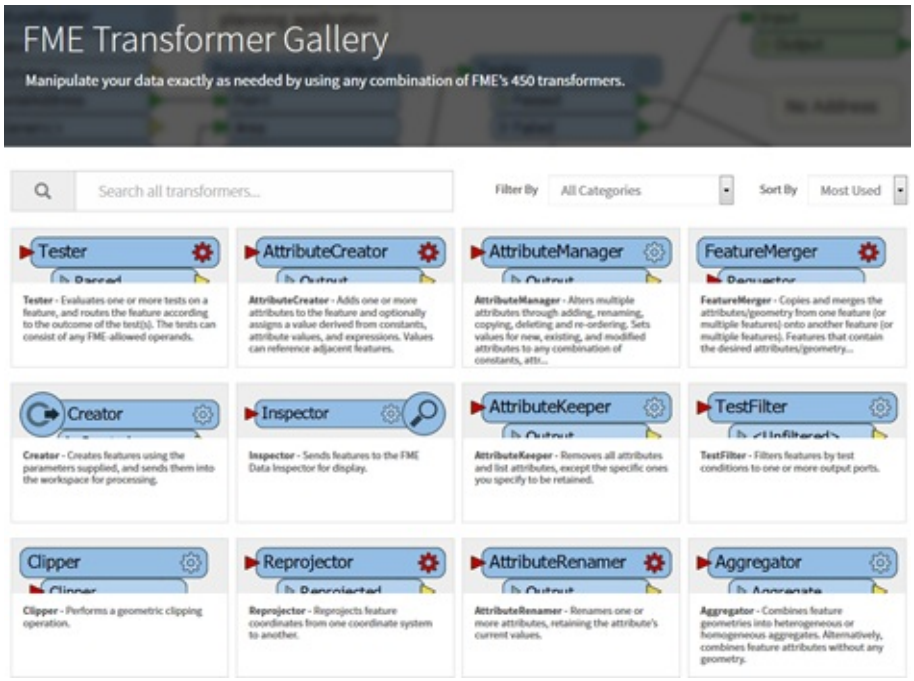
例如，许多转换器与能够在工作空间内过滤数据有关; 一旦理解了过滤数据的概念，就可以更容易地找到所需的转换器。

了解转换器的常见方面也很重要，要认识到不同转换器中的选项和参数通常以类似的方式运行。

本章介绍了查找转换器的各种搜索方法，最常用转换器的常见主题，以及如何在单个转换器对话框中构建参数值。

找到转换器

即使是经验丰富的FME用户也会发现转换器的完整列表令人望而生畏。在本节中，您将学会不再担心并喜欢转换器库。



FME拥有超过五百个转换器，具有很多功能; 可能比新用户实现的要多得多，其中大部分对他们非常有用。本节帮助您找到所需的转换器，即使您都不知道自己需要它。

虽然转换器列表看起来有点过于庞大，但不要惊慌！现实情况是，大多数用户关注的是与他们的日常工作流相关的20-30个转换器。您不需要知道每个转换器也能有效地使用FME。

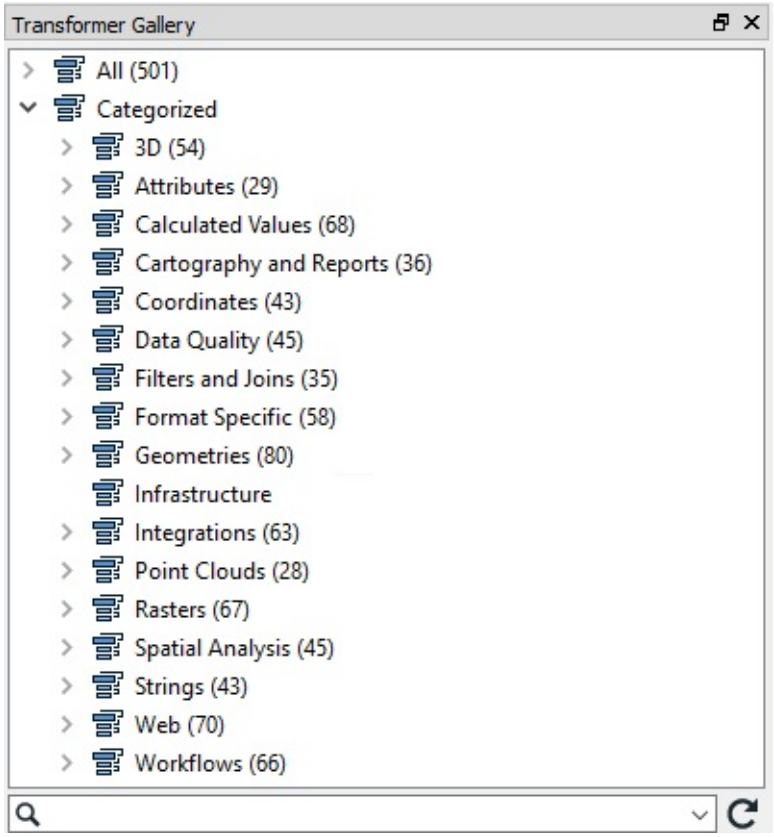
转换器库

转换器库

转换器库窗口是开始寻找转换器的明显地方。这里可以通过多种方式找到转换器。

转换器类别

转换器类别是探索转换器列表的好的起点。转换器按类别分组，以帮助找到与手头问题相关的转换器。



虽然所有这些都很重要，但最常用的转换器有以下几类：

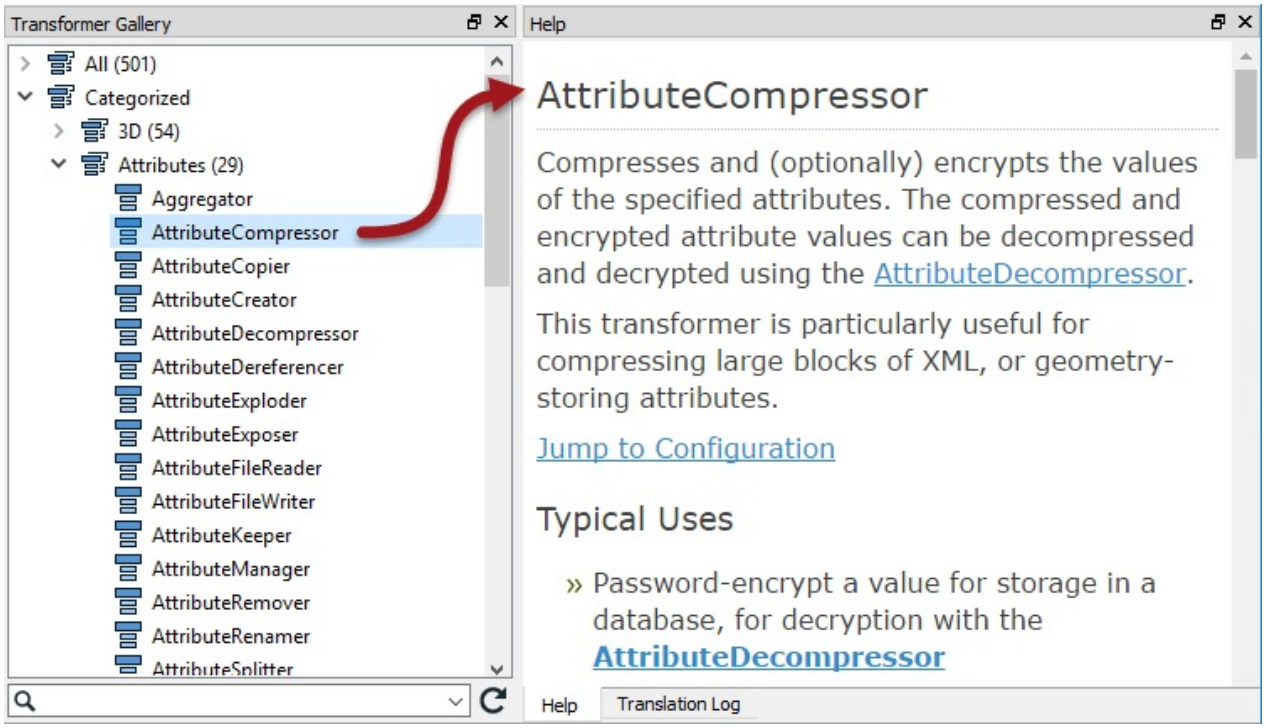
- 属性：属性/列表管理的操作
- 计算值：返回计算值的操作
- 过滤器和连接：用于分割和合并数据流的操作
- 几何：创建几何对象或将其转换为不同几何类型的操作
- 空间分析：返回空间分析结果的操作
- 字符串：操作字符串内容的操作，包括日期

只需单击展开按钮即可显示特定类别中的所有转换器。

转换器帮助

FME Workbench帮助工具显示有关转换器的信息。只需单击转换器，然后按F1键打开帮助对话框。

此工具链接到FME Workbench，以便选择的转换器（在库中或画布上）触发要在“帮助”工具中显示的内容。



技巧
另一个有用且可打印的文档是**FME**转换器参考指南。

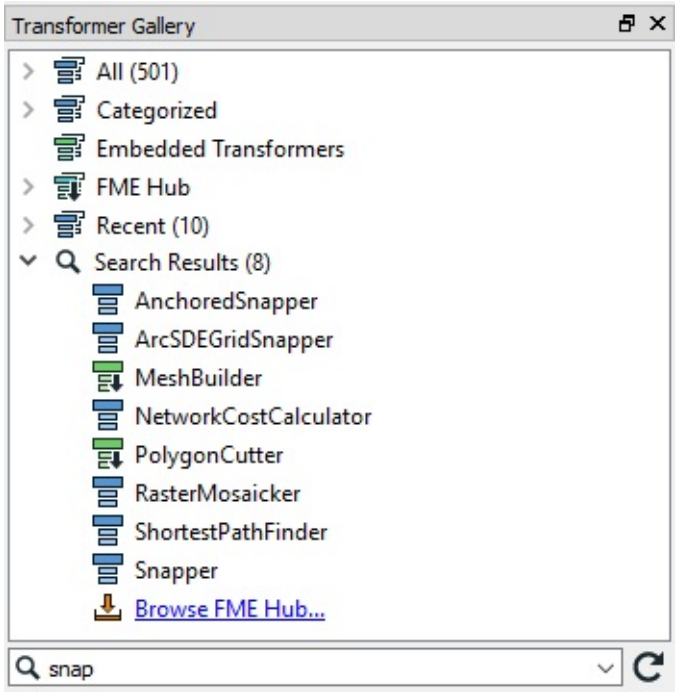
转换器搜索

转换器库和“快速添加”对话框中都有搜索功能。

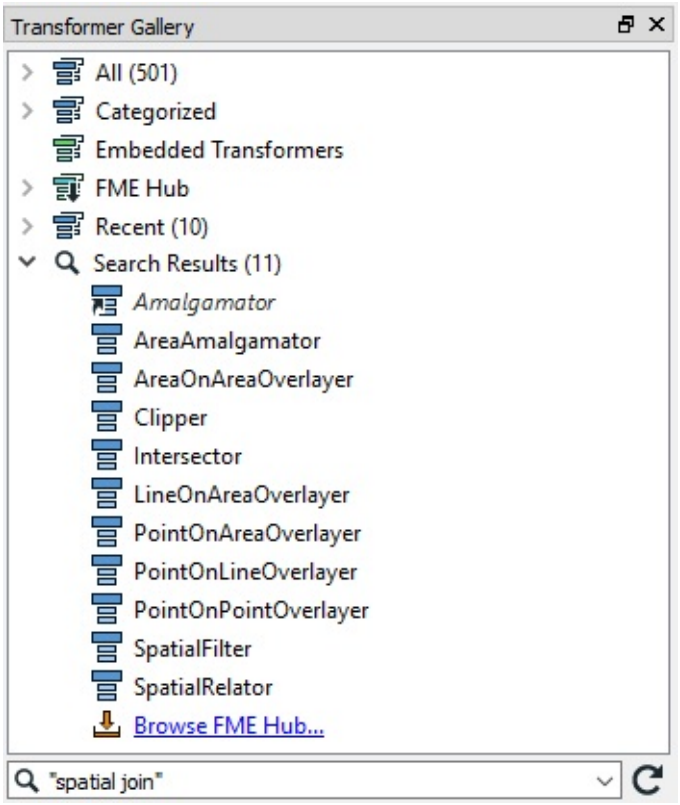
转换器库搜索

要在转换器库中执行搜索，请输入搜索词，然后按键或单击搜索图标（双筒望远镜图标）。

转换器库搜索在名称和描述中搜索。因此，搜索项可能是转换器的确切名称，或者它可能是一般性的关键字，通常指的是功能：

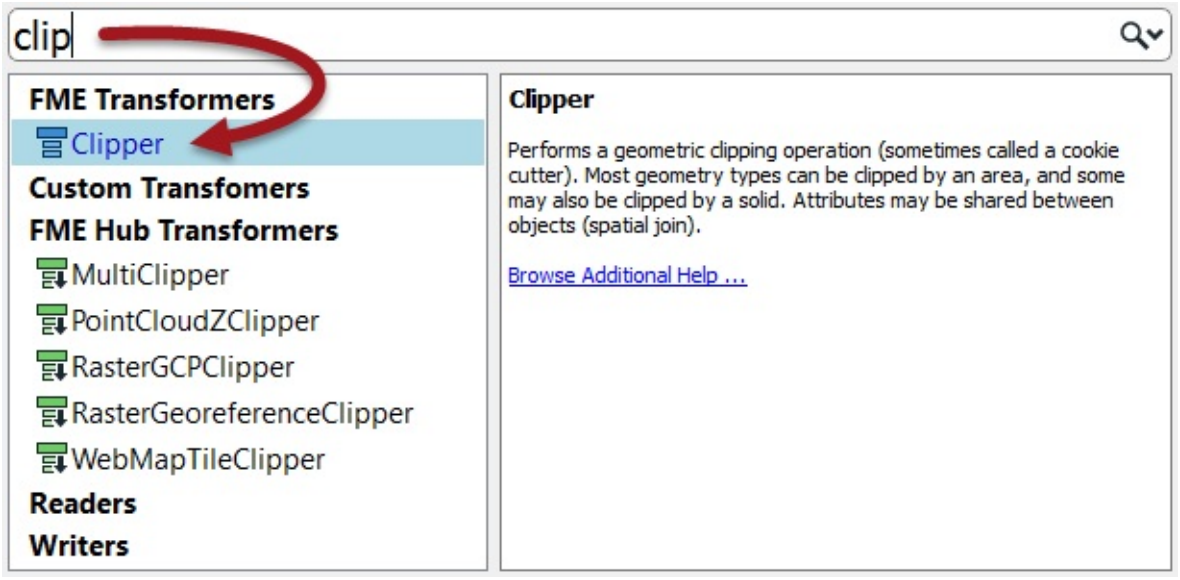


搜索字词可以是完整字词或部分字词，也可以包含多个关键字，包括用于包含单个搜索引用的引号：



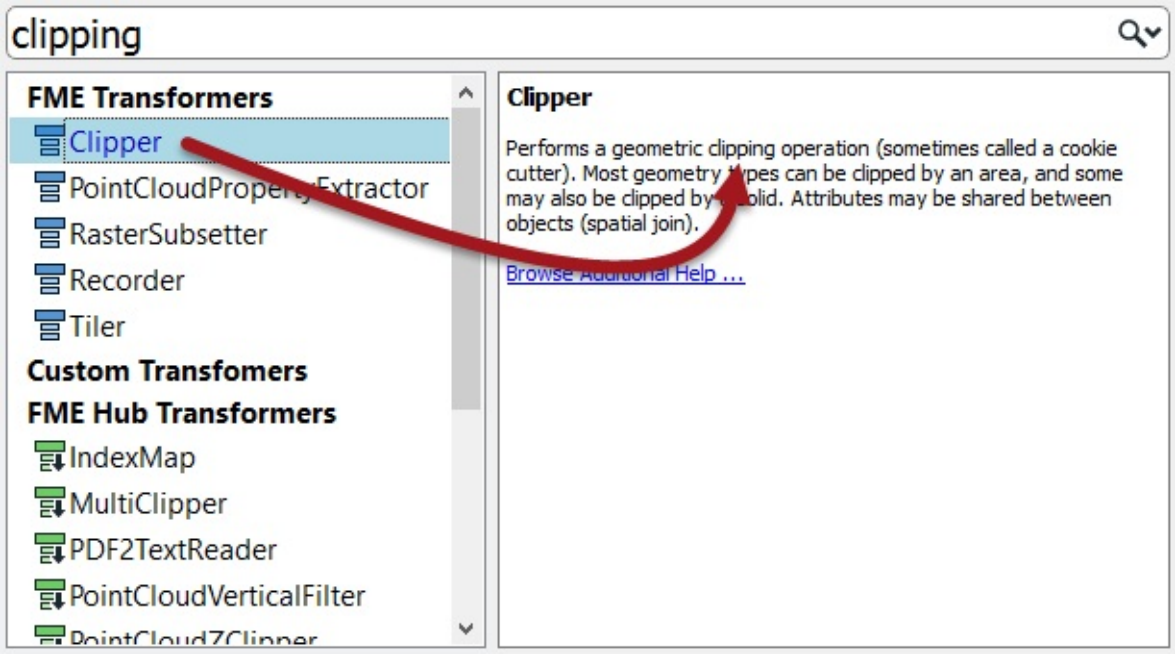
快速添加搜索

快速添加搜索字词也可以是完整或部分字词：

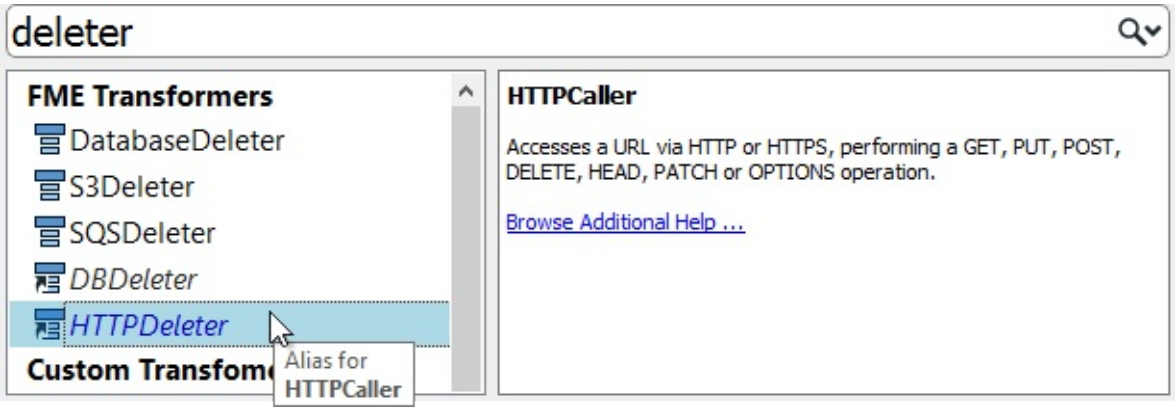


默认情况下，“快速添加”不会查找转换器描述，因此搜索项必须是转换器的实际名称：

但是，如果按<TAB>键，“快速添加”将搜索转换器描述：

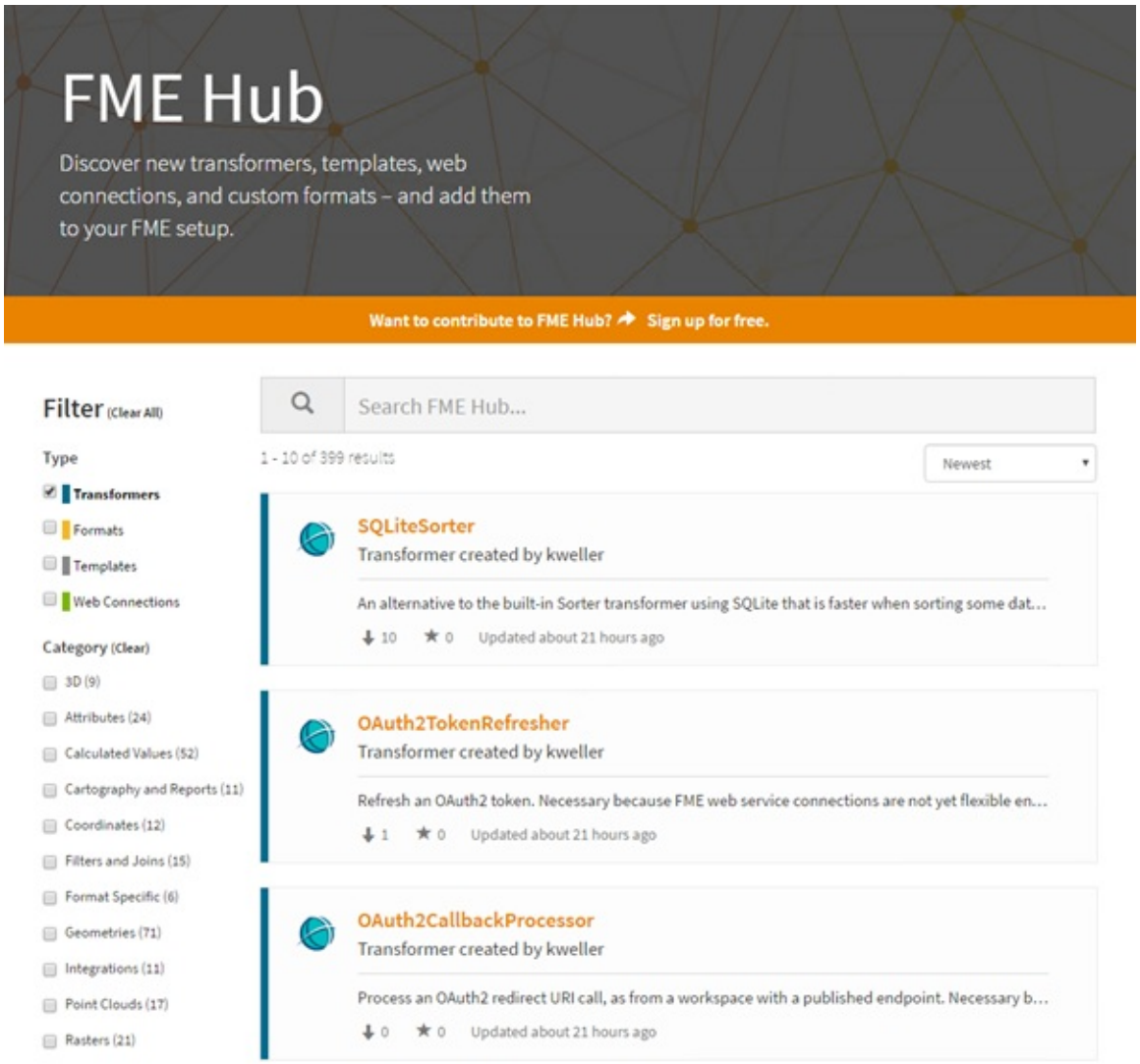


快速添加结果包括别名 - 例如，具有替代名称或已重命名的转换器 - 还包括在FME中心中找到的转换器：



消防队员**Mapp**说**.....**

以防您不知道，FME Hub（hub.safe.com）是一个用于共享FME功能的工具，例如自定义转换器，Web连接和格式： 集线器中的转换器显示在Quick Add中，带有一个小的向下 - 指向箭头，表示如果选择它们将被下载。



驼峰式命名法

快速添加还允许使用驼峰式命名法的首字母作为快捷方式。驼峰式命名法是一个单个关键字由几个连接的单词组成的单词，每个单词保留一个大写的首字母; 例如AttributeFileWriter（AFW）或ShortestPathFinder（SPF）。

Vector小姐说...

试试这些问题，看看你是否可以搜索转换器。
以下哪一项不属于转换器？

1. 属性
2. 计算
3. 数据质量
4. 工作流

这里有四个转换器和四个类别。将转换器与正确的类别匹配。

转换器	类别
Chopper	工作流
Terminator	字符串
Matcher	几何图形
DateTimeConverter	数据质量

最有价值的转换器

如果您对最常见的转换器有透彻的了解，那么您很有可能成为FME Workbench的高效用户。

任何人都可以只使用少数转换器精通FME，如果他们是正确的！

The Top 30

该转换器列表是通过用户的反馈来计算，在Safe Software网站上按照被使用最多的顺寻来排序。掌握这些信息可以告诉我们在哪些方面指导我们的开发工作进行改进，但它也让用户了解他们在工作中最需要的（500多个）FME转换器中的哪一个。

下表（最后更新于2018年6月）提供了最常用的30个转换器列表。Tester转换器每年都在列表中排名第一，突出了它的重要性。

排名	转换器	排名**k**	转换器
1	Tester	16	VertexCreator
2	AttributeCreator	17	Counter
3	AttributeManager	18	DuplicateFilter
4	FeatureMerger (FeatureJoiner)	19	StringReplacer
5	Inspector	20	StatisticsCalculator
6	Creator	21	SpatialFilter
7	AttributeKeeper	22	GeometryFilter
8	TestFilter	23	Bufferer
9	Reprojector	24	Sorter
10	Clipper	25	AttributeExposer
11	AttributeRenamer	26	Dissolver
12	AttributeFilter	27	ListExploder
13	Aggregator	28	AttributeSplitter
14	AttributeRemover	29	CoordinateExtractor
15	FeatureReader	30	FeatureWriter

除了明显用于转换几何对象的转换器（Clipper，Bufferer，Dissolver）和明显用于转换属性值的转换器（StringReplacer，Counter）之外，还有一些其他不同的转换器组。

新知识
FeatureJoiner是FME2018的新转换器，旨在最终取代FeatureMerger。

管理属性

这些转换器 - 通常命名为 *Attribute<Something>* - 主要用于管理模式映射的属性（创建，重命名和删除）。但是，它们也可用于设置新属性值或更新现有属性值。

过滤

这些转换器 - 通常命名为 *<Something>Filter* - 在流经工作空间时细分数据。通常filte是条件过滤器，其中决定哪些要素输出到哪个连接由某种形式的测试或条件决定。

数据连接

连接是过滤的相反动作; 它们的作用是当单独的数据流在工作空间中流动时组合在一起。 与过滤一样，有一个条件要满足 - 在这种情况下匹配键值 - 确定连接发生的方式和位置。

练习：转换器选择练习

练习**1**	转换器选择练习
数据	无
总体目标	练习搜索和选择转换器
启动工作空间	无
结束工作空间	无

您已申请成为**FME**认证专家。为了帮助您为考试练习，您的同事已经给您一些问题来测试您搜索和选择转换器的知识。

你能回答吗？

1）最常用的**Web**转换器
您的同事想要知道的第一件事是，Web类别中最常用的五大转换器是什么？

2）快速添加
这里显示的是随机字符列表：

- lineco
- reac
- rbic
- drape
- attributeexpl

第二个问题是，对于每个字符串，Quick Add返回哪些转换器，不用去试你知道他们中的任何一个会是什么吗？？

3）转换器搜索
第三，您的同事已经提出了一个不同场景的列表，并希望您搜索转换器来执行它们：

- 我们在文件中有一些文本行，然后希望读取该文本并将其添加为属性。
- 我们有一组矢量轮廓，并希望通过转置X和Z坐标来创建横截面。
- 我们有一个点云数据集，并希望通过重新采样来减少其大小以消除多余的点。
- 我们有一个文本字符串，想要找出字符串包含多少个字符。
- 我们有一组地址，并且每个地址都想找到最近的两个图书馆。

你的同事说**.....**
我们问题的答案可以在本章末尾找到.....

管理属性

前30个转换器中的很大一部分是用于管理属性的支持类转换器。这些转换器创建新属性，重命名它们，设置值并删除它们。

这些转换器的关键用途是重命名模式映射的属性。

属性管理转换器

主要的属性管理任务和可以使用的转换器如下：

任务	转换器
创建属性	AttributeCreator，AttributeManager
设置属性值	AttributeCreator，AttributeManager
删除属性	AttributeKeeper，AttributeManager，AttributeRemover，BulkAttributeRemover
重命名属性	AttributeManager，AttributeRenamer，BulkAttributeRenamer
复制属性	AttributeCopier，AttributeCreator，AttributeManager
排序属性	AttributeManager
更改属性大小写	BulkAttributeRenamer
添加前缀/后缀	BulkAttributeRenamer

这些转换器中的许多都能执行类似的操作，您可以看到AttributeManager完成了许多任务，您几乎可以完全使用它。

警告
不要误解BulkAttributeRenamer。它改变大小写 - 或者添加后缀/前缀 - 作用于属性名称，而不是属性值。

列表

FME中的列表是一种机制，允许每个属性有多个值。因此，在属性*myAttribute*只能包含单个值的情况下，列表属性*myList{}.myAttribute*可以包含多个值：

```
myList{0}.myAttribute
myList{1}.myAttribute
myList{2}.myAttribute
```

例如，表示森林区域的单个多边形可能具有list属性来记录该区域中包含的树种：

```
TreeList{0}.Species = Oak
TreeList{1}.Species = Chestnut
TreeList{2}.Species = Ash
```

各种转换器可以创建列表，其他转换器包括对列表的支持，并且有些转换器专门用于列表属性（例如ListSorter）。

有关进一步阅读，请查看Safe Software博客上有关属性管理的文章。

创建和设置属性

创建属性和设置值可能是FME中使用的主要属性功能。创建属性时，可以通过多种方式设置其值。

能够创建属性并设置其值的转换器是：

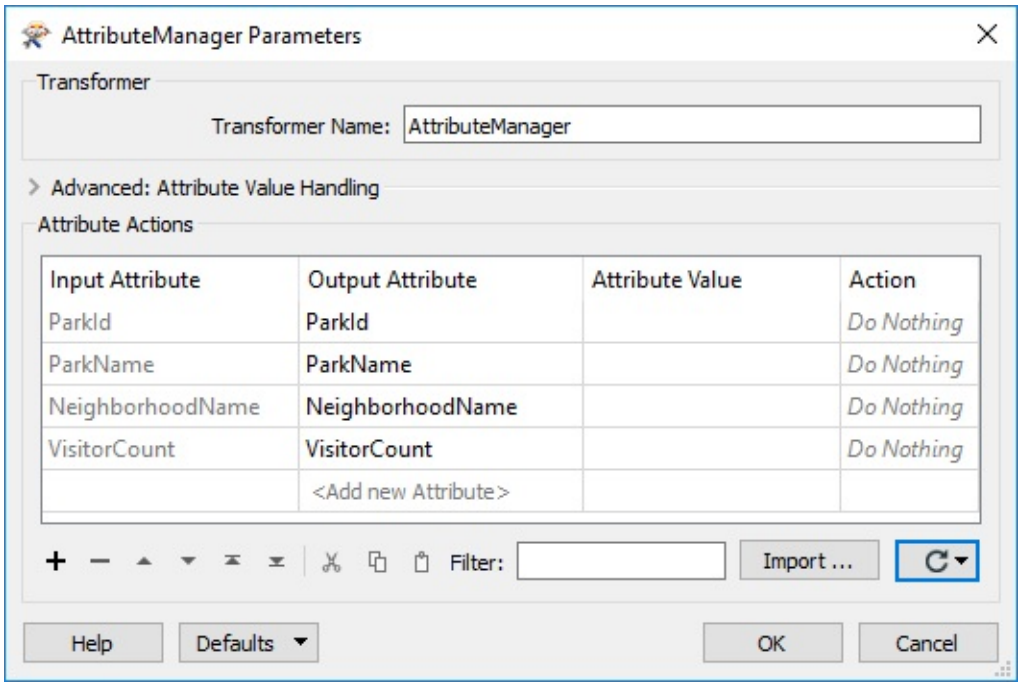
- AttributeCreator
- AttributeManager

注意: *The AttributeCopier*和*AttributeRenamer transformers*转换器可以设置属性值，但仅限于属性尚不存在的位置。

The AttributeManager

对于大多数操作，我们将集中于AttributeManager，下面是该转换器的简要概述。

AttributeManager参数对话框有许多字段：输入属性，输出属性，属性值和操作。在属性处理转换器中，它会自动填充与其相关的属性的详细信息：

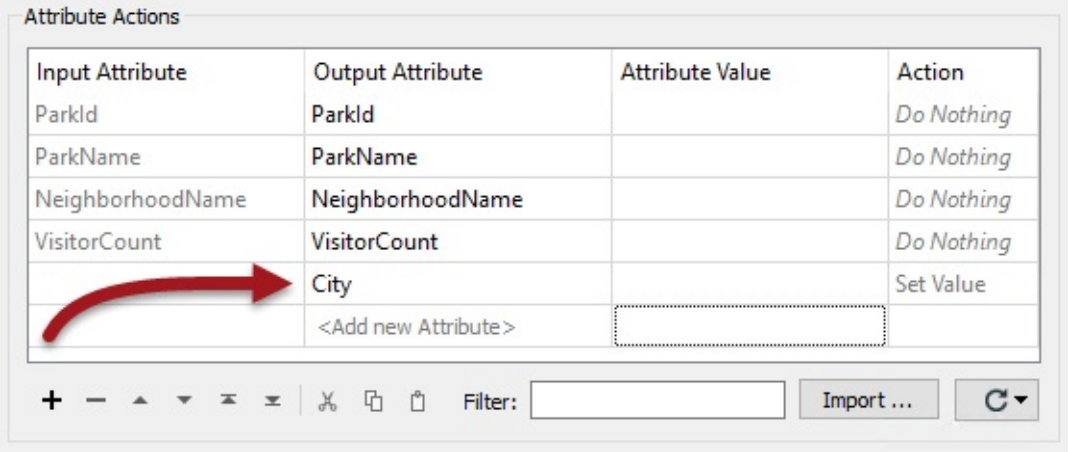


操作字段可以由用户设置，但也可以在对其他字段进行更改时自动设置。

新知识
AttributeManager在FME2018中有新选项，用于在对话框中剪切/复制/粘贴/复制单个行。

手动创建属性

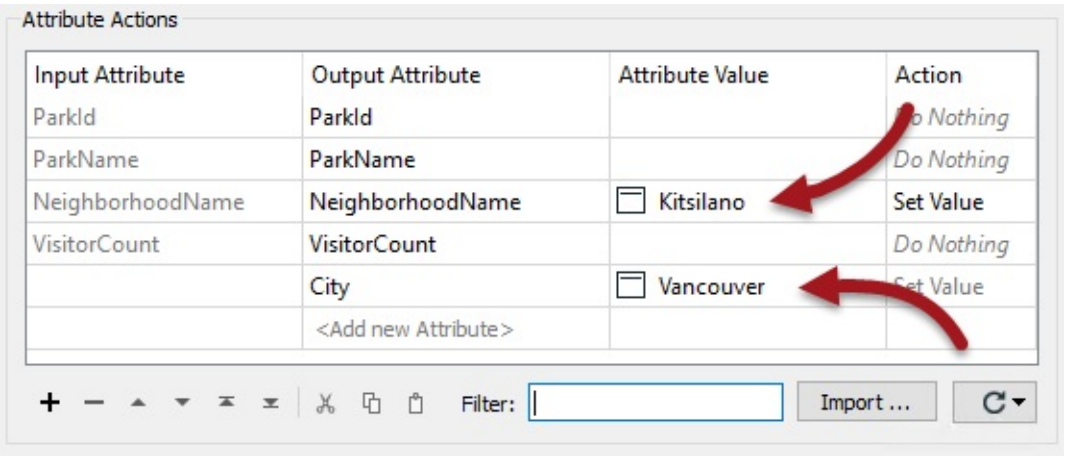
通过在“输出属性”字段中输入新的属性名称，它将在输出中创建。



文本<添加新属性>（< Add new Attribute >）突出显示可以创建新属性的位置。默认情况下，当“属性值”字段为空时，新属性没有值。

设置固定属性值

只需在“属性值”字段中输入值，即可创建属性的固定（或常量）值：



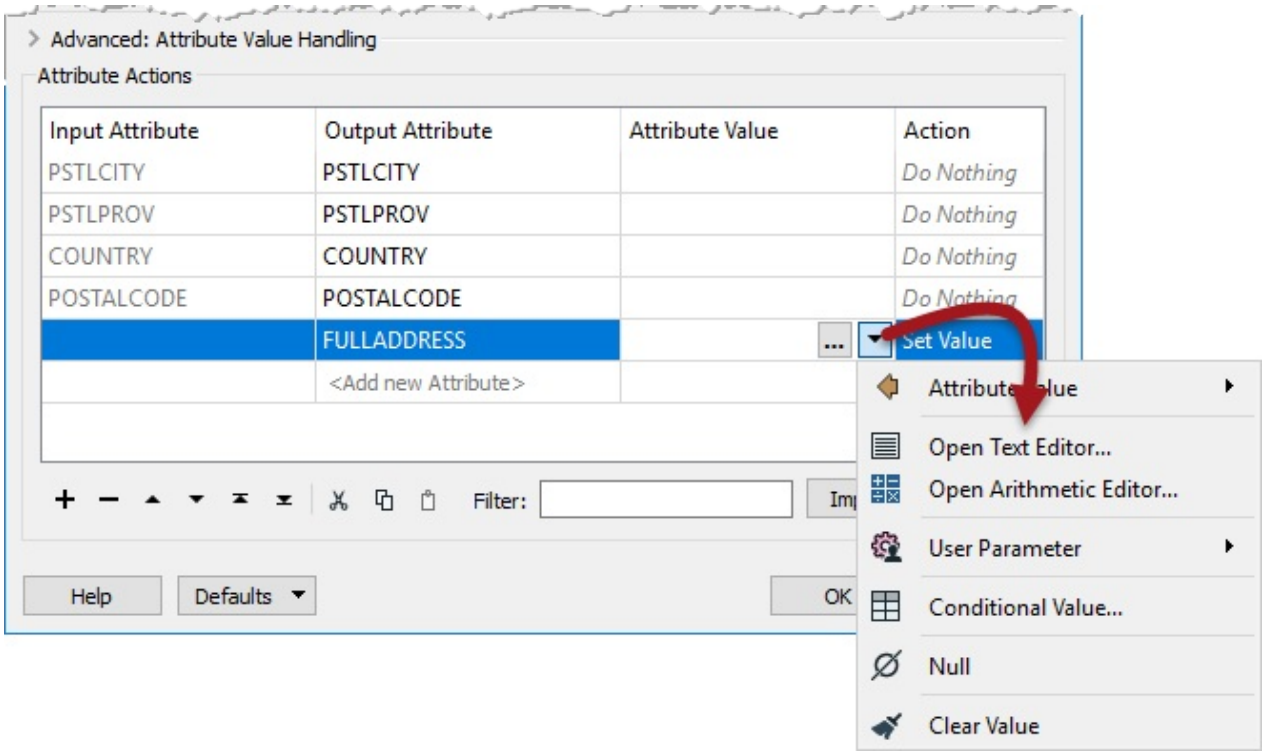
例如，在这里，名为City的新属性被赋予温哥华的固定值。

但是，还要注意现有属性NeighborhoodName也被赋予固定值。它被赋予了“Kitsilano”的值。请注意，通过在该字段中输入值，Action字段已自动从“ Do Nothing ”更改为“Set Value”。

除了像这样输入设定值之外，还可以用多种方式构造属性值.....

构造属性

除了常量属性值，FME还允许您使用字符串操作和算术计算来构造值。通过单击“属性值”字段中的箭头并选择“打开文本编辑器”或“打开算术编辑器”来实现此过程：

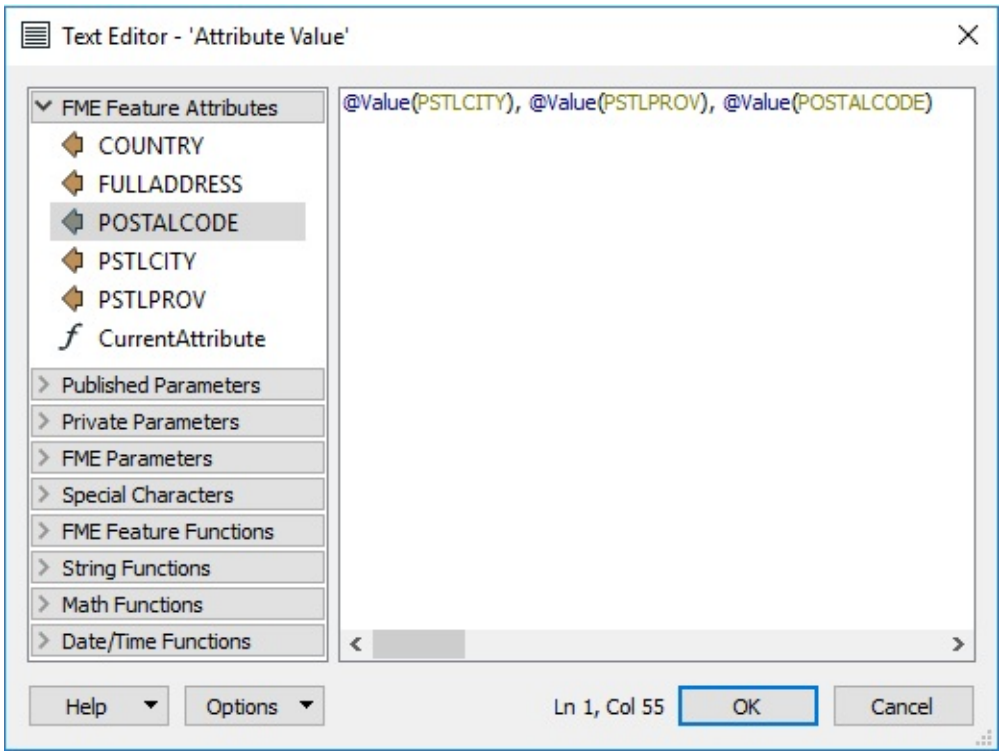


此方法非常有用，因为该属性不再需要是固定值：它可以由现有属性，参数和常量的混合构成。

文本编辑器

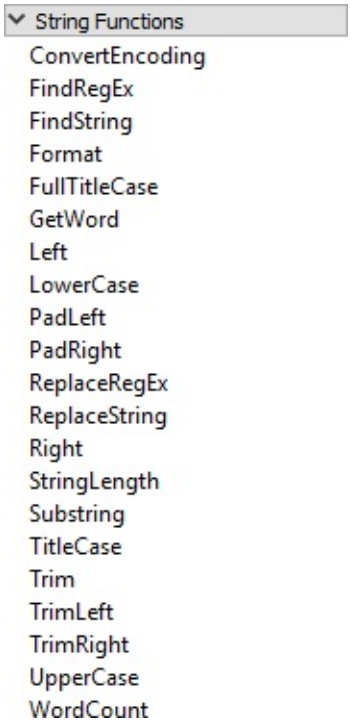
文本编辑器 - 正如您所期望的那样 - 允许您构造文本值。它包括您需要的所有常用字符串处理功能，例如连接，修剪，填充和大小写更改。

文本编辑器如下所示：

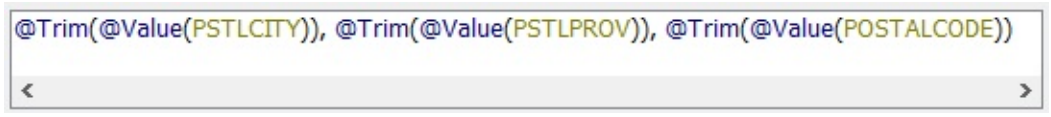


在这里，用户通过将各种现有属性与一些固定字符（逗号）连接来构造地址字符串。

注意左侧的菜单。此处列出了现有属性，并通过双击将其添加到字符串中。另外，请注意其他菜单选项。也许文本最重要的是字符串函数：



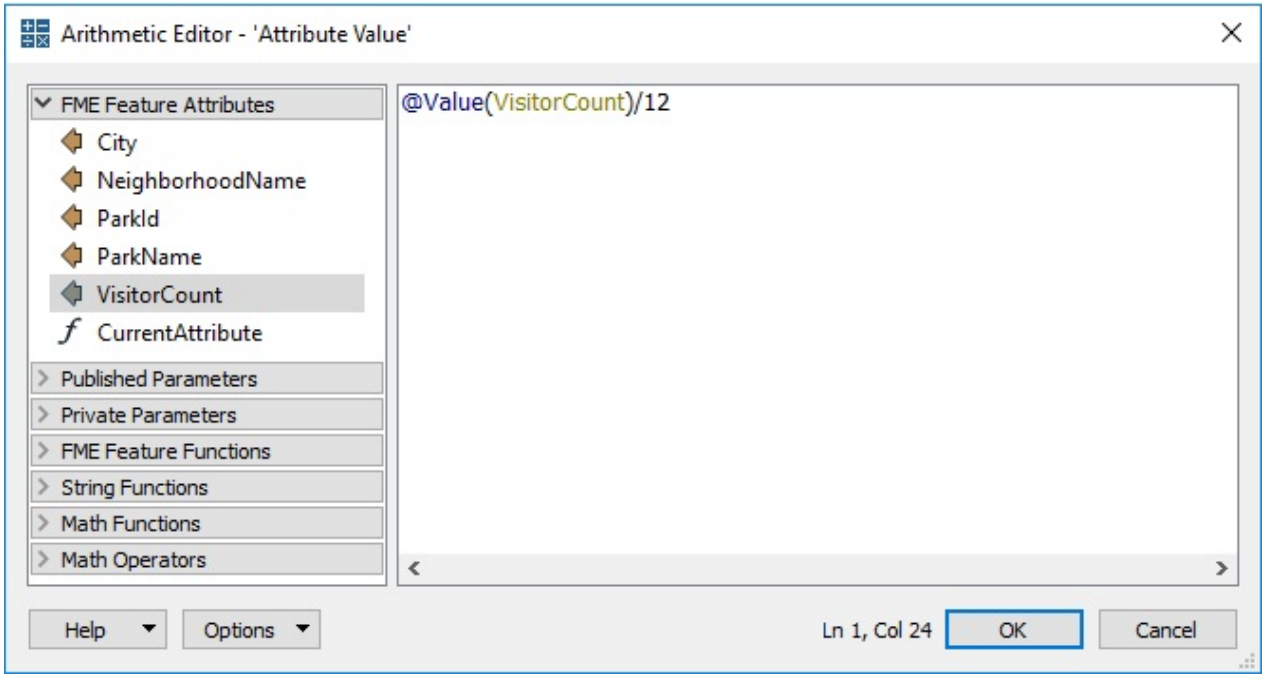
这些是可用于操做正在使用的字符串的函数。例如，在这里，用户要确保使用的属性在使用时被裁剪:



技巧
注意文本编辑器中的日期/时间函数，可用于操作日期，时间，日期时间字符串，包括TimeZone组件。

算术编辑器

算术编辑器与文本编辑器非常相似，只是输入对话框的内容将被计算为算术表达式，并返回数值结果：

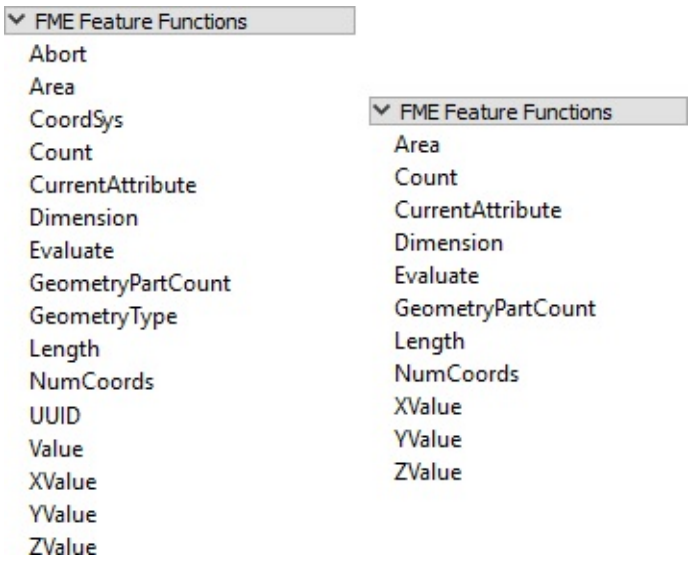


在这里，用户通过将每年的访客数量除以12来计算每月的公园访客数量。与文本编辑器一样，现有属性和算术函数是从左侧的菜单中获得的。

警告
算术编辑器的内容 必须 形成一个可以用数学方法计算的算术表达式。

FME要素函数

文本和算术编辑器菜单中的另一个项目是FME要素函数：



这些函数可以触及FME核心功能的核心。它们是构建转换器的基石; 是可以将值返回给编辑器的基本功能。

例如，@ Area（）函数返回当前要素的面积（假设它是多边形）。@CoordSys（）返回坐标系。它们是AreaCalculator和CoordinateSystemExtractor转换器的功能等价物。

一些函数返回字符串; 其他返回数值。因此，可用的函数取决于是使用文本编辑器还是算术编辑器。在上面的屏幕截图中，文本编辑器函数位于左侧，算术编辑器函数位于右侧。文本编辑器可以使用文本或数值; 算术编辑器只能接受数值。

FME要素函数非常有用，因为它们允许您直接在AttributeManager中构建处理，而不是使用单独的转换器。

替换其他转换器

集成的文本和算术编辑器为工作空间创建提供了很大的好处。它们允许属性创建函数直接在单个转换器中执行。

例如，AttributeManager文本编辑器可以用作StringConcatenator和ExpressionEvaluator转换器的直接替代。

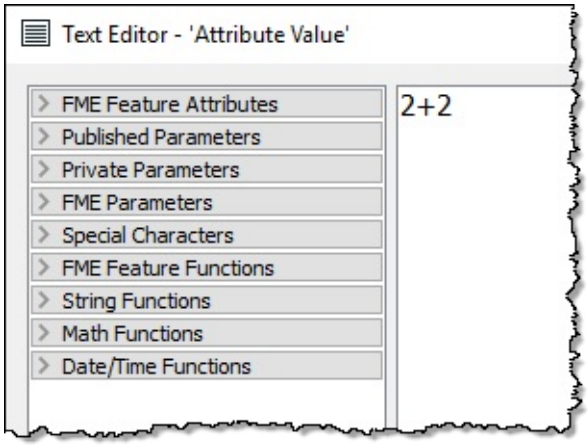
AttributeManager还可以替换StringPadder和AttributeTrimmer转换器，尽管用户友好性稍差。如果在编辑器中使用FME要素函数，则此转换器也可以在技术上替换转换器，例如AreaCalculator，LengthCalculator，CoordinateCounter，TimeStamper等等。

这个特性通常是件好事。当尽可能多的外围操作直接集成到单个转换器中时，工作空间将更紧凑，定义更明确。但是，由于AttributeManager可以执行许多操作，因此使用最佳实践并确保其具有正确的注释也更为重要。

如果没有正确注释AttributeManager，则无法通过查看Workbench画布确定它正在执行的操作！

Vector小姐说.....

这是一个问题，看看你是否注意到了。查看编辑对话框的屏幕截图，告诉我返回属性的值是：



1. 2 + 2 2. 4 3. 4.0 4.错误！

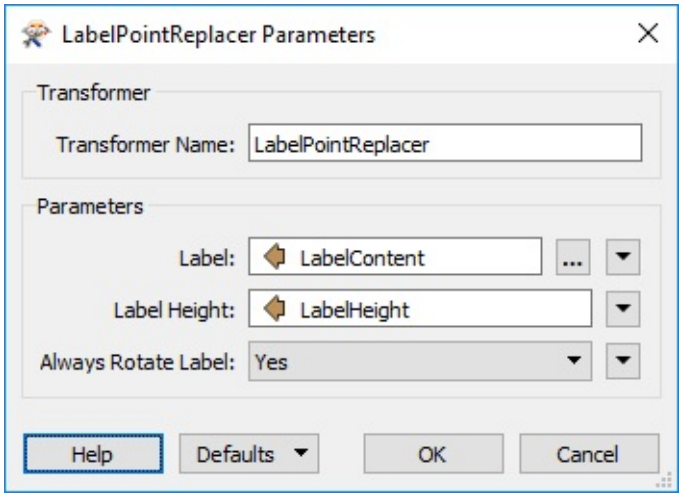
构建转换器参数

让我们暂时将属性放在一边，看一下转换器参数。

转换器参数通常以固定方式设置（硬编码）或设置为采用特定属性的值。但是，与构建属性的方式相同，可以使用文本或算术编辑器为转换器参数构建值。

使用参数属性

如上所述，大多数转换器参数允许用户选择属性值而不是手动输入固定值。例如，LabelPointReplacer可以创建一个标签，其内容和高度由属性值指定：

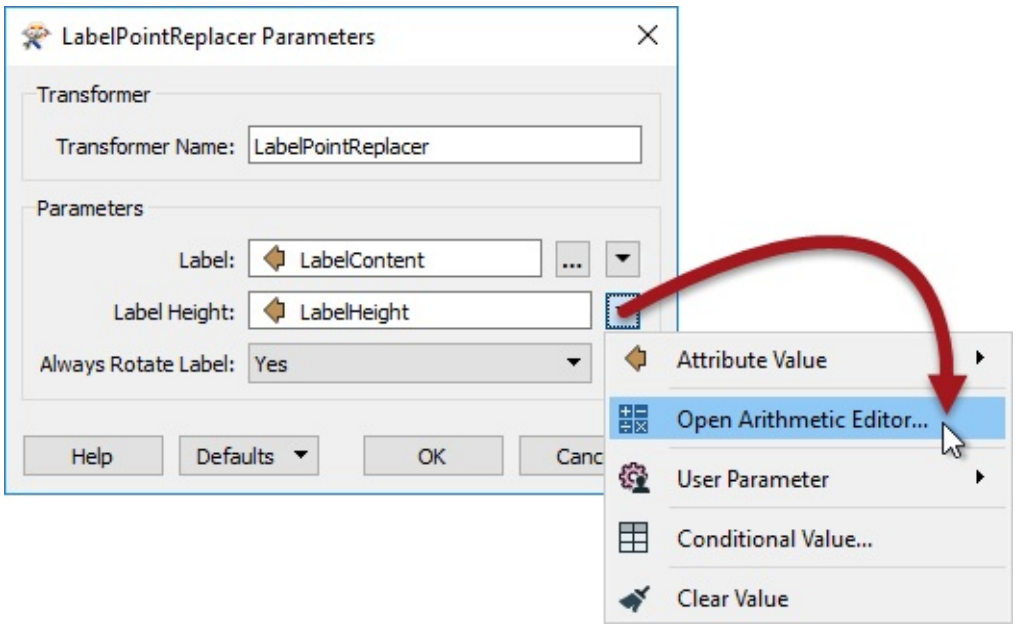


此特性很有用，因为它允许参数（例如标签大小）为每个要素获取不同的值。可以从源数据集中读取属性，也可以使用ExpressionEvaluator计算属性，以便一个要素创建高度为十个单位的标签，另一个要素创建高15个单位的标签，依此类推。它不再是固定值。

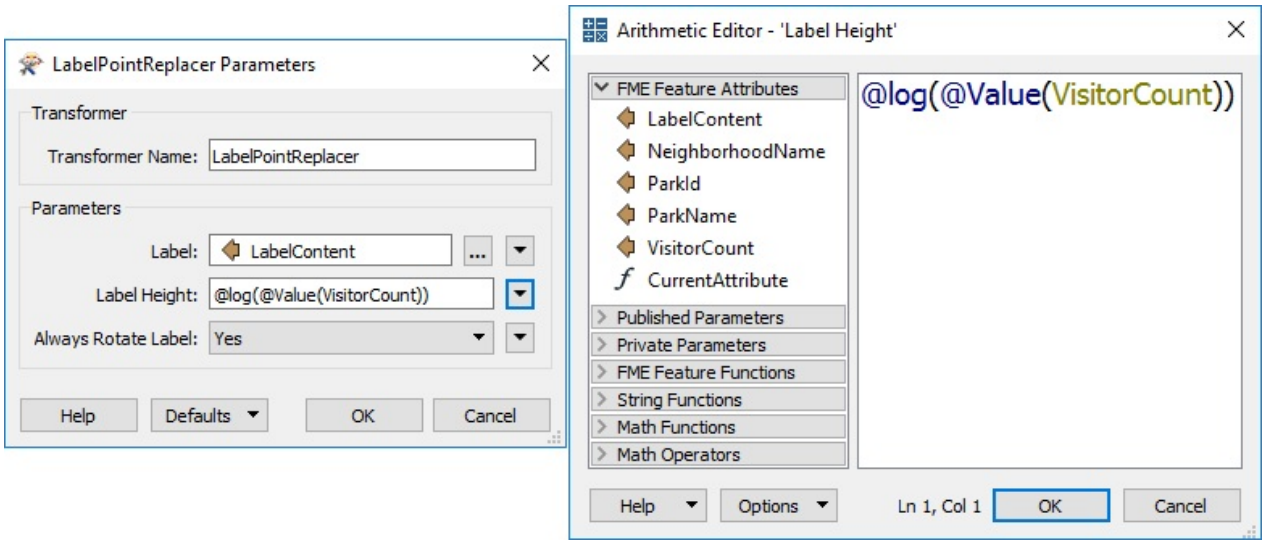
构建参数值

如果需要计算或构建参数值，而不是使用单独的转换器，FME已将集成的字符串和数字编辑器内置到参数对话框中。

例如，这里用户选择使用算术计算器计算标签高度：



计算器允许选择和使用FME属性，其他参数，以及数学和基于字符串的函数。例如，此处用户已选择使用公园的访客计数的对数来计算其标签的高度：



E. Dict先生（FME法务律师）说.....

编辑器对话框可用的固定规则取决于所设置参数的类型。例如，LabelPointReplacer中的Label参数打开文本编辑器，因为该参数需要文本值。Label Height参数打开算术编辑器，因为该参数需要数值。

减少工作空间拥塞

就像构建属性值一样，当尽可能多的外围操作直接集成到单个转换器或参数中时，工作空间更紧凑。但是，与属性一样，添加适当的注释很重要，否则一个普通的观察者很难理解工作空间的意图。

另一个缺点是，特别是对于参数来说，您也不能获取信息作为属性以在其他地方使用。例如，您如果在LabelPointReplacer中构建一个标签字符串，则该字符串不可用作工作空间中其他位置的属性。

重命名和复制属性

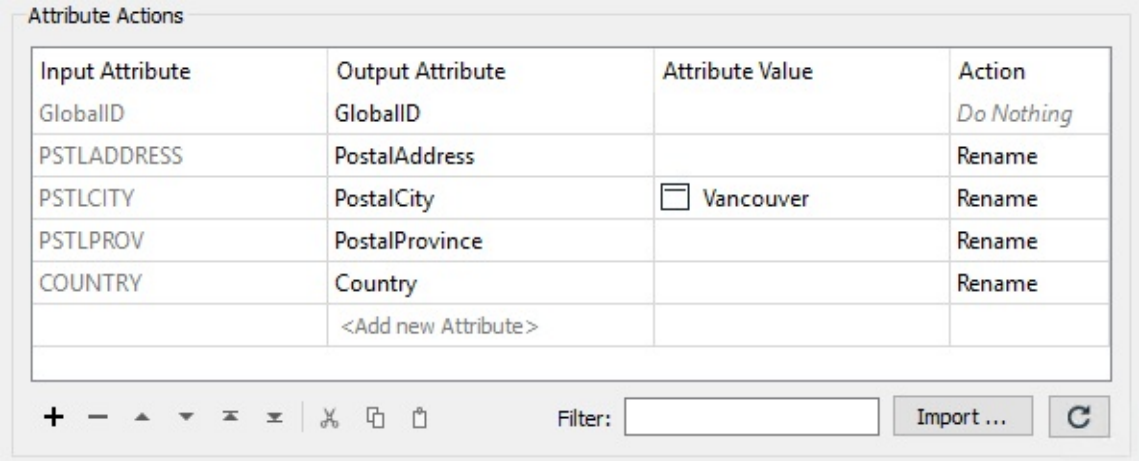
重命名和 - 在较小程度上 - 复制属性也是FME中的关键属性功能。重命名属性时，它在其先前名称下不再存在; 复制时，它既存在于新名称中，也存在于旧名称中。

能够重命名属性的转换器是：

转换器	功能
AttributeCopier	复制
AttributeCreator	复制
AttributeManager	复制并重命名
AttributeRenamer	重命名

重命名

重命名的基本目的是手动为所选属性输入新名称。旧属性将被删除并替换为新命名的属性：

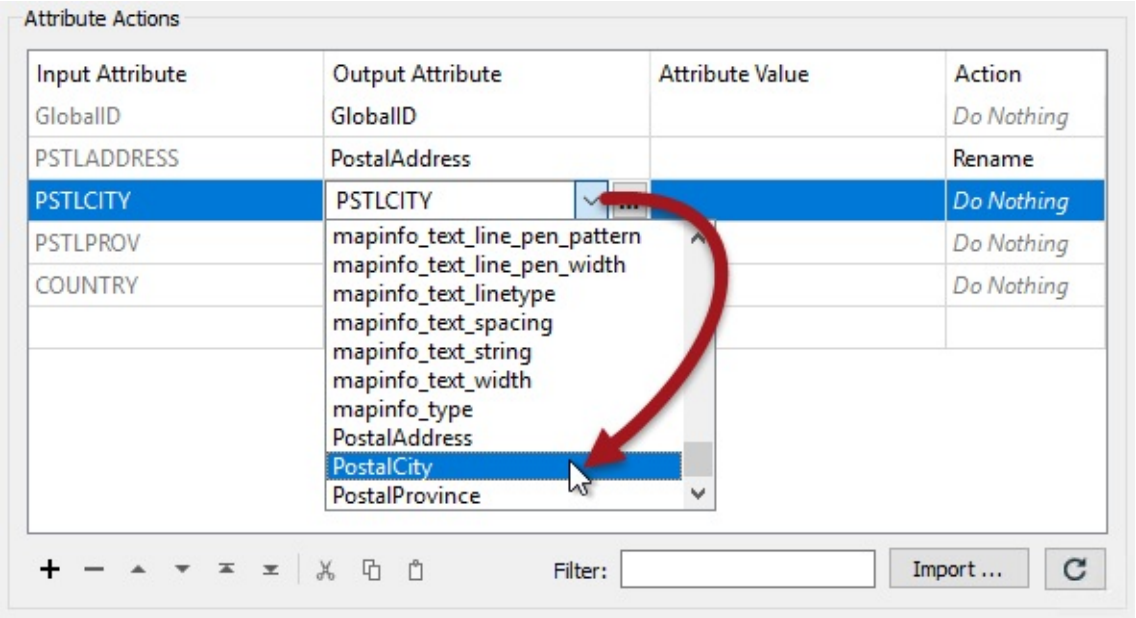


这里，AttributeManager用于通过为输出属性输入不同的名称来重命名多个字段。操作自动设置为重命名。请注意，用户还为PSTLCITY / PostalCity属性输入新的常量值。

当读模块模式（'我们拥有的'）需要重命名以匹配写模块模式（'我们想要的'）时，这种行为显然是有用的。

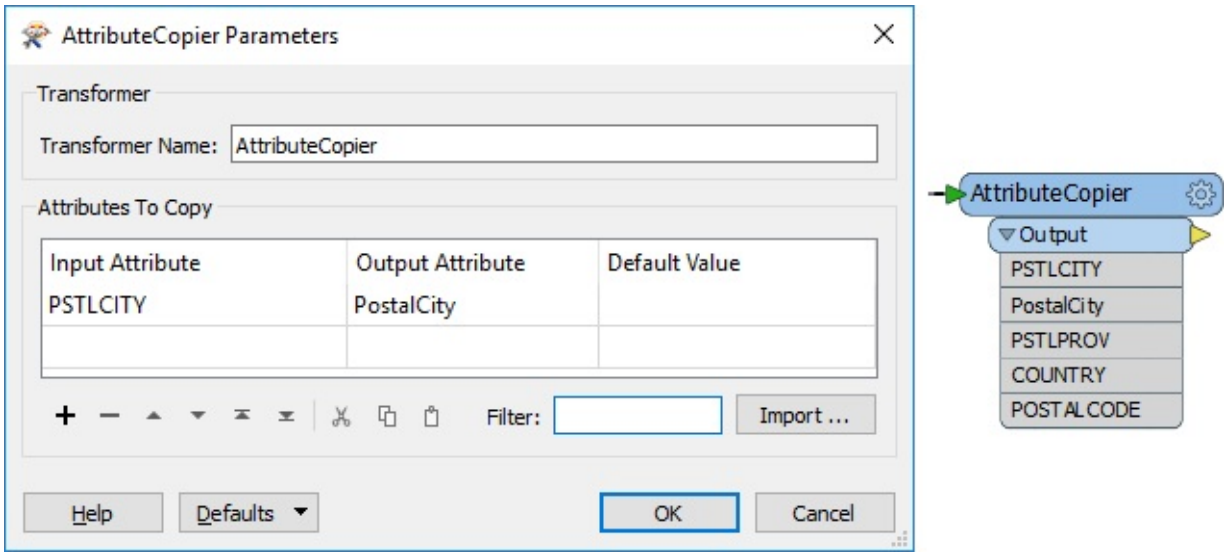
技巧

虽然您可以在“输出属性”字段中手动键入新属性名称，但如果转换器连接到具有正确属性的写模块要素类型，则其属性名称将自动可供选择：



复制

根据转换器的不同，复制属性可以是两种样式之一。



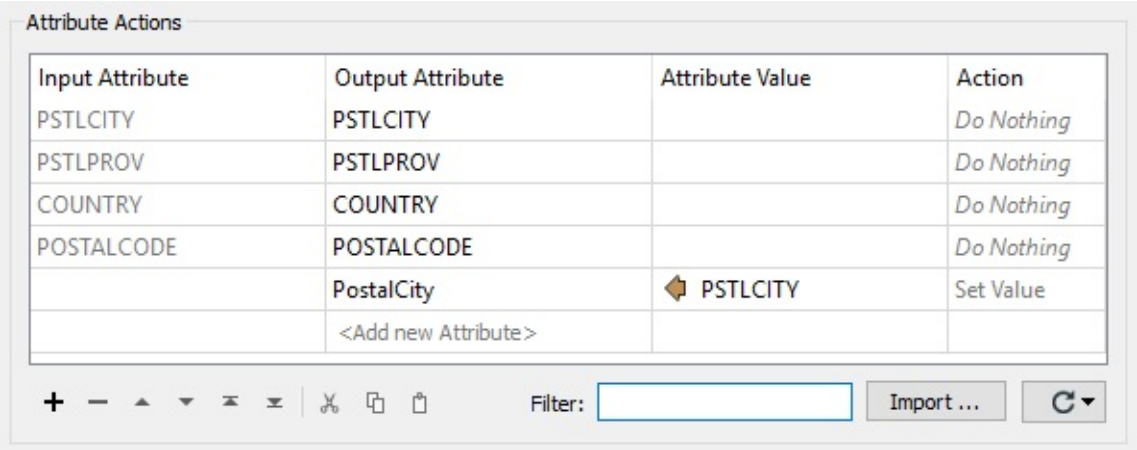
这里的AttributeCopier包括选择现有属性并为其输入新名称。同样，当连接到写模块要素类型时，可以使用其模式。

注意转换器输出上的PSTLCITY和PostalCity是如何存在的，证明它正在复制属性而不是重命名它。

技巧

您可以（如上所述）在AttributeCopier中输入常量属性值，但实际上在这种情况下它几乎不是复制操作; 它更像是一个属性创建任务。

对于其他转换器，设置样式相反：创建新属性并赋予现有属性的值：

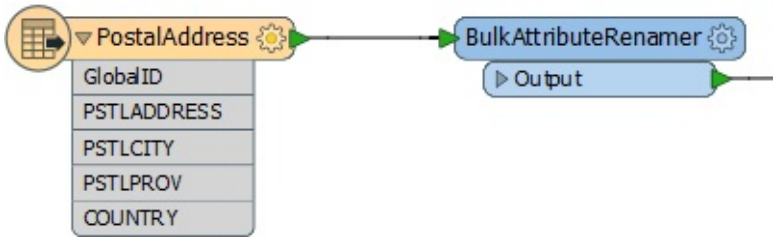


在此AttributeManager转换器中，用户创建一个新属性（PostalCity）并为其分配另一个属性（PSTLCITY）。实际上，他们已经制作了原始属性的副本。

批量属性重命名

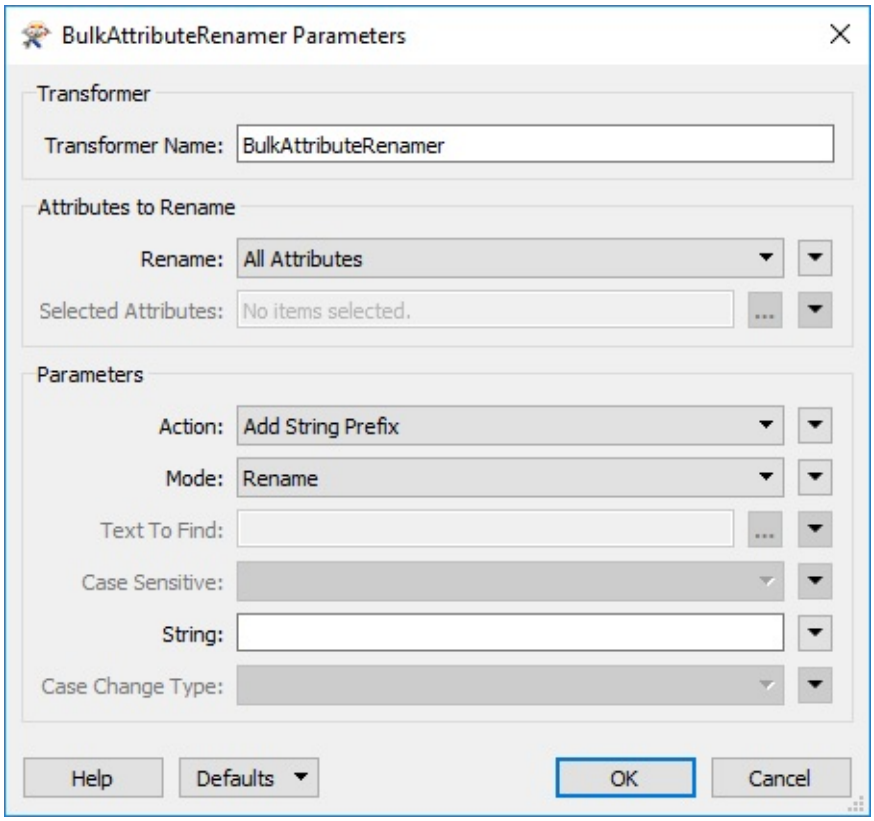
通常的属性重命名涉及选择要修改的各个属性。但是，在某些情况下，能够对大量属性执行相同的重命名操作非常重要。

这种情况由BulkAttributeRenamer转换器提供。



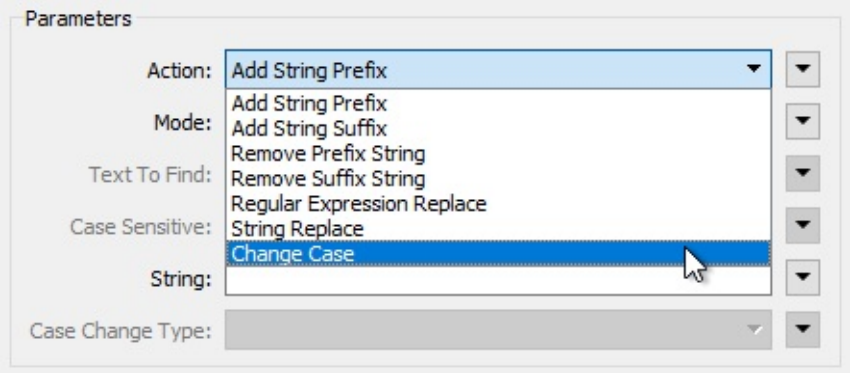
BulkAttributeRenamer

BulkAttributeRenamer执行重命名属性的核心功能。但是，这个转换器不是手动指定每个属性，而是让用户选择多个属性 - 或者全部：

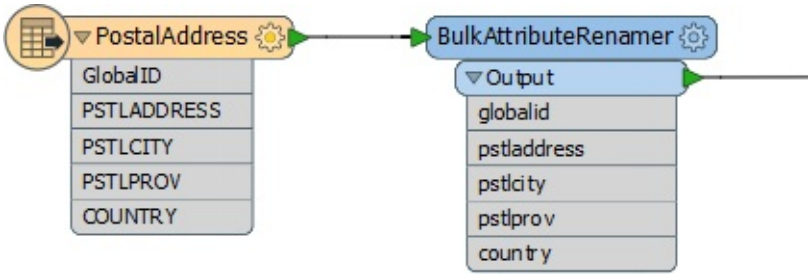


当选择多个属性时，操作必须 - 当然 - 对它们执行相同的重命名操作。这些操作是：

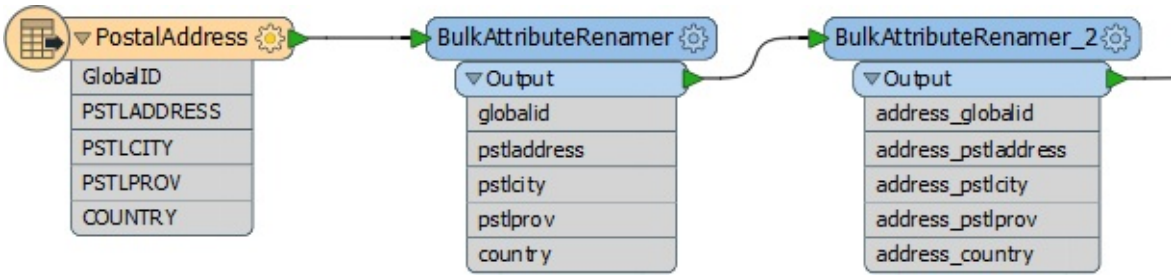
- 添加字符串前缀
- 添加字符串后缀
- 删除前缀字符串
- 删除后缀字符串
- 正则表达式替换
- 字符串替换
- 改变大小写



转换器的功能还在于能够一次操作多个属性，而无需单独选择它们。例如，在这里，传入的属性都被重命名为小写名称，以匹配不支持大写的写模块模式：



可以使用多个转换器来创建累积效果。例如，用户已转换为小写，然后使用第二个转换器添加前缀：



删除属性

删除属性可能被视为FME中不太重要的任务。这是因为 - 对于手动属性模式 - 只将写模块中定义的属性写入输出; 只需忽略不需要的额外属性。

但是，删除属性确实带来了有用的好处：

- 删除不需要的属性可以整理工作空间并使其更易于理解
- 工作空间是对象和模式的复杂网络。删除属性可简化此网络，并使Workbench界面更具响应性
- 所有数据处理都会产生时间和内存成本。删除属性意味着处理的数据更少，因此FME引擎执行速度更快

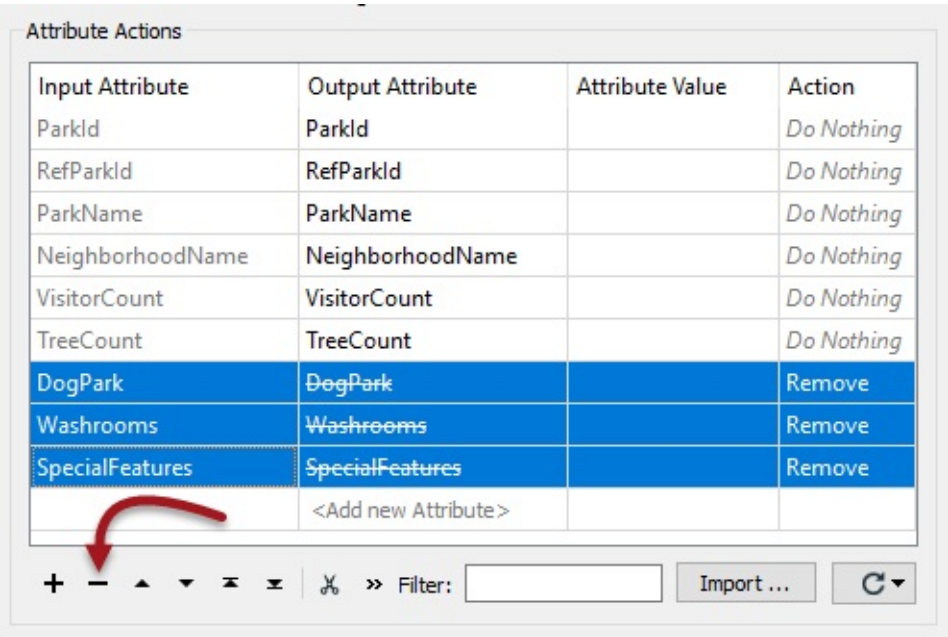
技巧
读模块要素类型具有隐藏模式中属性的参数。这有助于整理工作空间，但无助于提高转换性能。但是，某些格式（主要是数据库）也有“要读取的属性”的设置，使用它将有助于提高性能。

可以删除属性的转换器是：

- AttributeKeeper
- AttributeManager
- AttributeRemover
- BulkAttributeRemover

删除属性

AttributeManager和AttributeRemover具有相同的技术; 选择要删除的属性：



通过选择属性并单击 - 按钮，可以在AttributeManager中删除属性。或者，您可以将操作字段从“ 不执行任何操作”更改为“删除”。

请注意，上面已删除了三个属性。输出属性（选中时）显示已删除的名称，表示它不再存在。

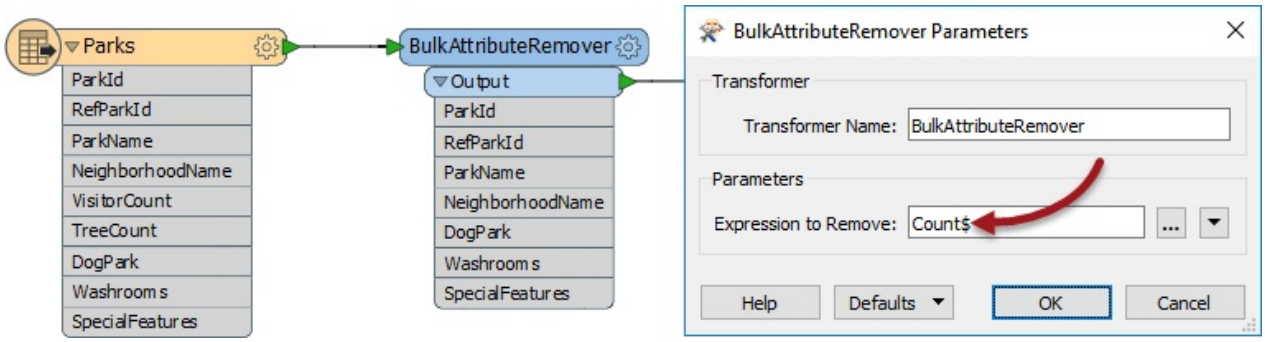
保持属性

AttributeKeeper转换器执行相同的功能，但从相反的方向接近它。它允许用户指定不删除哪些属性; 换句话说，这个转换器允许用户指定要保留的转换器。

因此，应该在要删除一个或两个属性的情况下使用AttributeManager，但保留大部分属性。当要删除大多数属性时，应使用AttributeKeeper，并且只保留其中的一个或两个。

批量属性删除

BulkAttributeRemover - 与BulkAttributeRenamer一样 - 允许用户对多个属性执行处理。在这种情况下，用户不是能够选择所有属性，而是输入字符串匹配表达式来定义要删除的属性：



在这里，用户正在删除名称以“Count”结尾的所有属性。

练习：地址开放数据项目

练习**2**	噪音控制法项目（地址）
数据	地址（文件地理数据库）
总体目标	将文件地理数据库转换为Microsoft Excel并映射模式
演示	模式映射的属性管理
启动工作空间	无
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformers-Ex2-Complete.fmw

市议员投票决定修改噪声控制法，生活在受影响地区的居民必须了解这些变化。

您的经理建议您承担查找所有受影响地址的任务。有一个紧迫的截止日期，至少有三位市议员正在观看你的工作。压力很大，由你来决定！

本练习是该项目的第一部分。您知道城市的地址数据库存储在Esri地理数据库中，该模式的模式与本地政府信息模型PostalAddress表相匹配。

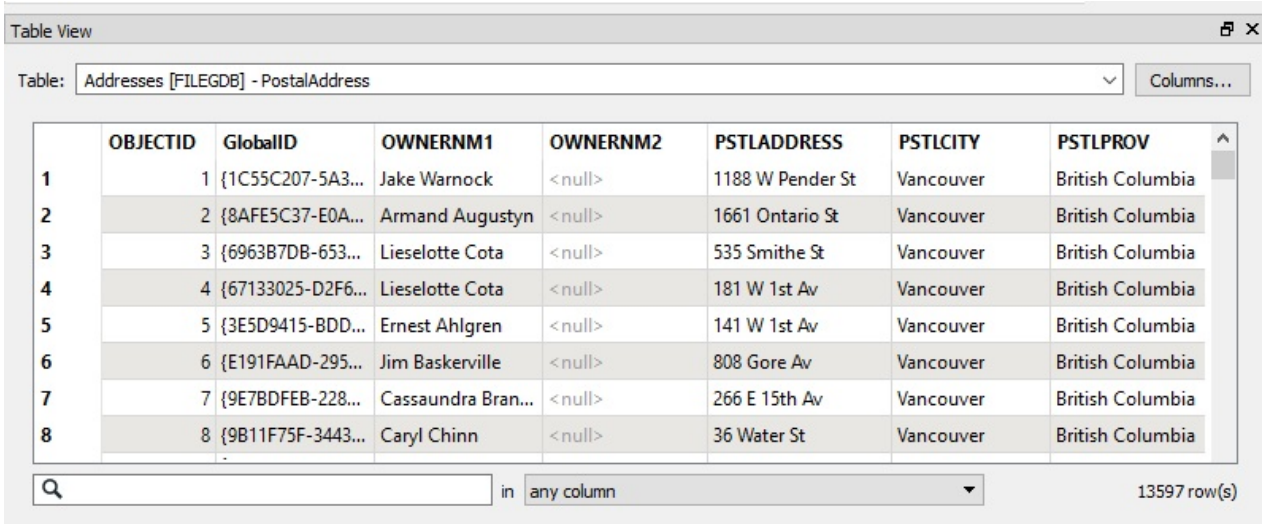
但是，您被告知用于执行自动批量邮件的软件需要使用完全不同的模式存储在Excel电子表格中的地址。

因此，您的第一个任务是创建一个工作空间，将地址从Geodatabase转换为Excel，同时映射模式。

1）检查数据
像往常一样，第一项任务是熟悉数据。为此，请在FME Data Inspector中打开以下数据集：

读模块格式	**Esri**地理数据库（**File Geodb API**）
读模块数据集	C:\FMEData2018\Data\Addresses\Addresses.gdb

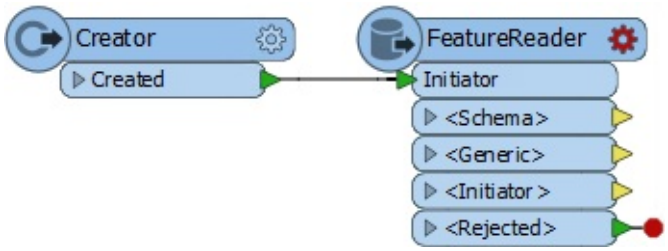
要转换的表称为“PostalAddress”。这里重要的不是数据在图形显示中的外观，而是表视图窗口中存在的更多属性。



2）添加**Creator / FeatureReader**
现在您已熟悉源数据，启动FME Workbench并从空工作空间开始。

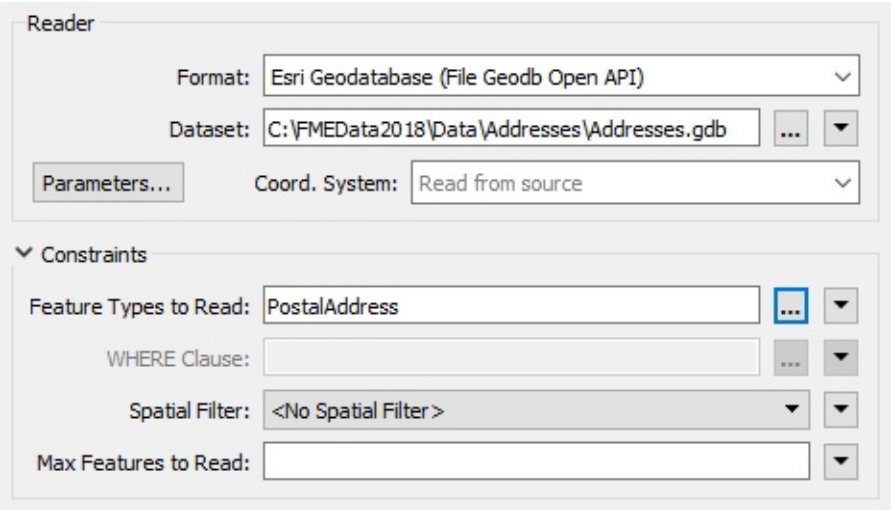
我们可以选择使用读模块或FeatureReader转换器来读取源数据。FeatureReader将允许我们构建空间过滤器，因为我们相信这个项目可能需要一些过滤 - 我们将使用FeatureReader转换器和Creator来创建一个触发它的要素。

所以放置一个Creator转换器并将其连接到FeatureReader：



检查FeatureReader参数。如下设置参数：

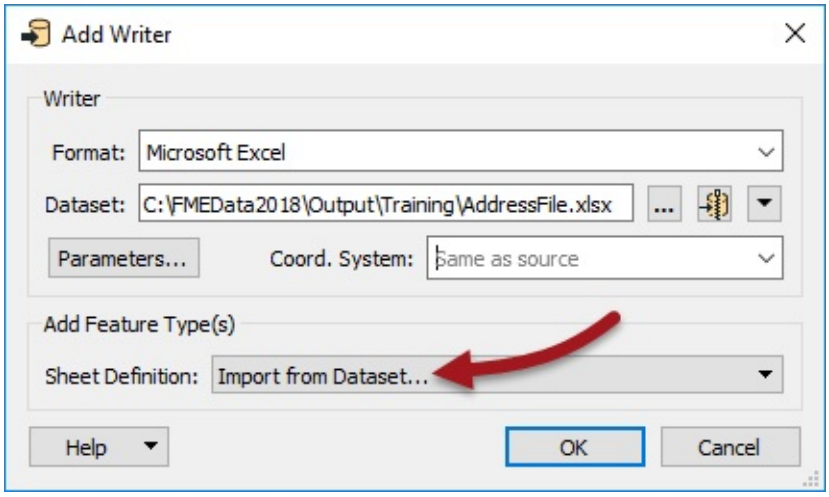
读模块格式	**Esri**地理数据库（**File Geodb API**）
读模块数据集	C:\FMEData2018\Data\Addresses\Addresses.gdb
要读取的要素类型	PostalAddress



确保已启用“使用要素缓存运行”并运行工作空间。检查FeatureReader：PostalAddress输出缓存以确认正确读取所有地址。

3）添加写模块
现在让我们添加一个写模块来写入输出数据。目前似乎没有任何好处或需要使用FeatureWriter，因此从菜单栏中选择Writers> Add Writer并使用以下内容：

写模块格式	**Microsoft Excel**
写模块数据集	C:\FMEData2018\Output\Training\AddressFile.xlsx
工作表定义	从数据集导入



设置“从数据集导入”将允许我们导入Excel电子表格以用作指南。单击“确定”以添加写模块。

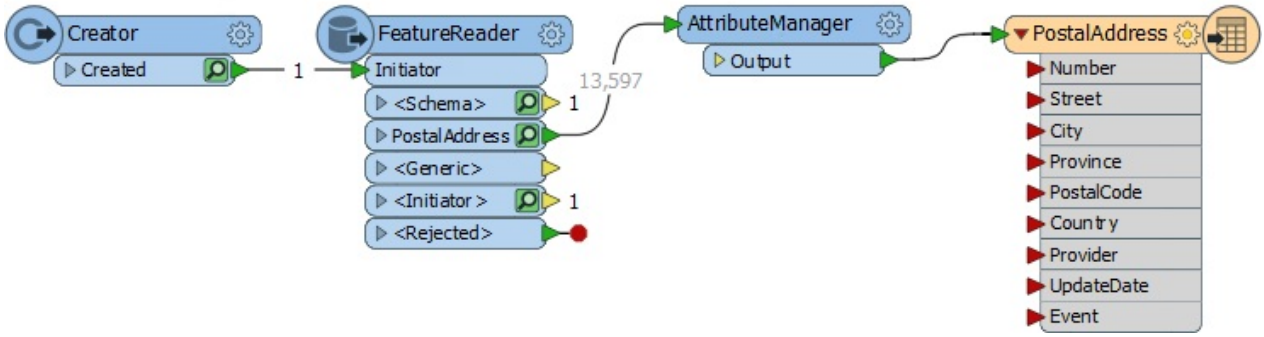
4）导入要素类型
此时，系统会提示您选择要从中导入模式定义的数据集。应使用与写模块相同的值设置这两个字段。设置数据集参数如下：

读模块数据集	**C:\FMEData2018\Resources\DesktopBasic\AddressSchema.xlsx**

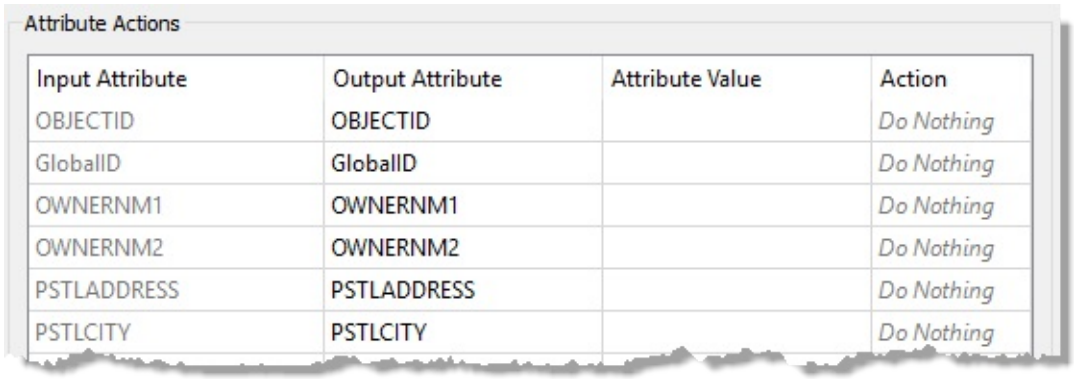
此文件是我们的指南/模板。单击“确定”接受值。将创建新要素类型以匹配所选的Excel模式。

5）添加**AttributeManager**
现在我们可以开始将模式从读模块（FeatureReader）映射到写模块。正如你已经注意到的那样，两者目前并不匹配。

因此，将两者之间的AttributeManager连接起来。



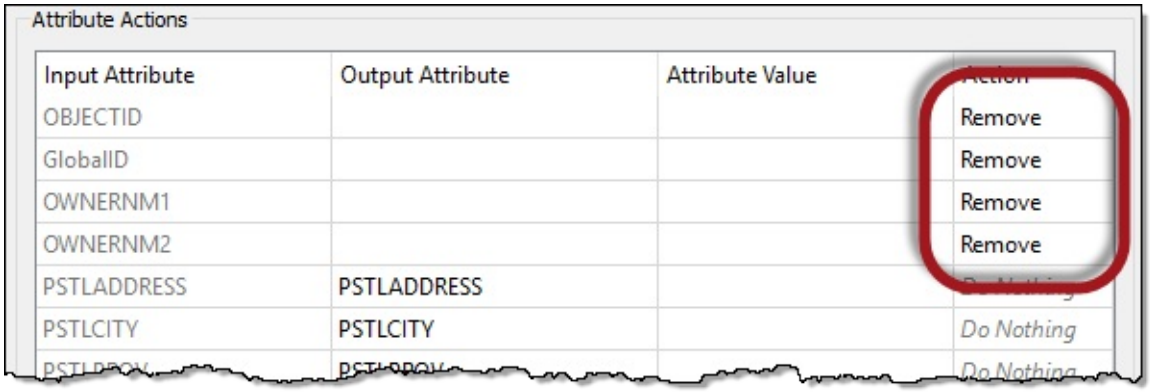
它的参数如下所示：



首先，让我们通过删除一些不需要的属性来清理读模块模式。

单击以下属性，然后按 - 按钮将其删除：

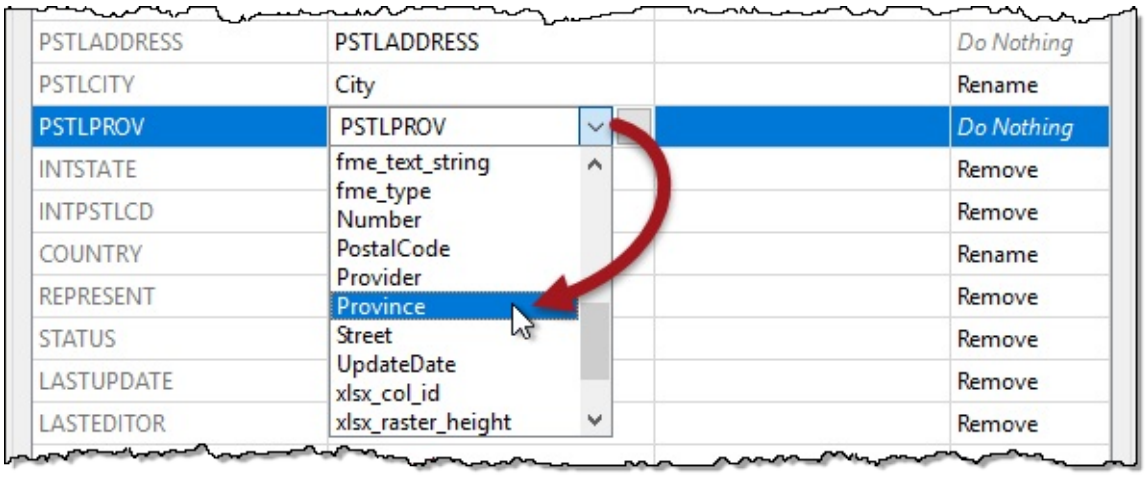
- OBJECTID
- GlobalID
- OWNERNM1
- OWNERNM2
- INTSTATE
- INTPSTLCD
- REPRESENT
- STATUS
- LASTUPDATE
- LASTEDITOR



6）重命名属性
可以将多个源属性按原样写入输出，但需要先重命名。

在AttributeManager中重命名以下内容：

- PSTLCITY到City
- PSTLPROV到Province
- POSTALCODE到PostalCode
- COUNTRY到Country



如果AttributeManager连接到写模块要素类型，那么您应该能够从下拉列表中选择“输出属性”字段，而不是键入它。

7）创建属性（**Provider**）
输出上的两个属性（Provider和UpdateDate）是新的，不能从源数据中复制。必须创建它们。

在AttributeManager中创建新属性“Provider”。由于该属性存在于输出模式中，因此您可以再次从下拉列表中选择该属性。

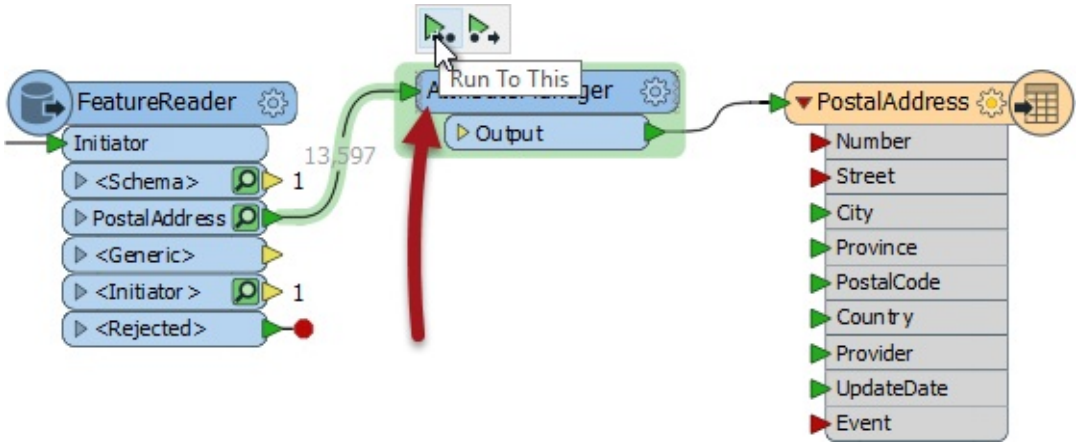
设置固定值，例如您自己的组织名称，“Safe Software”或“Interopolis城市”。

8）创建属性（**UpdateDate**）
现在创建新属性“UpdateDate”。不要对值进行硬编码，而是单击“属性值”字段中的下拉箭头，然后选择“打开文本编辑器”。

在文本编辑器中找到名为DateTimeNow的日期/时间函数，然后双击将其放在编辑器中。默认情况下，它会在ISO语法中创建一个日期时间，这对我们来说很好，所以单击“确定”接受此操作。

再次单击“确定”以关闭“AttributeManager”对话框。

9）运行转换器
通过单击AttributeManager转换器并选择Run to This运行工作空间：



这将使用以前的缓存自行运行转换器，但不写入任何输出（我们还不需要）。

检查AttributeManager：输出缓存以确认该过程按预期工作：

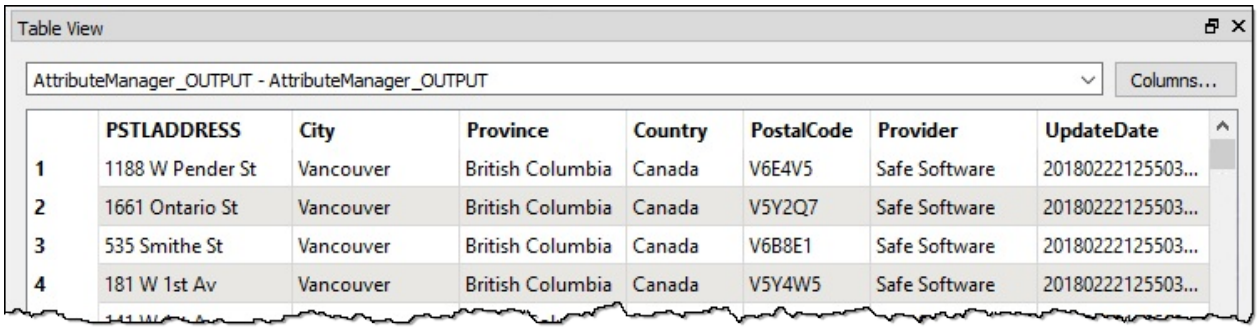


Table View

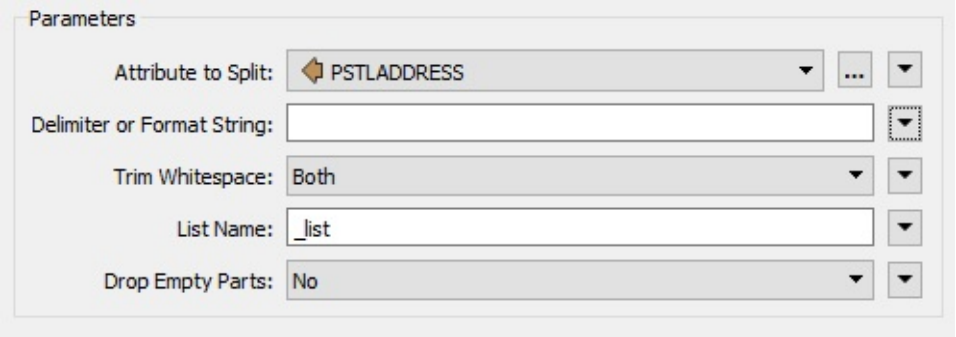
AttributeManager_OUTPUT - AttributeManager_OUTPUT　Columns...

	PSTLADDRESS	City	Province	Country	PostalCode	Provider	UpdateDate
1	1188 W Pender St	Vancouver	British Columbia	Canada	V6E4V5	Safe Software	20180222125503...
2	1661 Ontario St	Vancouver	British Columbia	Canada	V5Y2Q7	Safe Software	20180222125503...
3	535 Smithe St	Vancouver	British Columbia	Canada	V6B8E1	Safe Software	20180222125503...
4	181 W 1st Av	Vancouver	British Columbia	Canada	V5Y4W5	Safe Software	20180222125503...

10）添加**AttributeSplitter**
查看输出模式，Number和Street有两个字段（例如“3305”和“W 10th Av”）。但是，源模式将该信息压缩到一个字段中，其中<space>字符分隔字段（“3305 W 10th Av”）。因此，我们必须将源属性拆分为与写模块模式匹配。

插入AttributeSplitter转换器。在AttributeManager 之前插入它- 然后如果要对split属性执行任何操作，我们可以使用相同的AttributeManager转换器。

查看AttributeSplitter参数。将PSTLADDRESS设置为要拆分的属性，并在Delimiter参数中输入空格字符。请注意，列表名称在List Name参数中设置（我们将很快使用该列表）：



单击“确定”关闭对话框。如果现在运行工作空间并检查缓存，您将在“要素信息”窗口中看到该地址作为列表属性：

```
_list{0}（encoded：utf-8）3305
_list{1}（encoded：utf-8）W
_list{2}（encoded：utf-8）10
_list{3}（encoded：utf-8）Av
```

请记住，列表属性是可以在单个名称（_list）下存储多个值的属性。

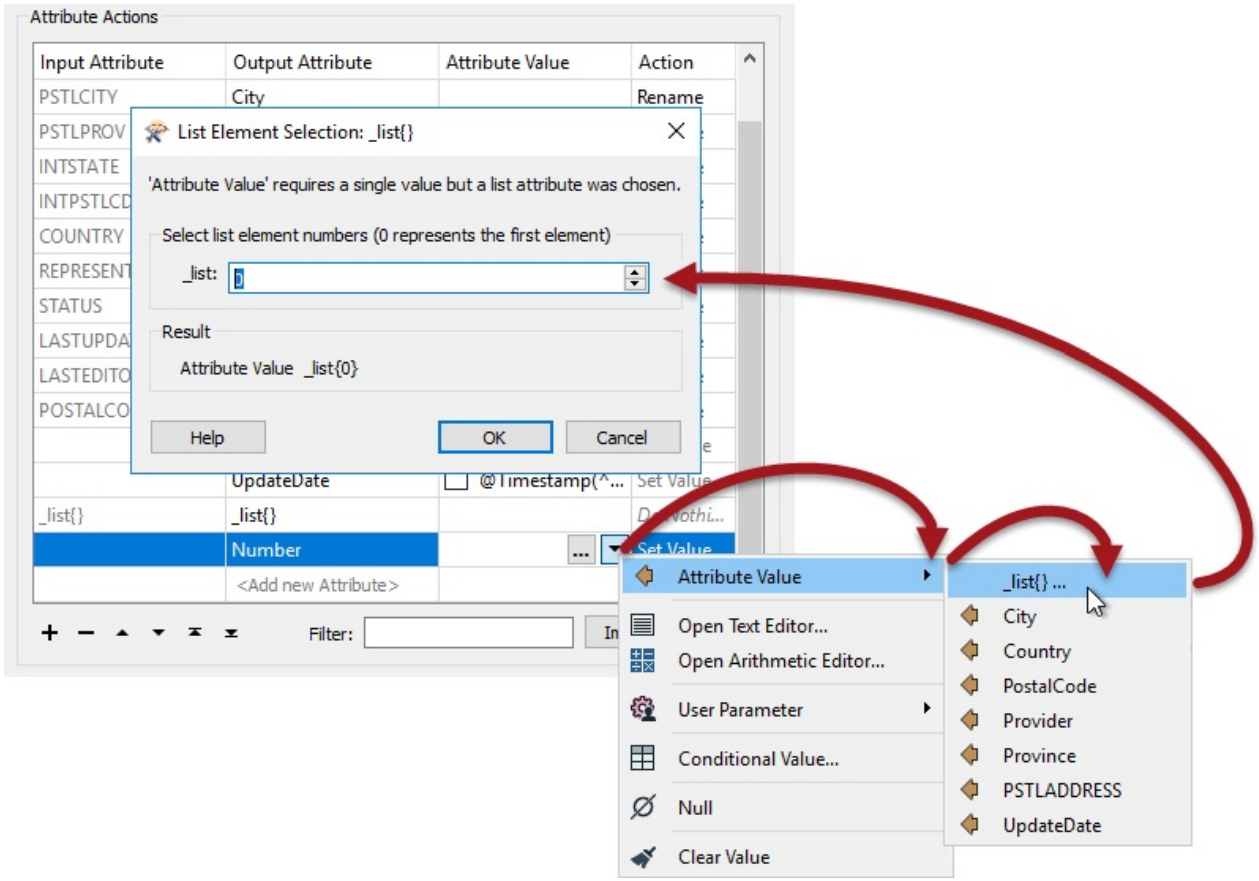
11）重命名属性
现在让我们处理输出中的Number字段。返回AttributeManager参数。

请注意，现在有一个名为_list{}的list属性的条目。但是，这只是“一般的”列表属性- 它没有显示列表中的每个元素（值）。

我们需要做的是创建一个新属性并将我们想要的列表元素复制到其中。因此，在“输出属性”字段中，通过从下拉列表中选择一个名为“数字”的新属性。

对于“属性值”字段，单击下拉箭头并选择“属性值”>“_list {}”。

现在将提示您选择列表中的元素。确保将其设置为零（0）并单击“确定”。



单击“应用/确定”以确认更改。运行工作空间并检查AttributeManager:Output缓存以确保正在复制该编号。

12）构建属性
最后一步是重新创建Street属性，而不用地址编号作为前缀。

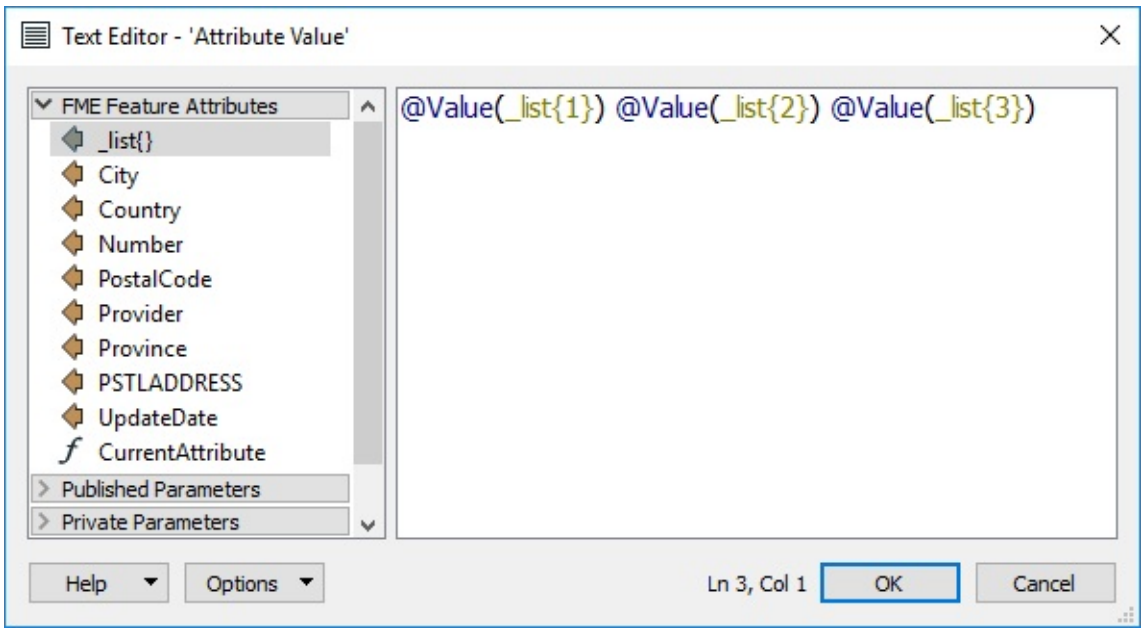
再次查看AttributeManager参数。这次在输出属性字段中，通过从列表中选择它来创建一个名为Street的新属性。

与Number字段不同，我们希望通过连接多个列表元素来创建此值。因此，单击“属性值”字段中的下拉箭头，然后选择“打开文本编辑器”。

在FME Feature Attributes菜单中找到_list {}并执行以下步骤：

- 双击_list{}，然后在提示时选择元素1.单击“确定”
- 按空格键输入<space>字符
- 双击_list{}，然后在提示时选择元素2.单击“确定”。
- 按空格键输入<space>字符
- 双击_list{}，然后在提示时选择元素3.单击“确定”。

该对话框现在看起来像这样：



通过这种方式，我们将街道名称的所有部分连接在一起，例如：

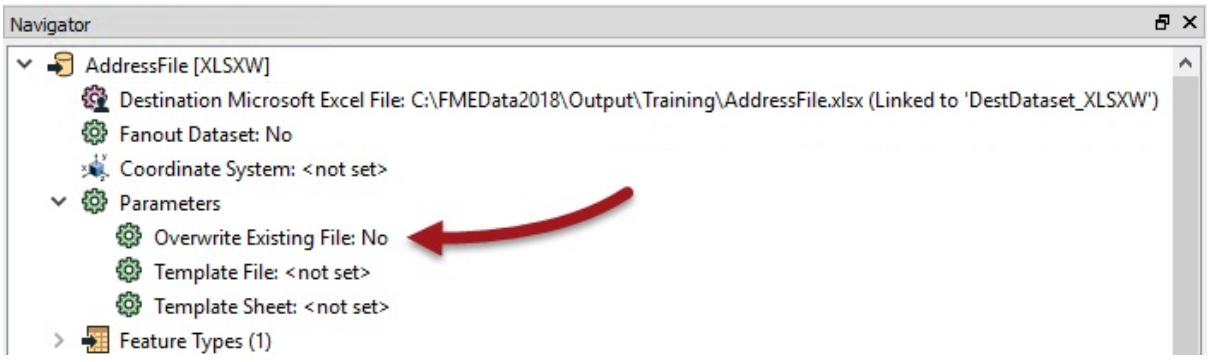
```
“W”+“17th”+“St”变成“W 17th St”
```

我们假设没有街道名称有三个以上的部分，但这对我们的例子来说是合理的。

13）运行工作空间
保存工作空间。关闭要素缓存，然后运行工作空间。

警告

Excel的写模块有一个参数名为覆写现有文件，默认设置为**No**。 这可能是将参数更改为Yes的好时机，因为我们已经多次运行工作空间！



打开包含文件夹以检查输出是否已写入。检查FME Data Inspector中的数据。输出（在“表视图”窗口中）应如下所示：

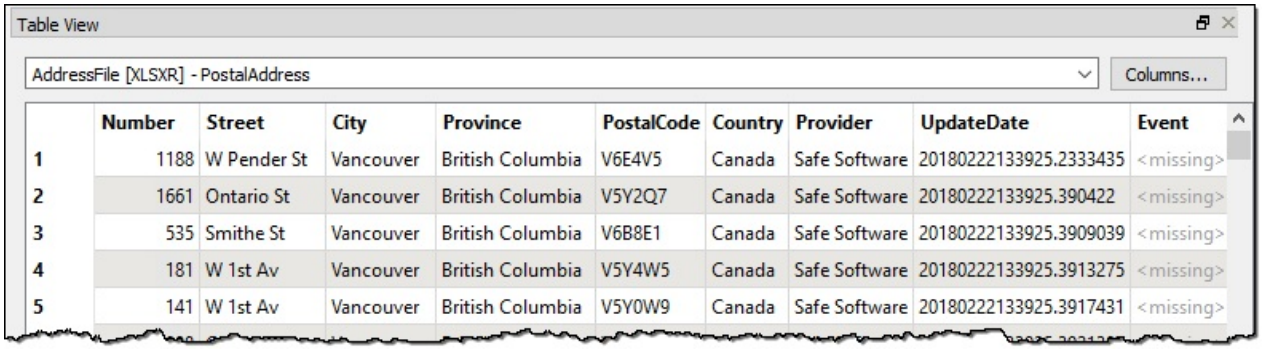


Table View

AddressFile [XLSXR] - PostalAddress

Columns...

	Number	Street	City	Province	PostalCode	Country	Provider	UpdateDate	Event
1	1188	W Pender St	Vancouver	British Columbia	V6E4V5	Canada	Safe Software	20180222133925.2333435	<missing>
2	1661	Ontario St	Vancouver	British Columbia	V5Y2Q7	Canada	Safe Software	20180222133925.390422	<missing>
3	535	Smithe St	Vancouver	British Columbia	V6B8E1	Canada	Safe Software	20180222133925.3909039	<missing>
4	181	W 1st Av	Vancouver	British Columbia	V5Y4W5	Canada	Safe Software	20180222133925.3913275	<missing>
5	141	W 1st Av	Vancouver	British Columbia	V5Y0W9	Canada	Safe Software	20180222133925.3917431	<missing>

技巧

此时，“事件(Event)”字段仍为空，但将在后续练习中完成。

恭喜

通过完成本练习，您已学会如何：

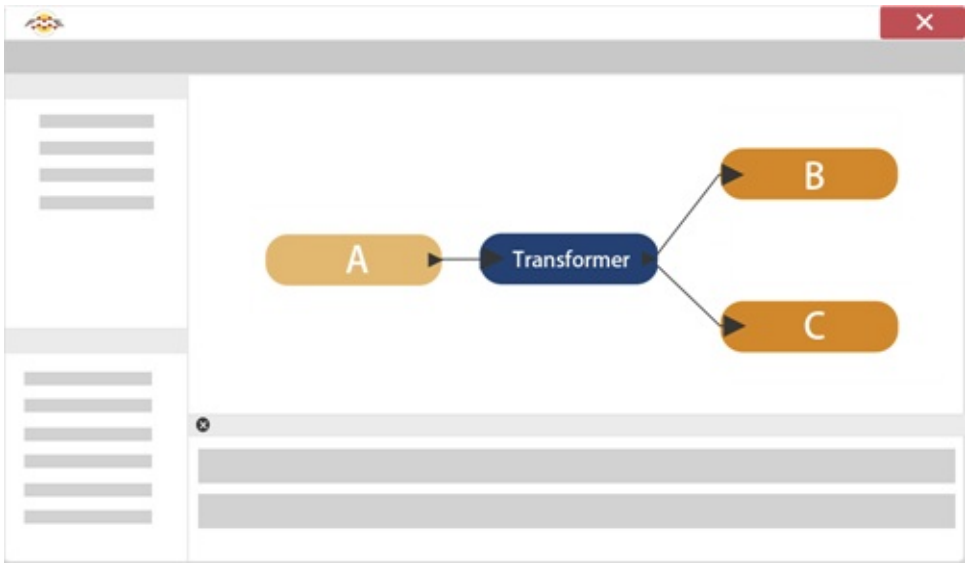
- 使用AttributeManager转换器创建，删除和重命名属性
- 使用AttributeSplitter将属性拆分为列表属性
- 处理AttributeManager中的列表属性
- 在AttributeManager文本编辑器中使用日期/时间要素

条件过滤

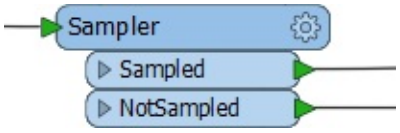
过滤转换器不会转换数据内容，但调查表明它们是最常用的转换器类型！

什么是过滤？

过滤是在数据流经工作空间时细分数据的技术。在这种情况下，转换器有多个输出连接，每个输出都承载要以不同方式处理的数据。这里（例如）传入流A被过滤成两个新流B和C：



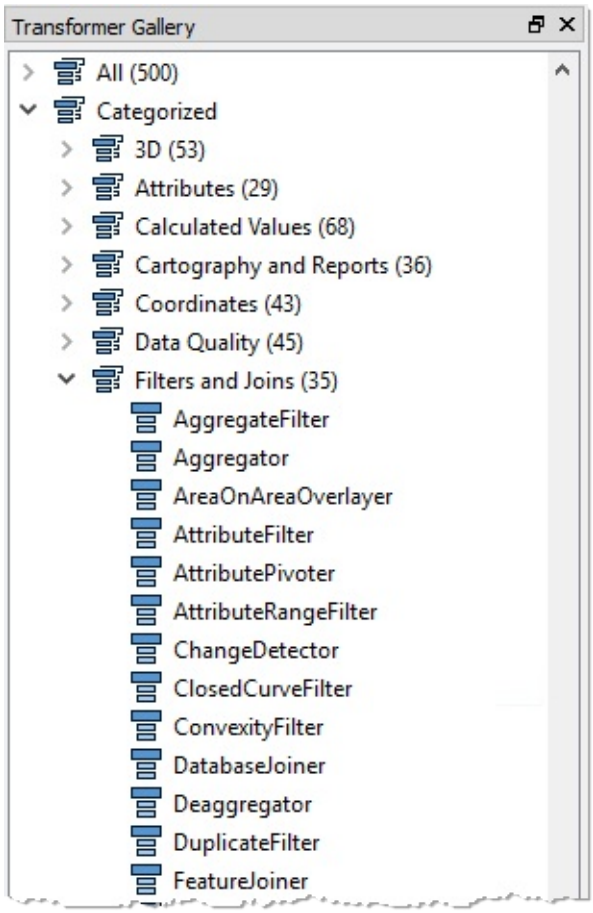
过滤转换器可以是具有多个输出端口的任何转换器，例如GeometryFilter或Sampler转换器，后者创建数据样本选择并通过Sampled和NotSampled输出端口将其分离：



但是，这些大多是内置的固定测试。条件过滤是决定哪些要素输出到哪个连接由某种形式的用户定义的测试或条件决定的。Tester转换器就是最明显的例子。它执行测试并具有不同的输出端口，用于通过和未通过测试的要素。

过滤转换器

“过滤和连接”类别中的许多转换器执行这些测试并根据结果重定向数据：



尽管Tester变换器是此类别中使用最多的变换器，但还有许多其他变换器，例如TestFilter, GeometryFilter, AttributeFilter, SpatialFilter, and Sampler。

Vector小姐说...

是时候快速提问了。过滤和连接类别中有多少个过滤转换器出现在前30个最有价值转换器列表中？

1. 一 (1)
2. 四 (4)
3. 七 (7)
4. 十 (10)

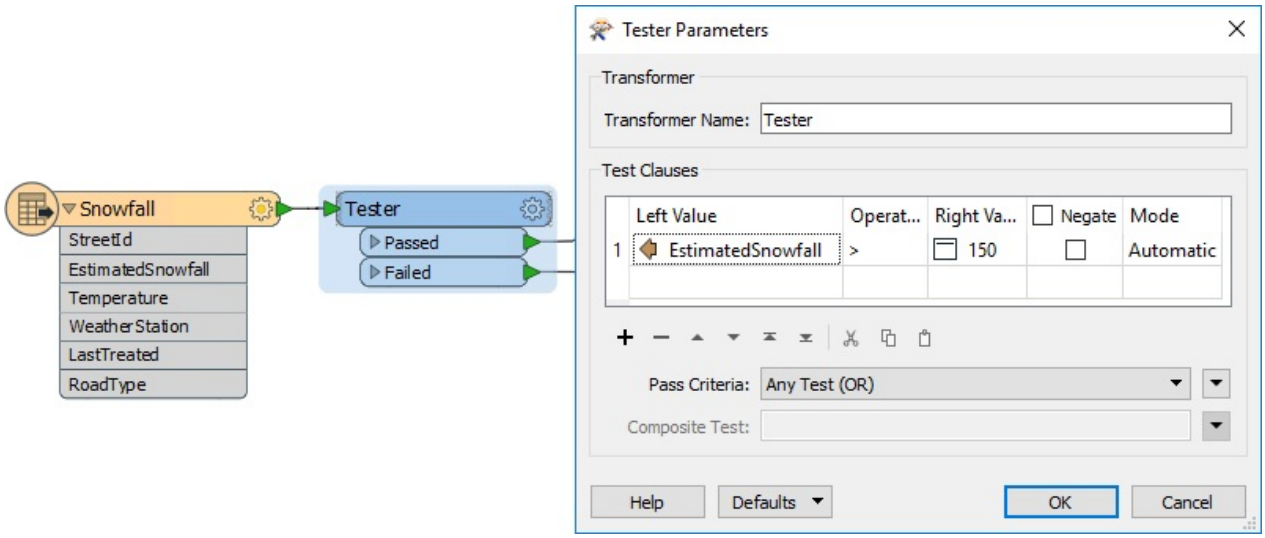
Tester和TestFilter

Tester和TestFilter是条件过滤的两个关键转换器。它们测试属性值的值。

Tester

Tester转换器（前30位中的编号1）通常用于产生“是/否”结果的单次测试。

例如，在这里我们希望根据Snowfall属性的值是否大于150毫米（约6英寸）来决定是否将扫雪机发送到特定道路：



如果降雪量大于150，道路要素将通过测试，并将发送扫雪机。

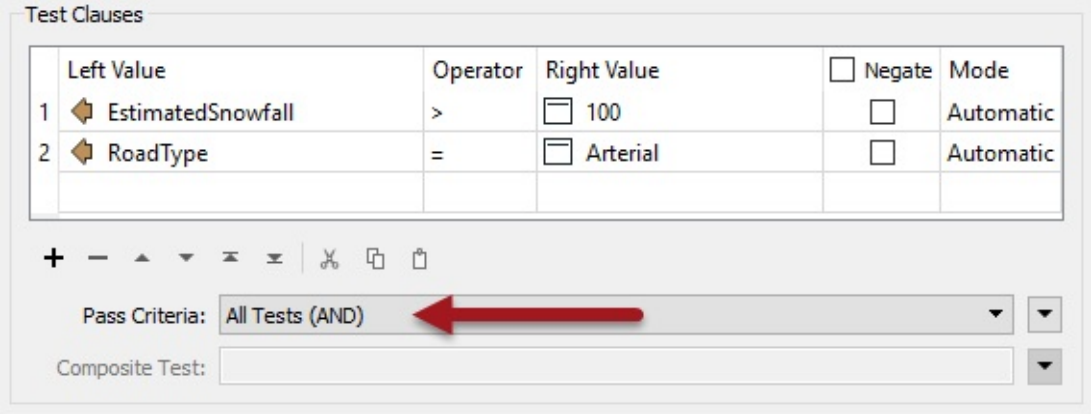
多个子句

Tester中的每个子句都是一个允许通过/失败结果的单独测试。例如，以下每个条件可能是单独的测试：

- 有超过100毫米（4英寸）的降雪？
- 这是一条主要道路吗？
- 温度是否低于零摄氏度？
- 沙子上次使用超过24小时了吗？

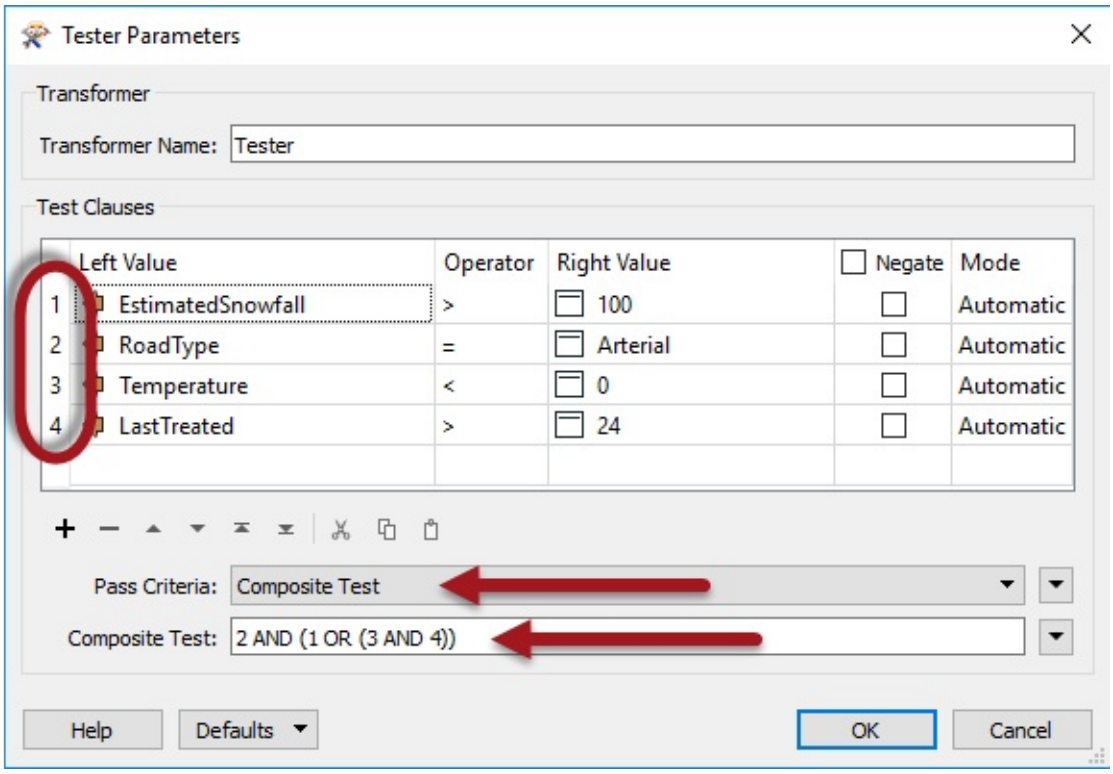
但是，测试程序允许组合多个测试，其中用户可以使用AND和OR语句组合任意数量的子句。因此，我可能会问：

- 这是一条动脉（主要）道路并且有超过100毫米的雪吗？



Tester还允许使用所谓的复合测试混合AND和OR语句。例如：

- 这是一条主干道AND(有超过100毫米的雪OR(温度小于零AND是24小时前(用沙子)处理的道路))？



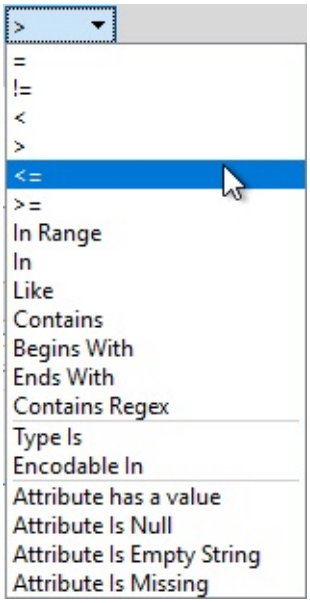
请注意，复合测试中使用的数字是Test Clauses窗口左侧列出的子句编号。

但是 - 无论测试变得多么复杂 - 它仍然会导致单个是/否（二进制）结果; 要素将通过或未通过这组测试。

值得记住的是，我们不仅限于简单的平等测试（A = B）; 在上面，还有“大于”和“小于”的测试。那是因为有许多不同的运算符可用于测试子句。

运算符

Tester转换器（或许多其他使用Tester对话框的位置）中可用的运算符列表如下所示：



除了通常的运算符之外，还有一些基于SQL where子句的运算符。这些包括:

- In
- Like
- Contains
- Begins With
- Ends With
- Contains Regex

...还有其他测试可以检查属性和值的存在：

- 属性有一个值
- 属性为空
- 属性为空字符串
- 属性缺失

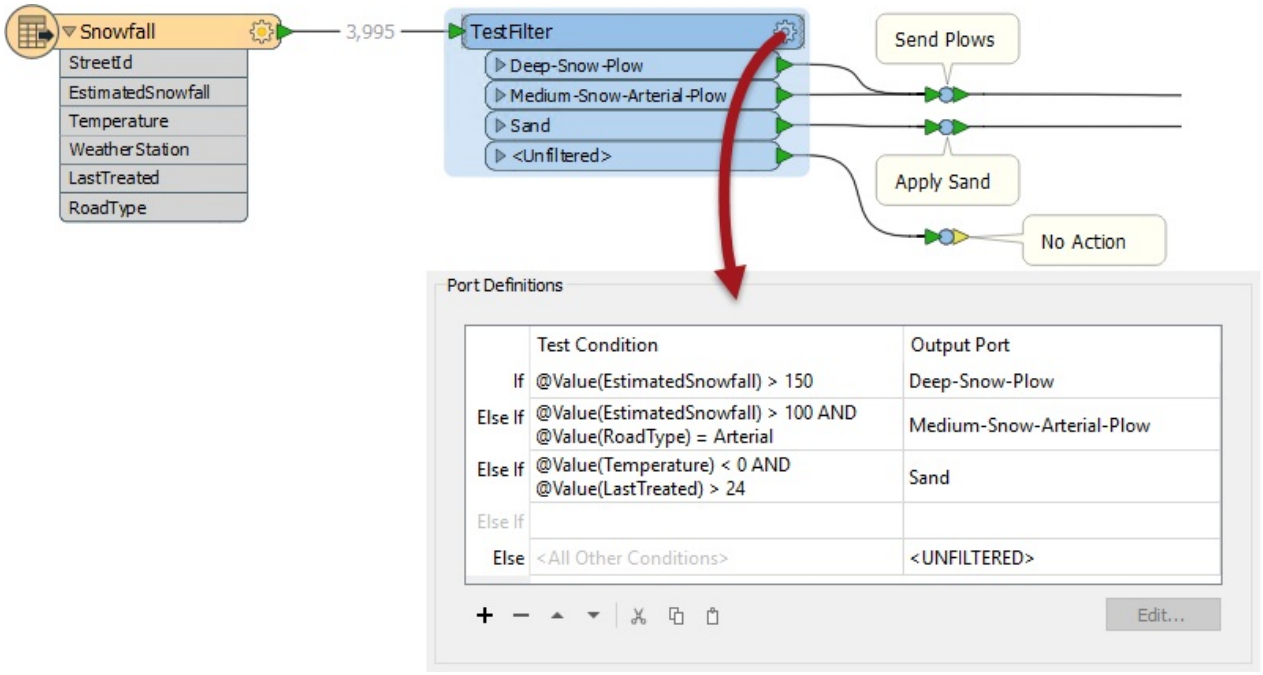
技巧
“属性有一个值”与其他三个测试相反; 即此属性不是空，并且它不是空字符串，并且它不会丢失。顺便提一下，“缺失”表示该属性在被测试的要素上根本不存在。

技巧
“包含正则表达式”（Contains Regex）表示只需要匹配字符串的一部分。例如... 即整个字符串不需要匹配。

TestFilter

TestFilter（前30名中的＃8）允许测试许多条件，每个条件都可以有许多测试条款。每个条件都有一个输出端口，还有一个额外的输出端口，用于在所有测试条件下失败的要素。

TestFilter与编程或脚本语言中的CASE或SWITCH命令非常相似。在Workbench中它看起来像这样：



请注意，有多个条件，每个条件都有一个输出端口。每个条件/端口组合相当于一个Tester转换器; 因此，TestFilter是将多个Tester转换器合而为一的好方法。

TestFilter输出端口可以给出自定义名称，而不是简单的PASSED / FAILED，这是该转换器相比Tester的另一个优势。

TestFilter具有Tester可用的全套运算符，例如相等，大于，小于等等。每个条件依次进行测试。

通过的要素通过匹配的输出端口输出。失败的要素将发送到列表中的下一个条件。因此，以正确的顺序获得条件非常重要。

Intuitive修女说.....

TestFilter非常适合通过一系列级联条件过滤要素，例如，这里有一组测试可以再次确定是否派出扫雪机：

- 降雪量超过150毫米？
- 降雪量超过100毫米并且这是一条干道吗？
- 温度是否低于零摄氏度并且是否超过24小时前最后施加的沙子？

这是一套级联测试，因为如果有超过150毫米的雪，扫雪机就会被送出去; 您不需要测试任何其他条件。因此测试顺序非常重要。如果每次测试都失败，那么扫雪机不会派出。

另请注意，您可以包含复合测试（包含ANDs或OR的测试）。

如果使用上面的降雪示例，您使用了三个Tester转换器，则可以节省工作区画布上的空间，通过仅使用一个TestFilter替换该设置。

厨师**Bimm**说.....

因为TestFilter可以执行单个测试（以及多个测试），所以可以完全使用它而不是Tester转换器。

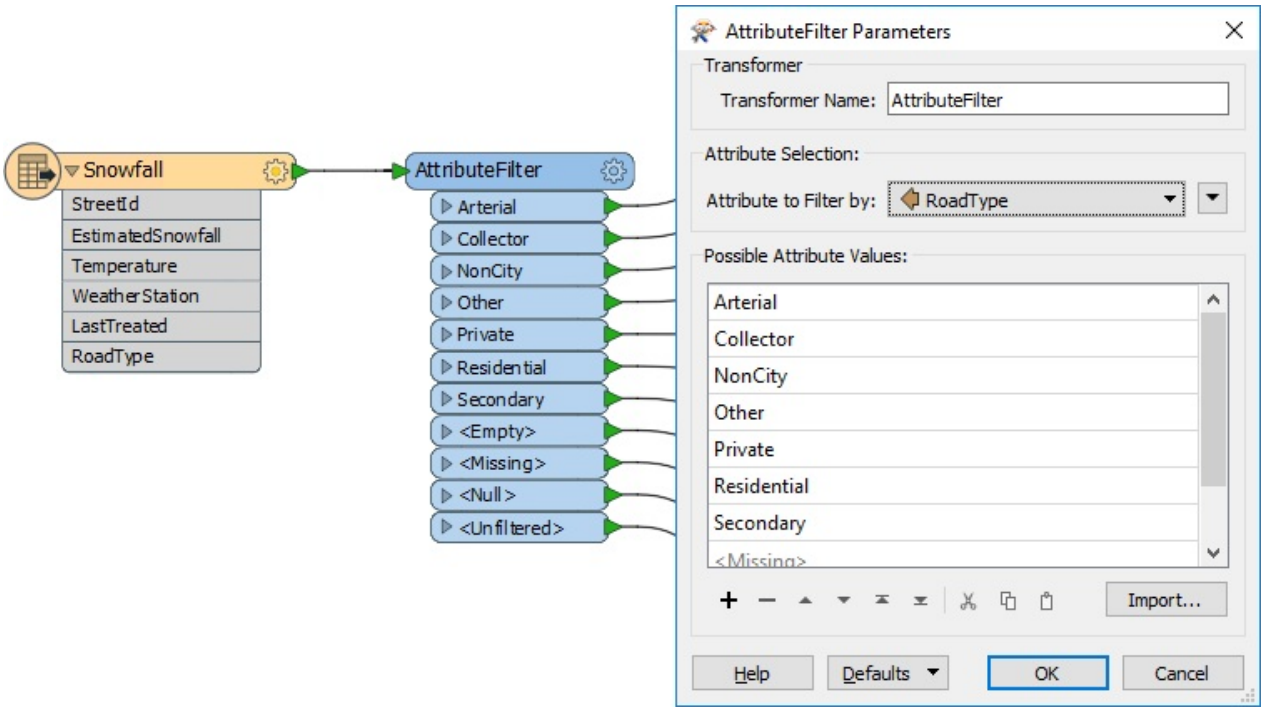
其他主要过滤转换器

Tester和TestFilter不是唯一有用的过滤转换器。

AttributeFilter

AttributeFilter转换器（前30个中的＃13）通过所选属性中的值来定向要素。它不是二进制测试（是/否），而是一种为单个属性分隔多个值的方法，例如：

- 那条道路是主干道，联络道路，非城市的，其他，私人的，住宅区道路还是二级公路 ？



如果您使用七个Tester转换器来分隔这些数据，则可以节省工作空间画布上的空间，并仅使用一个TestFilter替换该设置。

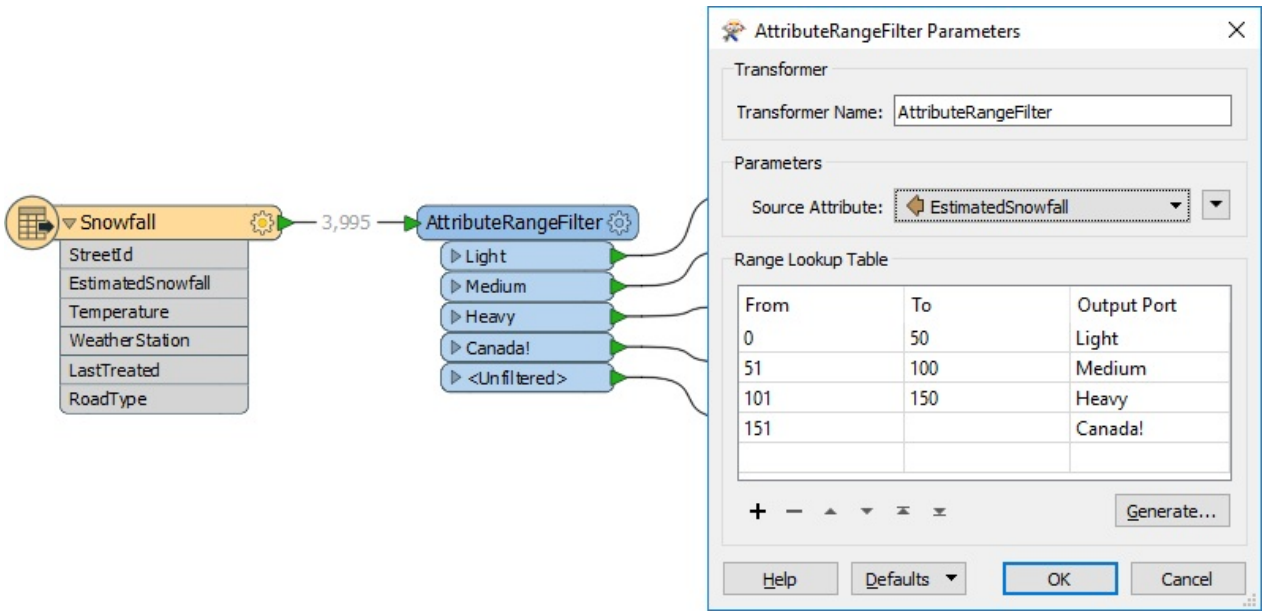
Intuitive修女说.....
在几乎所有使用多个Tester转换器的场景中，可以使用不同的过滤转换器来实现相同的结果，但在画布上使用的空间要少得多。

AttributeFilter也可以使用处理数值; 但是，它唯一的“运算符”是求等价(=)，所以你会很少用它来进行算术测试。在这种情况下，更好的解决方案是AttributeRangeFilter。

AttributeRangeFilter

AttributeRangeFilter执行与AttributeFilter相同的操作，除了它可以处理一系列数值而不是简单的一对一匹配。

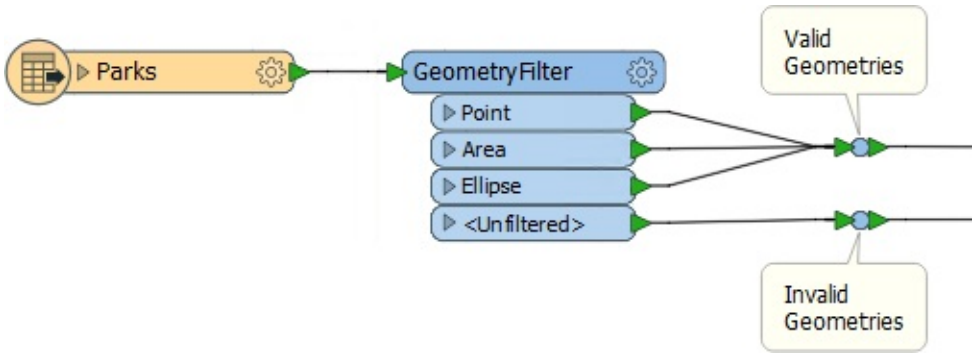
例如，我们可能希望根据一系列降雪值来分离数据，如下所示：



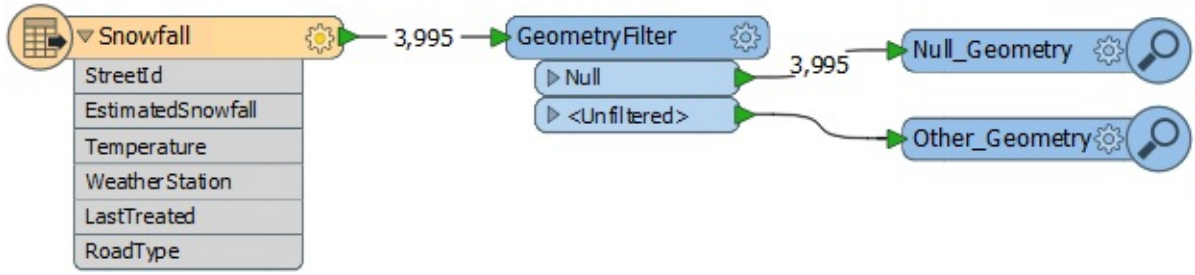
请注意，AttributeRangeFilter参数对话框有一个Generate按钮，可以从一组用户定义的区段自动生成范围。

GeometryFilter

GeometryFilter（前30名中的＃22）基于几何类型定向要素; 例如，点，线，面，椭圆：



它甚至可以根据空几何对象（表格记录）过滤数据：



GeometryFilter对以下内容非常有用：

- 过滤掉不需要的几何类型; 例如，在使用AreaBuilder转换器之前删除非线性要素
- 根据允许类型列表验证几何对象; 例如，数据集被约束到点或面要素（上图）
- 划分几何类型以写入单独的目标要素类; 例如，写入到有几何限制的格式（如Esri Shapefile）时

来自**Interopolis**的**Confused**说.....

亲爱的Interop阿姨，
如果Tester，TestFilter和AttributeFilter都根据属性条件过滤要素，那么有什么区别？我什么时候会用到它们？

Interop阿姨说.....

亲爱的Confused，
最好的解决方案是在Safe Software博客上查看这两篇文章：
- FME中的条件处理
- FME过滤转换器的简单指南
我还做了一个有用的表格:

	单一测试		多个测试		测试类型		运算符	属性
	单一子句	多个子句	单一子句	多个子句	字符串	数字		
Tester	Y	Y	–	–	Y	Y	16	多个
TestFilter	Y	Y	Y	Y	Y	Y	16	多个
AttributeFilter	Y	Y	–	–	Y	–	1	1
AttributeRangeFilter	Y	Y	–	–	–	Y	6	1

练习：噪音控制法律项目

练习**3**	噪声控制法项目（空间过滤）
数据	地址（文件地理数据库） 分区（MapInfo TAB） 道路（AutoCAD DWG）
总体目标	查找主干道50米范围内的所有住宅地址
演示	条件过滤的方法
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformers-Ex3-Begin.fmw
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformers-Ex3-Complete.fmw

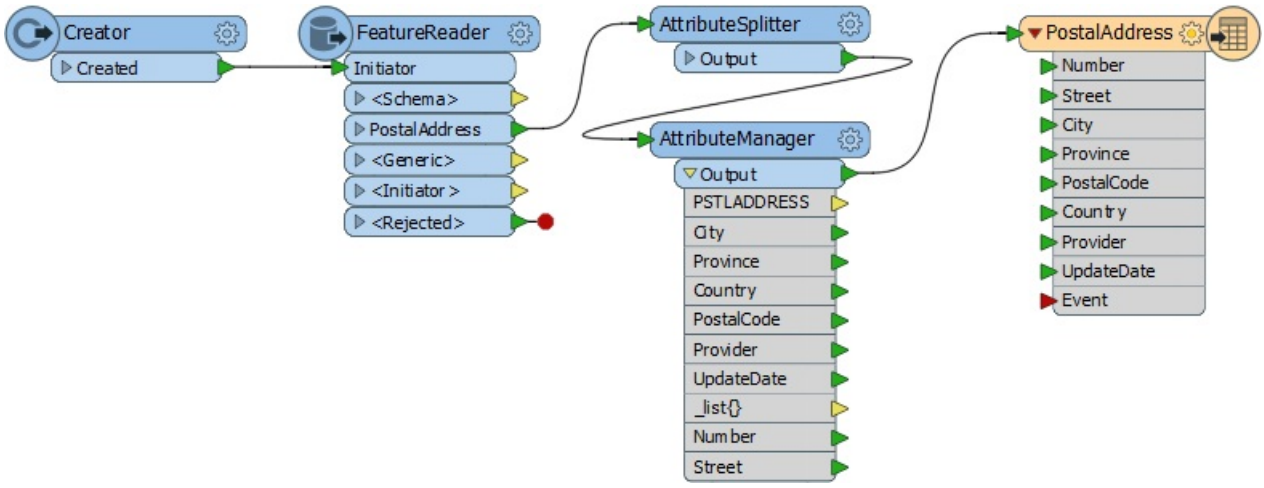
众所周知，市议员投票决定修改噪音控制法，生活在受影响地区的居民必须了解这些变化。

您的经理建议您接受任务，并且有一个紧迫的截止日期。

在项目的第一部分中，您创建了一个工作空间，用于将地址从地理数据库转换为Excel，同时映射模式。

该练习是该项目的第二部分：定位所有受影响的居民。您必须在距主要高速公路50米范围内找到所有单户住宅，并从地址数据流中过滤掉所有其他住宅。

1）启动**Workbench**
启动Workbench（如有必要）并从练习2打开工作空间。或者，您可以打开C：\ FMEData2018 \ Workspaces \ DesktopBasic \ Transformers-Ex3-Begin.fmw



工作空间已有一个FeatureReader用于读取地址，转换器用于编辑地址模式，还有一个写模块用于将数据写入Excel电子表格。

2）添加读模块（道路数据）
使用读模块>添加读模块为道路数据添加读模块。道路数据将用于确定与主干路线的距离。

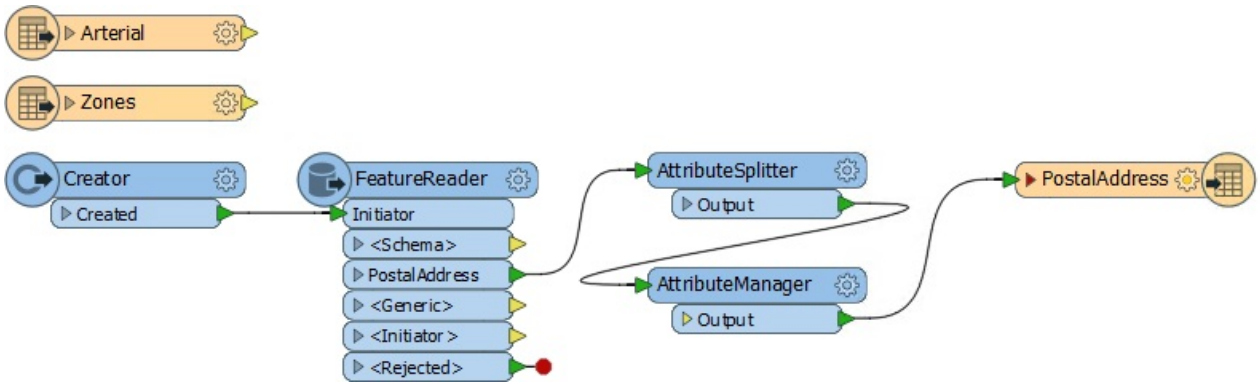
读模块格式	**Autodesk AutoCAD DWG / DXF**
读模块数据集	C:\FMEData2018\Data\Transportation\CompleteRoads.dwg

出现提示时，仅选择名为Arterial的要素类型（图层）。

3）添加读模块（分区数据）
使用Readers> Add Reader为分区数据添加读模块。分区数据将用于确定地址是否是单户住宅。

读模块格式	**MapInfo TAB（MITAB）**
读模块数据集	C:\FMEData2018\Data\Zoning\Zones.tab

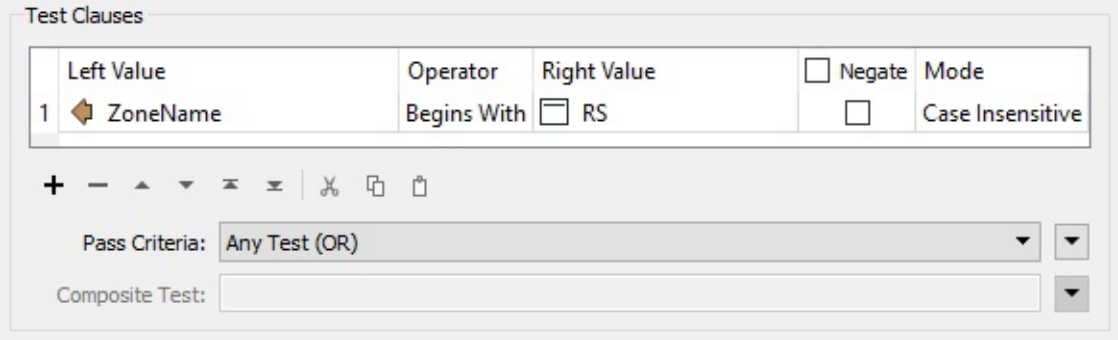
随着属性列表的折叠，工作空间现在看起来是这样的：



请随意检查所有源数据，以熟悉其内容。您甚至可以运行工作空间以确保所有缓存都是最新的。

4）添加**Tester**转换器
将Tester转换器添加到Zoning 要素类型。

该Tester将用于过滤其他分区的住宅区。所有单户住宅区都将以RS开头，因此Tester应如下设置：

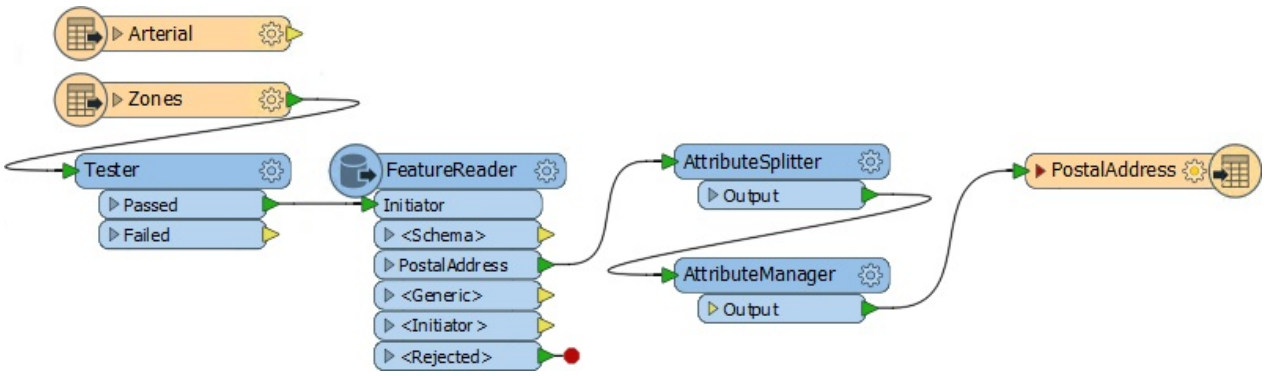


重要的是使用“Begins With”运算符设置测试。

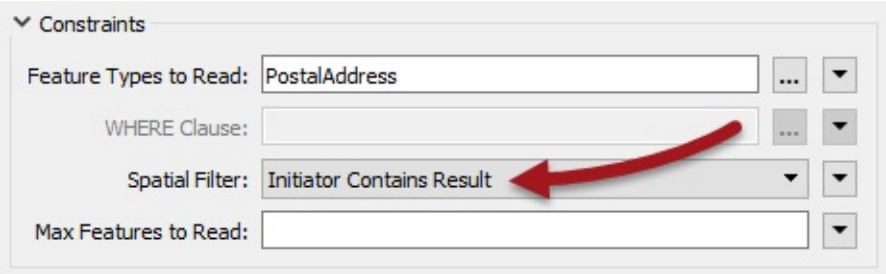
Vector小姐说.....
所以...为什么是Tester呢？为什么不使用AttributeFilter？你知道吗？

5）连接到**FeatureReader**
过滤数据的一种方法是使用SpatialFilter转换器，我们将使用道路要素来做这件事。但另一种方法是在FeatureReader转换器中使用过滤。

因此，删除Creator转换器并将Tester：Passed端口连接到FeatureReader：Initiator端口：



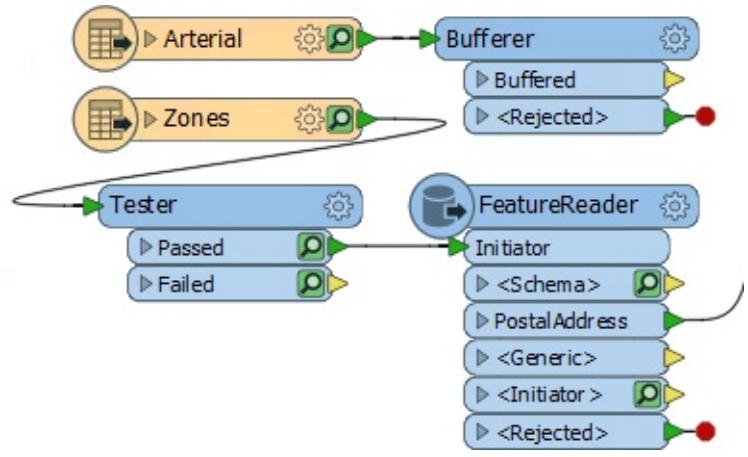
6）设置**FeatureReader**
现在检查FeatureReader的参数。将Spatial Filter参数设置为*Initiator Contains Result*：



这可确保只从数据库中读取属于单户居民区域的地址。确保打开要素缓存并运行工作空间。检查Tester:Passed和FeatureReader:PostalAddress缓存以确认结果是否正确。

7）添加**Bufferer**
现在我们可以确定哪个过滤地址落在主干路线的50米范围内。这次我们将使用SpatialFilter转换器。

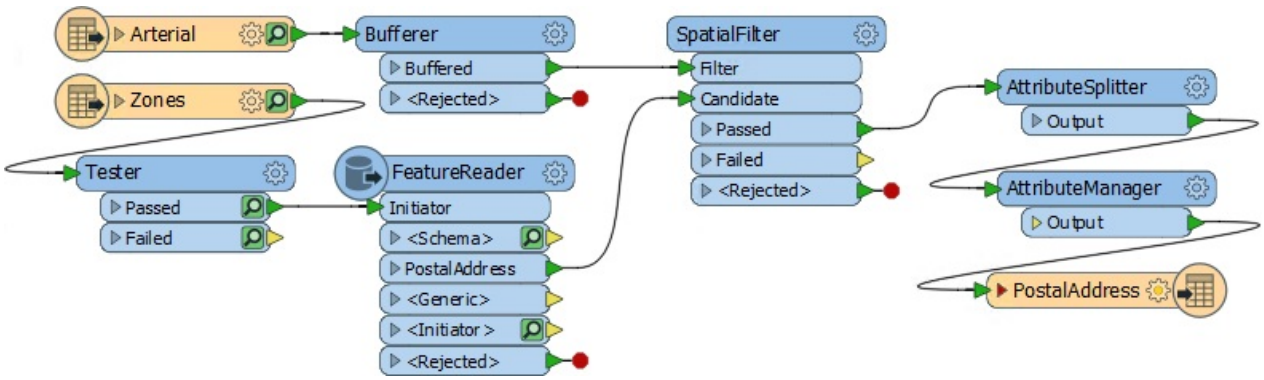
SpatialFilter没有“在X距离内”的测试，因此我们必须对它进行一些不同的设置。将Bufferer转换器添加到工作空间。将它连接到干道道路数据：



将Bufferer缓冲区数量参数设置为50。

技巧
您可以选择在Bufferer之后添加Dissolver转换器，以将所有缓冲区要素合并在一起。 转换结果将相同（就所选地址而言），但数据在FME Data Inspector中看起来会更好。

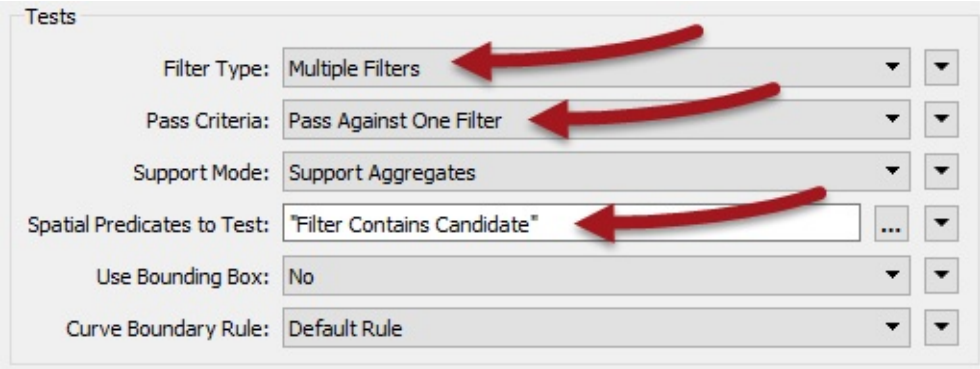
8）添加**SpatialFilter**
添加SpatialFilter转换器。buffered的主干路线是Filter.。Candidate端口可以连接在FeatureReader和AttributeSplitter之间：



这样，AttributeSplitter和AttributeManager仅在过滤的要素上运行。如果在AttributeManager之后连接了SpatialFilter，则数据将被处理然后被丢弃。

9）设置**SpatialFilter**参数
设置SpatialFilter参数，如下所示：

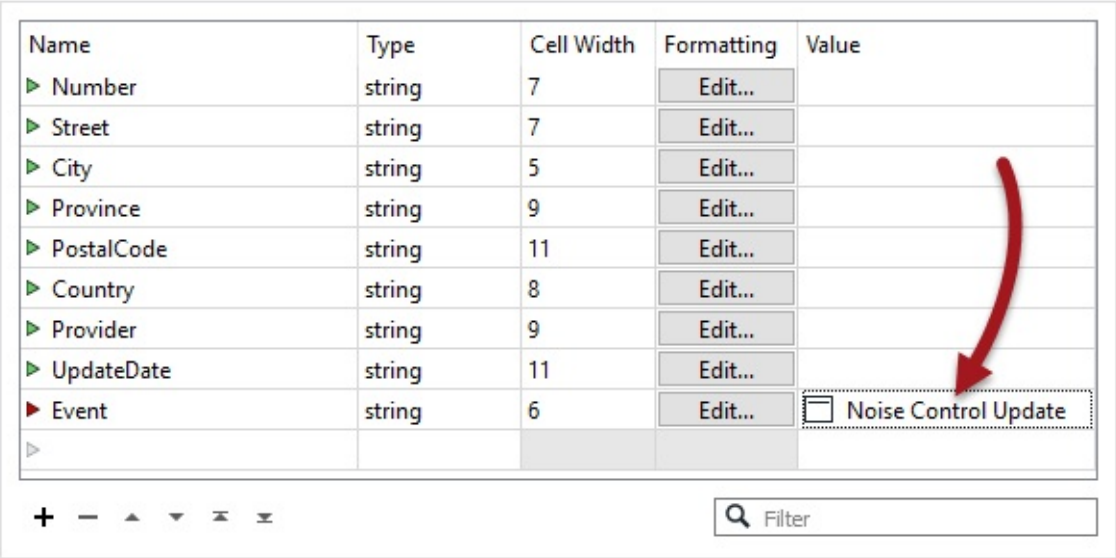
过滤类型	多个过滤器	有多个缓冲多边形
通过标准	通过一个过滤器	单个地址不能存在于所有缓冲区中
空间谓词测试	过滤器包含候选者Filter Contains Candidate	查找主干缓冲区中包含的地址



也就是说，有多个道路缓冲区，但地址只需要在一个缓冲区内传递，而不是全部。

10）运行工作空间
使用SpatialFilter上的**"**从此运行**"**运行工作空间。检查缓存输出以证明只有单户家庭区域和主干道路缓冲区内的地址已通过过滤过程。

11）设置事件**(Event)**字段
最后一步，打开Excel写模块的"要素类型"对话框，单击"用户属性"选项卡，然后为"事件"字段设置固定值：



关闭缓存，重新运行工作空间，并检查输出以确认数据集已正确写入。电子表格中应该有148条记录，随时可以发送给管理部门进行批量邮寄。

恭喜

通过完成本练习，您已学会如何：

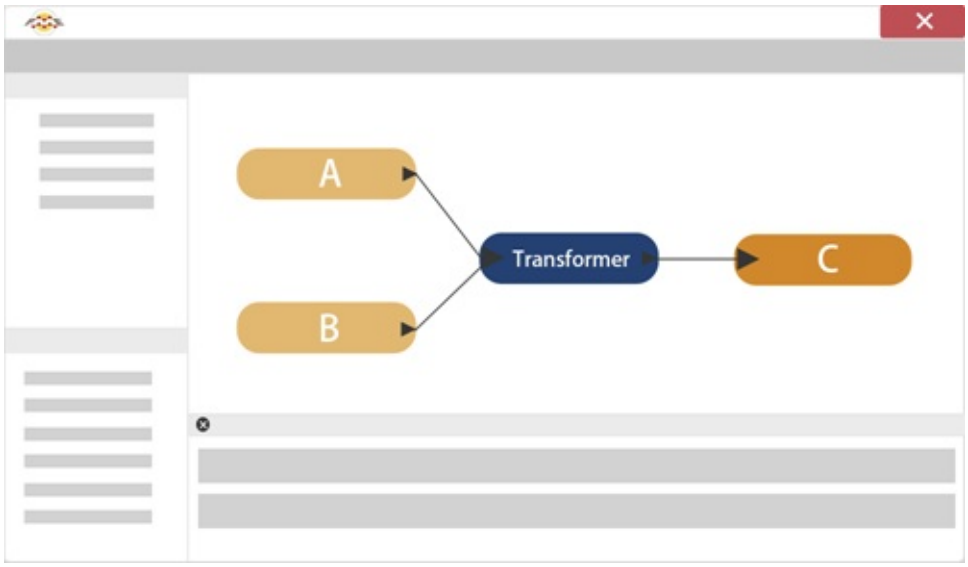
- 使用Tester转换器按属性值过滤
- 使用FeatureReader转换器中的空间过滤器选项
- 使用Bufferer转换器设置“在x距离内”测试
- 使用SpatialFilter转换器按几何对象过滤

数据连接

将数据连接在一起的转换器在FME中也非常常用。它与过滤有关，转换器类别称为过滤和连接。

什么是数据连接？

过滤转换器将数据分成不同的流，而其他转换器将数据流组合在一起，根据一组用户定义的条件合并数据。这里（例如）传入流A和B连接在一起形成一个新流C：



有必要做的不仅仅是将两个连接绘制到同一个输入端口以合并FME Workbench中的数据。以这种方式连接数据流只会将数据合并为一个流，而不是将它们融合在一起。您可以把这看作是一个数据联合。

新知识
为了强调这一事实，Junction转换器 - 它被设计为在不合并数据的情况下将多个连接汇集在一起 - 已在FME2018中被赋予联合的别名。

要合并数据，必须为连接的基础定义关系，这是通过多个转换器中的一个来完成的。

这些转换器不仅允许合并工作空间正在处理的数据，还可以合并数据库或其他外部数据集的连接。

FME中的连接可以基于匹配的属性值（DatabaseJoiner或FeatureMerger / FeatureJoiner），也可以基于空间关系，例如要素之间的重叠或从一个要素到另一个要素的近似度（NeighborFinder或SpatialRelator）。

基于键的连接转换器

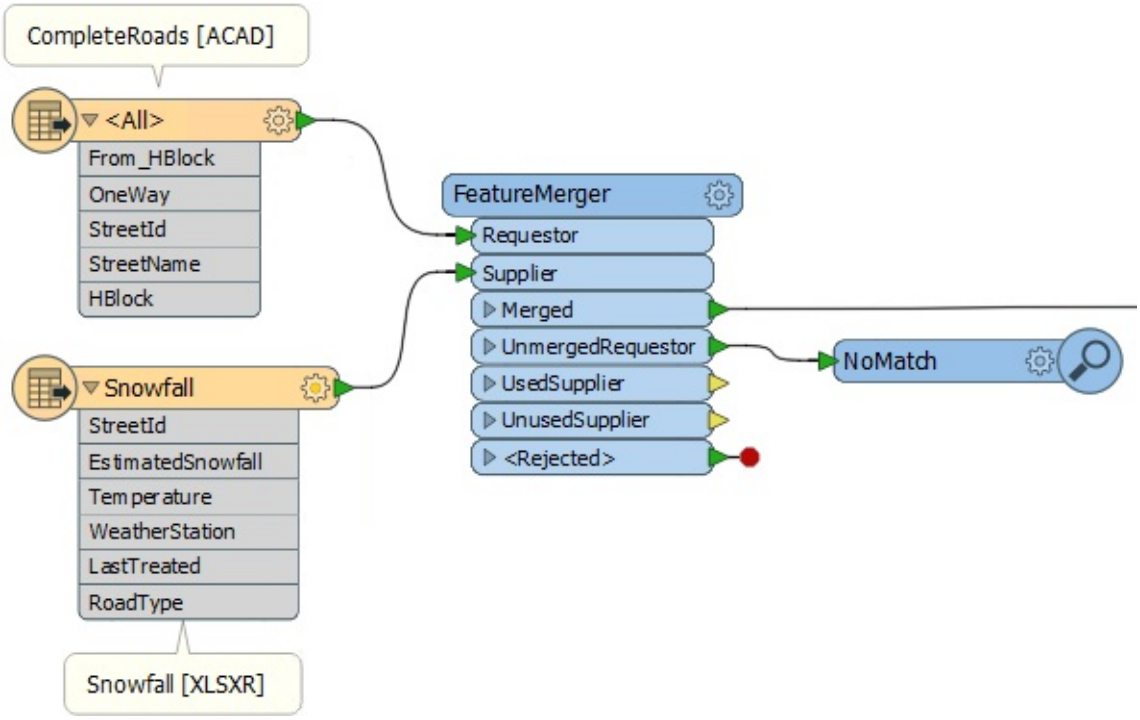
多个转换器可以通过匹配属性值（键）来连接数据。其中一些更倾向于几何对象，而另一些则更像SQL风格。一些在一个工作空间内连接数据流，而另一些则将一个数据流连接到外部数据库。

您使用哪种方法取决于您的连接要求和性能需求。

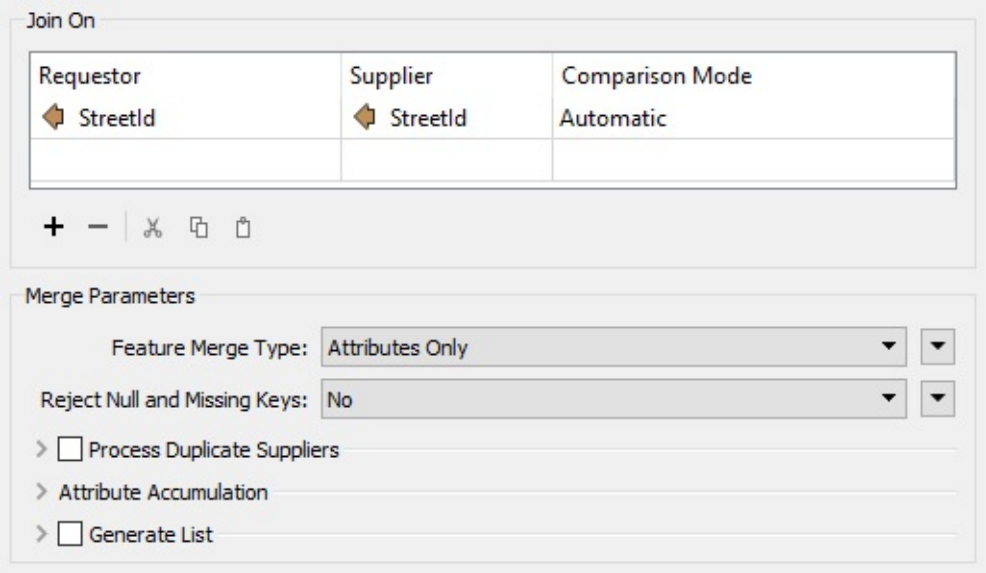
FeatureMerger

FeatureMerger是一个转换器，用于根据键字段匹配连接工作空间中的两个（或多个）数据流。

例如，这里的道路数据集具有StreetId编号。FeatureMerger用于将降雪信息电子表格中的信息组合到道路数据上：



FeatureMerger的参数对话框如下所示：



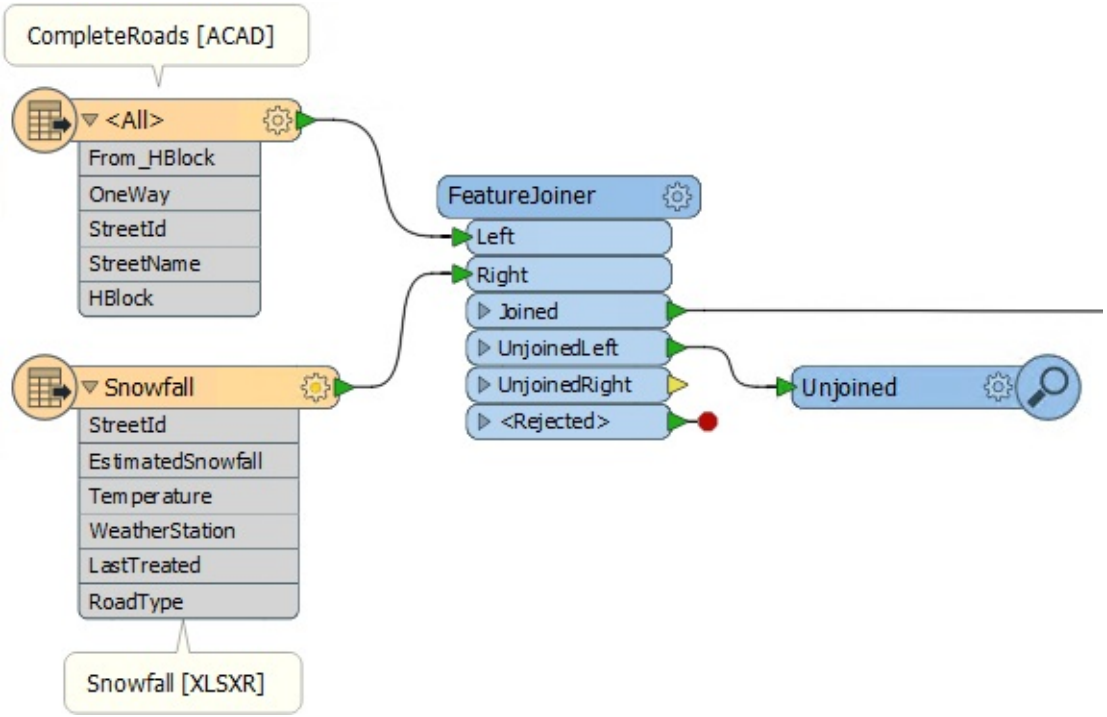
此屏幕截图显示使用StreetId作为键进行连接。具有匹配降雪记录的所有Requestor（Road）要素都通过Merged输出端口输出。没有匹配的所有Road要素都通过UnmergedRequestor端口输出以进行检查，以确定未发生匹配的原因。

还有其他参数可以处理信息冲突，重复键，以及是否仅合并属性或几何对象。

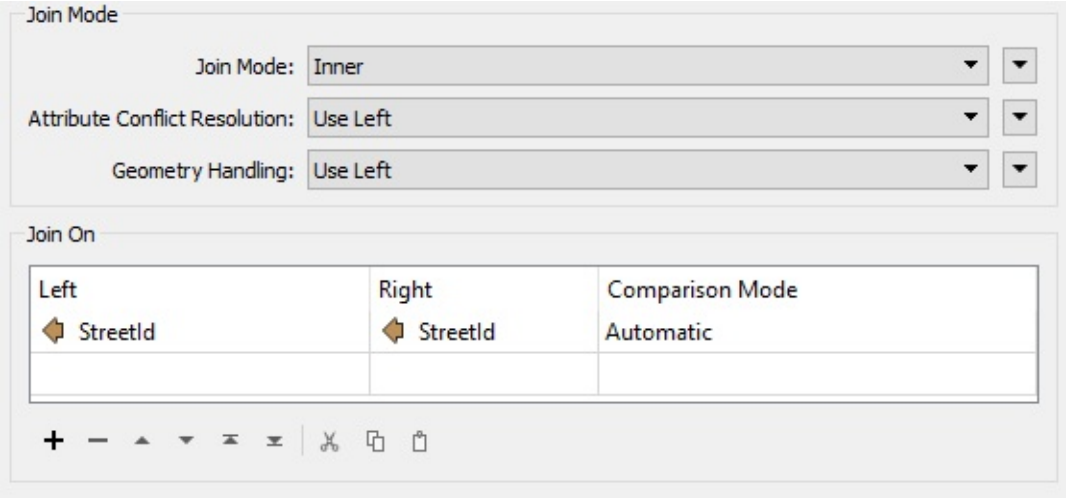
FeatureJoiner

FeatureJoiner是另一个转换器，用于根据键字段匹配连接工作空间中的两个数据流。

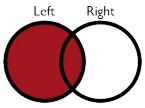
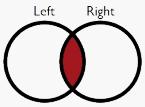
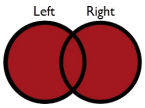
新知识
请记住，FeatureJoiner是FME2018的新转换器，旨在最终取代FeatureMerger。



例如，在这里，FeatureJoiner中的Roads / Snowfall相同。转换器的参数如下所示：



如您所见，此转换器更多地基于传统的SQL查询。Join Mode参数可以采用以下三个值之一：

模式	描述	描画	**Joined**输出	**Unjoined Left**	**Unjoined Right**
左(Left)	左侧要素会查找匹配项，无论它们是否找到匹配项，都会输出		所有匹配以及不匹配的左侧要素	无	未使用的右侧要素
内部(Inner)	左侧要素会查找匹配项，如果找到则会输出		仅所有匹配项	不匹配的左侧要素	未使用的右侧要素
完全(Full)	左侧和右侧两者都通过Joined输出端口输出，无论它们是否找到连接		所有匹配以及不匹配的左右侧要素	无	无

您可能熟悉的其他术语是外部连接和右侧连接。外部联接只是完全连接在此处所做的不同名称。要执行右侧连接，您可以切换将哪些要素发送到哪个输入端口并使用“左侧连接”选项。

警告
这里要注意的关键是为每次发生的匹配输出一个要素。 例如，如果1道路要素匹配5个降雪记录，则将有5个要素输出为已连接。 Joined要素始终作为连接输出。Left，Inner和Full实际上只控制Joined输出中包含哪些不匹配的记录。

使用左连接时，用户要么认为所有道路都有匹配的降雪记录，要么如果没有匹配也无关紧要。实际上，Unjoined Left输出端口不会显示任何要素。

如果必须确保匹配，则所选模式应为内部(Inner)。然后，离开Unjoined Left输出端口的记录可被视为错误，并调查为什么没有匹配。

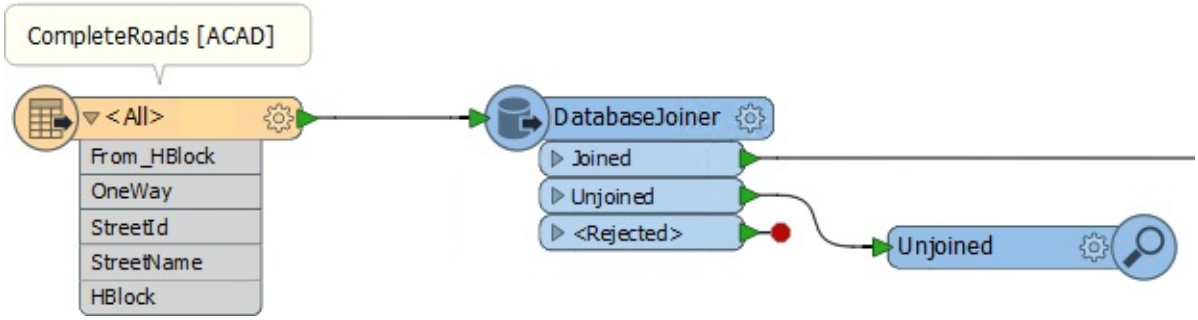
与FeatureMerger一样，有一些参数可以处理信息冲突以及是否仅合并属性或几何对象。

技巧
因此，FeatureMerger和FeatureJoiner之间的关键区别在于有多个匹配会发生什么。 如果FeatureJoiner有多个匹配项，它将输出多个要素。 如果FeatureMerger有多个匹配项，它将仅输出一个要素。该要素将只有一个匹配的记录，或匹配的记录列表，具体取决于Process Duplicate Suppliers参数。

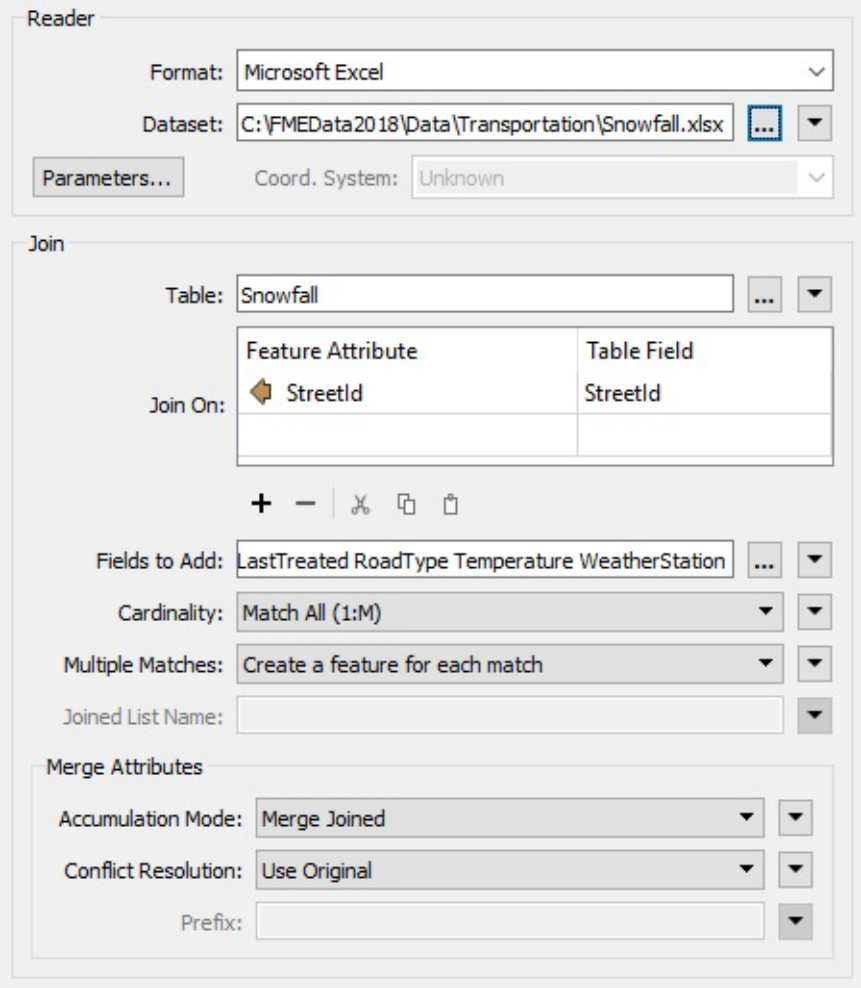
DatabaseJoiner

DatabaseJoiner转换器与FeatureMerger和FeatureJoiner不同，因为它不是合并两个要素流，而是将一个（或多个）数据流与来自外部数据库的记录合并。

以下是与上面的FeatureMerger相同的示例。在这种情况下，道路要素直接从Excel电子表格中的表格获取降雪数据：



DatabaseJoiner的参数对话框如下所示：



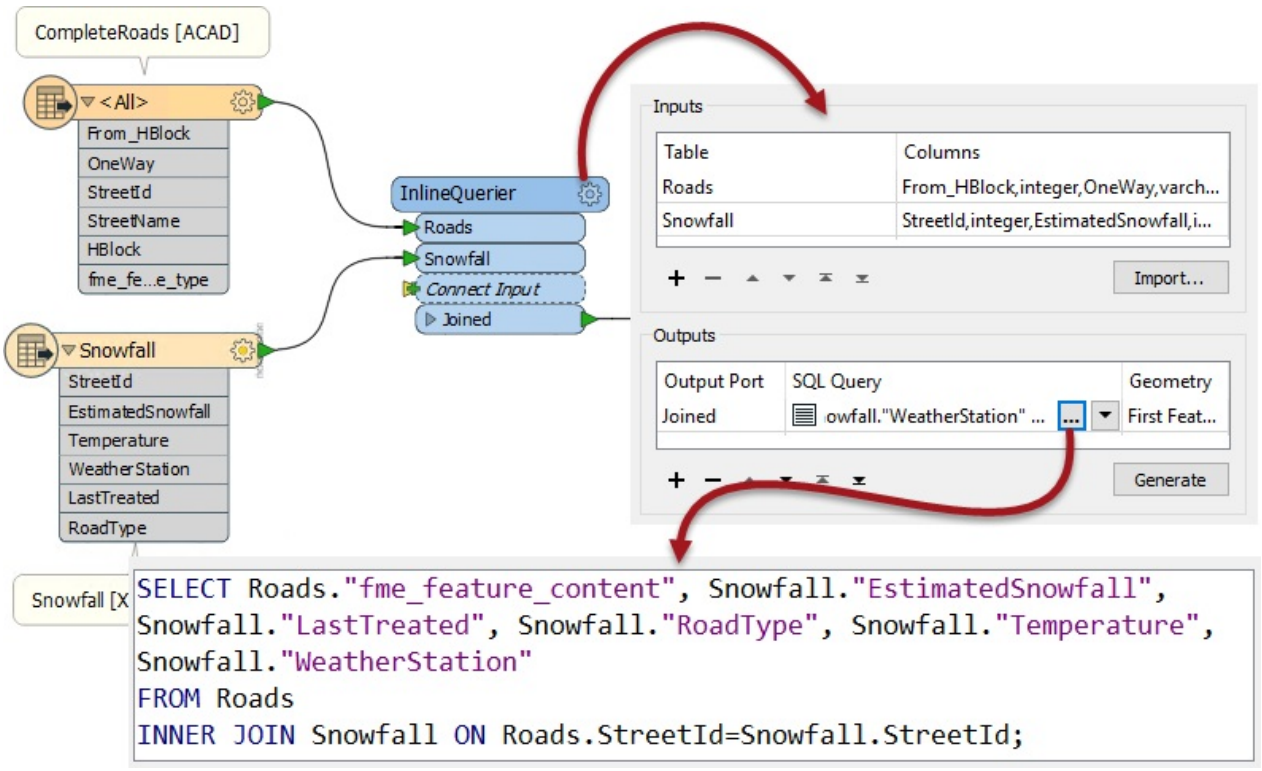
同样，从要素和数据库表中使用AddressID以促进两者之间的合并。

与其他转换器一样，有一些参数可以控制聚合的属性以及如何解决冲突。

副驾驶**Transformer** 说.....
DatabaseJoiner与FeatureMerger相比具有许多优势。首先，它具有控制如何处理多个匹配的参数，以及用于优化数据库查询的参数。 其次，它允许连接要素而无需将整个数据集读入工作空间。FME可以只查询数据库并选择所需的单个记录。这可以大大提高性能。当然，它确实要求提供者(supplier)记录以适当的数据库格式存储！

InlineQuerier

InlineQuerier转换器接受工作空间中的要素并生成临时数据库。使用该数据库，可以在多个表中应用所需的任何SQL命令（包括连接）：



InlineQuerier具有明显的优势，在单个转换器中允许它的输入重复使用多次; 而多连接则需要多个FeatureJoiner转换器。但是，生成该初始数据库会产生性能开销。

基于空间的连接转换器

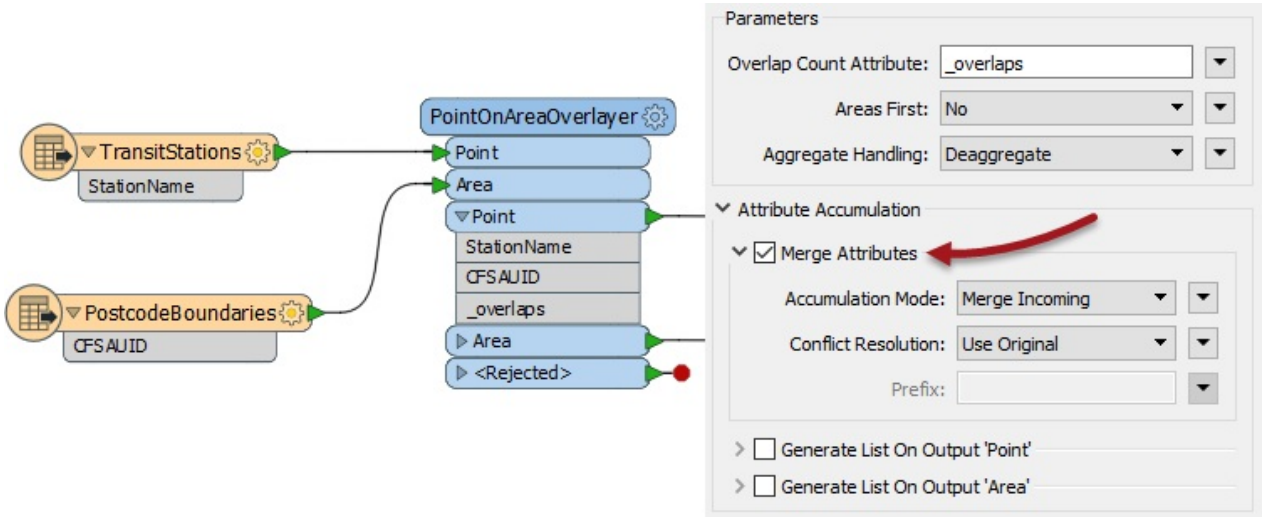
多个转换器可以通过空间关系连接数据。您使用哪个取决于要测试的空间关系和您的确切连接要求。以下是一些主要的转换器。

Overlayers

有许多不同的“overlaye”转换器，每个转换器处理不同形式的重叠层。

例如，PointOnAreaOverlayer在落在面（多边形）要素内的点上执行空间连接。此操作有时称为“多边形点”重叠。

正如帮助解释的那样，“每个点都接收包含它的面的属性，每个包含面接收它包含的每个点的属性。”



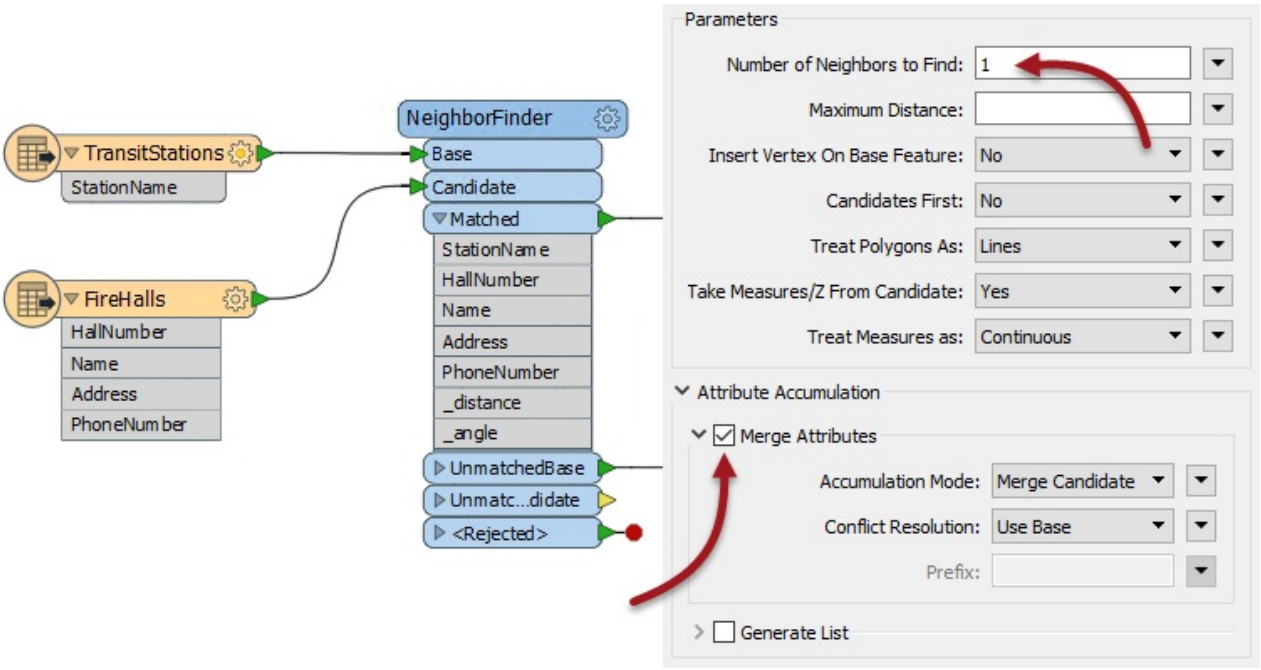
中转站(TransitStation)要素在这里提供了邮政编码（CFSAUID），具体取决于它们落入哪个PostcodeBoundary多边形。

"_overlaps"属性是此转换器的另一个有用结果。它告诉我们每个站落入多少个多边形; 在这种情况下，重叠的邮政编码可能被具有多于一个重叠的站点发现。

相反，面输出将具有 "_overlaps" 属性，该属性将告诉我们每个邮政编码中有多少站点。

NeighborFinder

NeighborFinder转换器基于临近关系执行空间连接。在这里，NeighborFinder用于识别每个中转站最近的消防站：



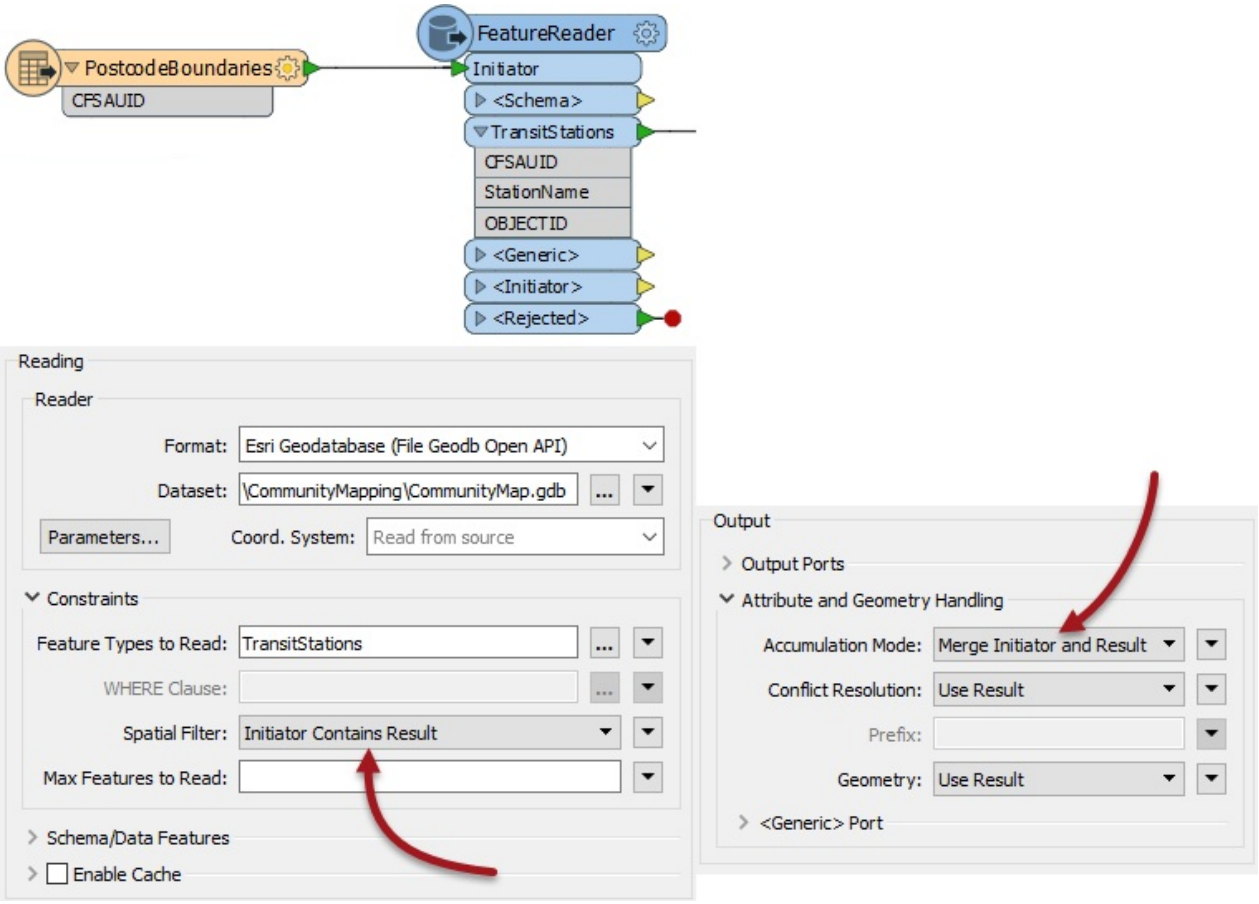
消防站号码，名称，地址和电话号码属性将合并到每个设施（Facility）要素以及许多有用的属性（并非全部显示），记录最近的消防站的X/Y坐标，方向和距离。

NeighborFinder的参数包括指定关系的最大距离或要查找的最大邻居数的能力。

FeatureReader

FeatureReader是DatabaseJoiner转换器的空间等价物。它从外部数据集读取，并基于初始要素和数据集中的要素之间的空间关系形成匹配。

一个区别是输出不是原始要素，而是查询要素; 因此命名为FeatureReader:

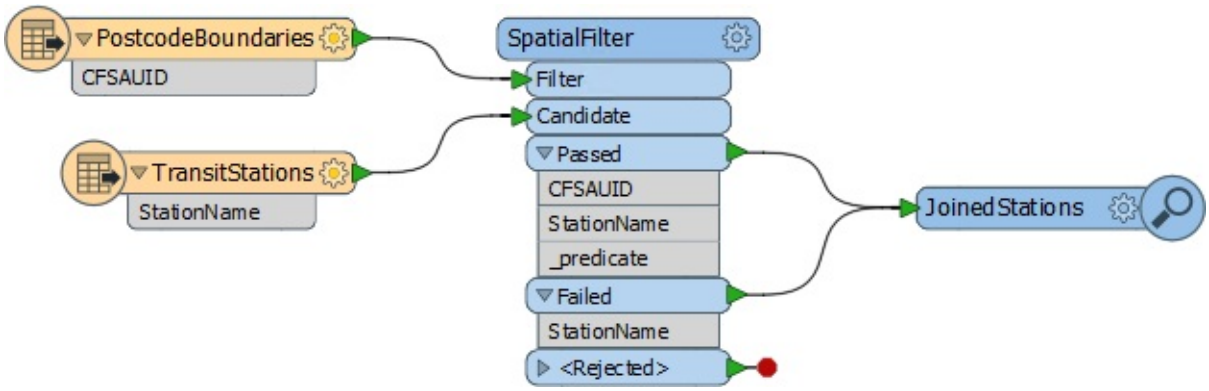


例如，这里使用FeatureReader来执行中转站和邮政编码的相同叠加。PostcodeBoundaries要素被读入工作空间，用于对中转站进行空间查询的一种方法（地理数据库中的一个表）。检索站点时将使用它们所属的邮政编码要素的属性。

这也可以作为一种过滤器的形式，因为除非站点落在邮政编码边界内，否则不会输出。

SpatialFilter

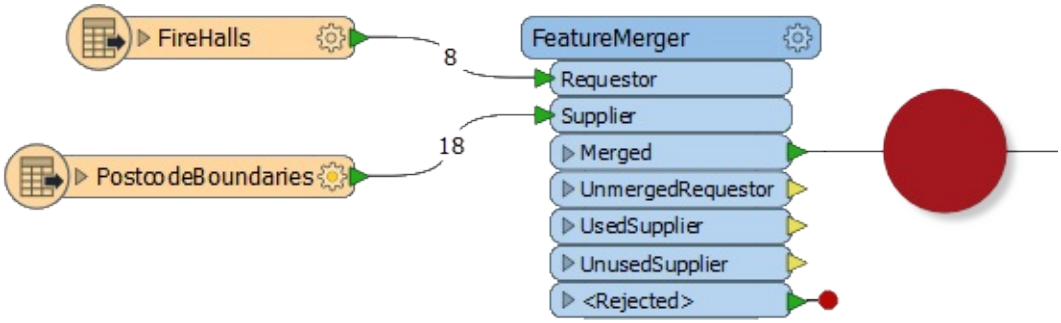
SpatialFilter - 顾名思义 - 根据空间关系过滤数据。但是，它也会将属性从一个要素合并到另一个要素，因此可以说是一种空间连接。



重要的是连接Passed和Failed输出端口，除非您还要过滤数据。

Vector小姐说...

这是关于数据连接的问题。看下面的截图，然后回答输出连接中会出现多少要素...



1. 8
2. 18
3. 26
4. 说不清楚

练习：犯罪映射数据请求

练习**4**	噪音控制法项目（犯罪数据加入）
数据	犯罪统计（CSV）
总体目标	在犯罪统计数据和地址要素之间进行连接
演示	基于属性的连接
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformers-Ex4-Begin.fmw
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\Transformers-Ex4-Complete.fmw

如你所知，市议员投票决定修改噪音控制法，生活在受影响地区的居民被告知这些变化。。

在项目的第一部分中，您创建了一个工作空间，用于将地址从地理数据库转换为Excel，同时映射模式。在项目的第二部分，您继续工作空间，找到距离主要高速公路50米范围内的所有单户住宅，并从地址数据流中过滤掉所有其他住宅。

现在，一家全国性报纸的数据记者担心，放宽噪音控制法可能会导致该市犯罪率上升。因此，他们要求每个受影响地址的最近犯罪数字。他们打算将此与未来数据进行比较，看看他们的理论是否正确。

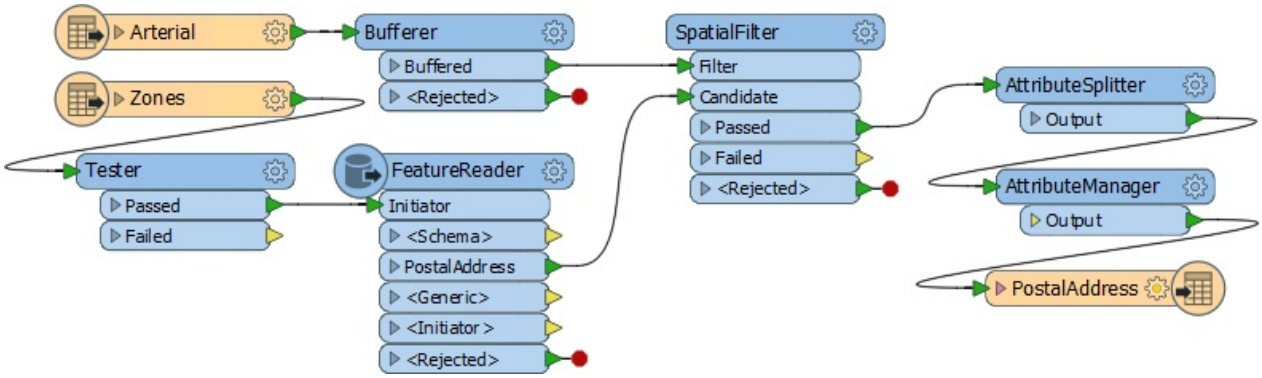
该请求是对该城市开放数据政策的重要考验，并且不存在不遵守的问题。然而，由于当前的犯罪数据集（CSV，非空间）没有以任何方式加入地址数据库，因此出现了危机。

因此，对于该项目的最后部分，您必须采用现有的噪声控制工作空间并对其进行修改以纳入犯罪统计数据。

完成这个任务，你就会成为一个空间超级英雄!

1）启动**Workbench**
启动Workbench（如有必要）并从练习3打开工作空间。或者，您可以打开C：\ FMEData2018 \ Workspaces \ DesktopBasic \ Transformers-Ex4-Begin.fmw

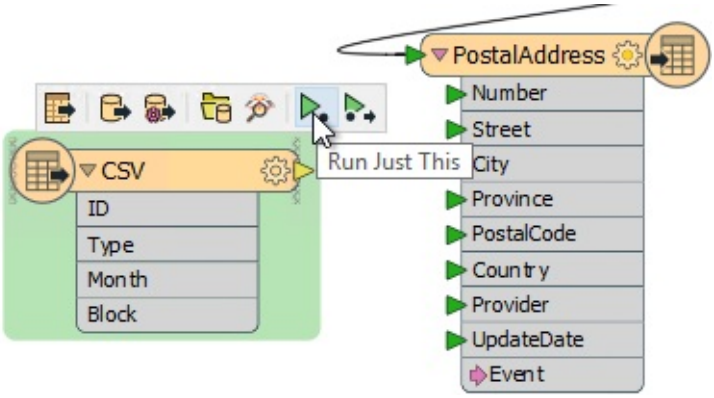
工作空间已设置为读取地址，在空间上过滤它们，并将它们写入Excel电子表格。



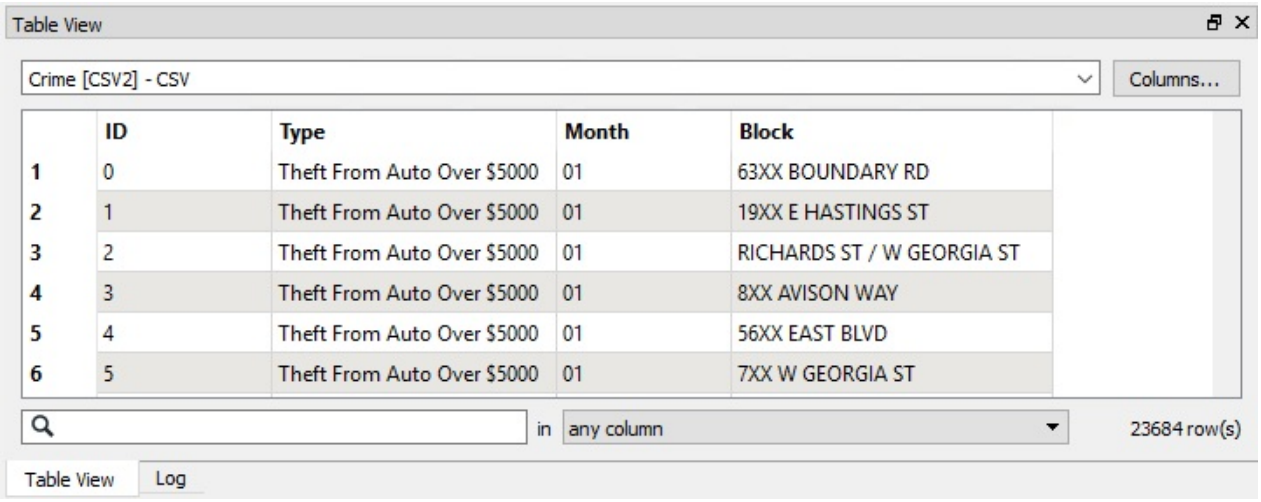
2）添加读模块
现在让我们开始处理犯罪数据。使用FeatureReader没有任何好处，因此使用菜单栏中的“读模块”>“添加读模块”将读模块添加到工作空间。使用以下参数：

读模块格式	逗号分隔值（**CSV**）
读模块数据集	C:\FMEData2018\Data\Emergency\Crime.csv
读模块参数	字段:分隔符字符:,(逗号) 字段:字段名称行:1

3）检查数据
下一个任务是熟悉源数据。启用要素缓存并使用Run Just This运行CSV读模块：



检查缓存的数据。它将在Data Inspector表视图窗口中显示如下：



注意没有这样的空间数据，但是有一个块号。

警察局长**Webb-Mapp**说.....
犯罪？！在我的城市？我想不是。但如果有.....请注意，7XX W Georgia Street意味着安大略街以西乔治亚街的第七个街区，覆盖700-800号楼。7XX E Georgia Street距离安大略省东部第七个街区14个街区。明白了吗？

您可能已经发现每个地址要素都有一个数字（不是像“7XX”那样的块ID），并且道路数据以首字母大写（“W Georgia St”）存储在道路数据集中，而犯罪数据集是大写（“W GEORGIA ST”）。

这两者都会使两组数据加在一起变得更加困难，但并非不可能。

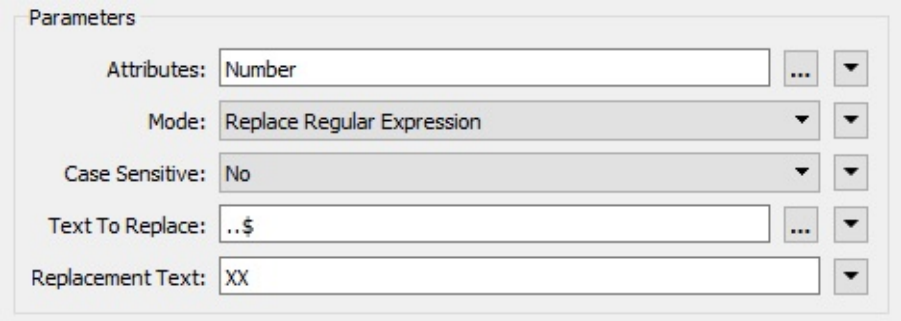
4）添加**StringReplacer**
要合并数据，我们需要将地址编号减少到与结构中的犯罪数据匹配的块编号; 例如，我们需要74XX而不是7445。

因此，添加StringReplacer转换器并在AttributeManager和PostalAddress要素类型之间进行连接。

设置以下参数：

属性	数字
模式	替换正则表达式
要替换的文本	..$
替换文本	XX

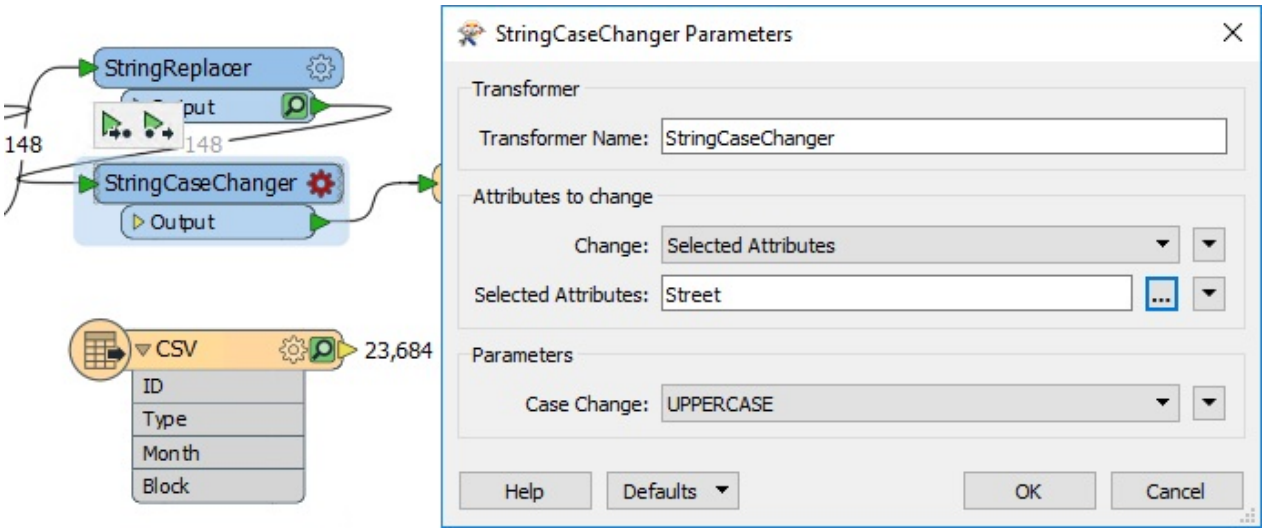
此要替换的文本（.. $）表示替换字符串的最后两个字符，并将它们替换为XX以匹配犯罪数据。



运行工作空间（使用StringReplacer上的*Run to This*）并检查缓存以确保转换器按预期工作。

5）添加**StringCaseChanger**
犯罪/道路数据的另一个区别在于大写/ 首字母大写街道名称。可以使用StringCaseChanger转换器修复此差异。

在StringReplacer之后添加一个StringCaseChanger转换器并设置参数以将Street的值更改为大写：



Vector小姐说.....

那么，回答我。为什么我们使用StringCaseChanger而在地址数据上（改为大写）不是更改犯罪数据（到首字母大写）？你知道吗？

6）构建连接密钥
更新了属性以匹配犯罪数据，我们现在必须构造建一个密钥。

在StringCaseChanger之后将AttributeManager添加到画布。创建一个名为JoinKey的新属性。打开属性的文本编辑器，然后输入（选择）：

```
@Trim（@Value（Number）@Value（Street））
```

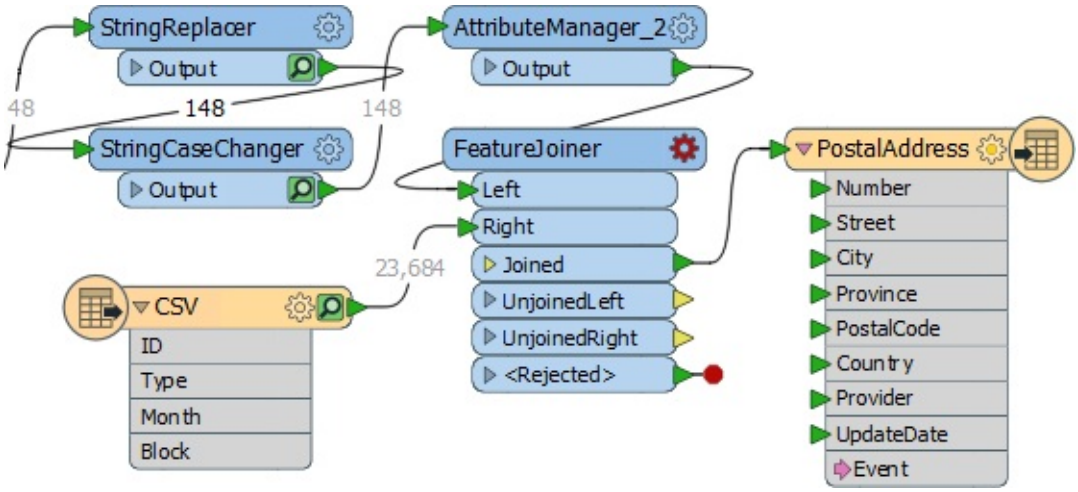
这将匹配犯罪数据的结构（确保在两个属性之间包含空格字符）。Trim函数用于确保这些属性上没有多余的空格。

7）添加**FeatureJoiner**
现在我们已经整理了连接键的结构，我们可以用FeatureJoiner将数据合并在一起。

将FeatureJoiner添加到画布。

将地址数据（AttributeManager输出）连接到左输入端口。将犯罪数据（CSV读模块要素类型）连接到右输入端口。

将FeatureJoiner:Joined输出端口连接到PostalAddress写模块要素类型：



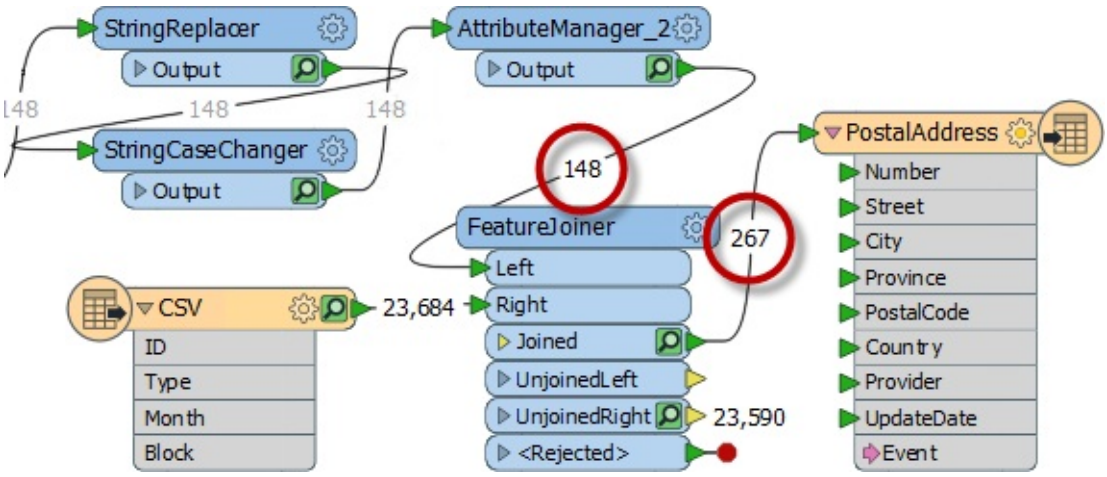
检查FeatureJoiner的参数。

对于Join Mode，选择*Left*。这意味着我们希望输出所有地址，无论它们是否与犯罪记录相匹配。

在Join On参数中，为Left值选择JoinKey属性，为Right值选择Block属性。

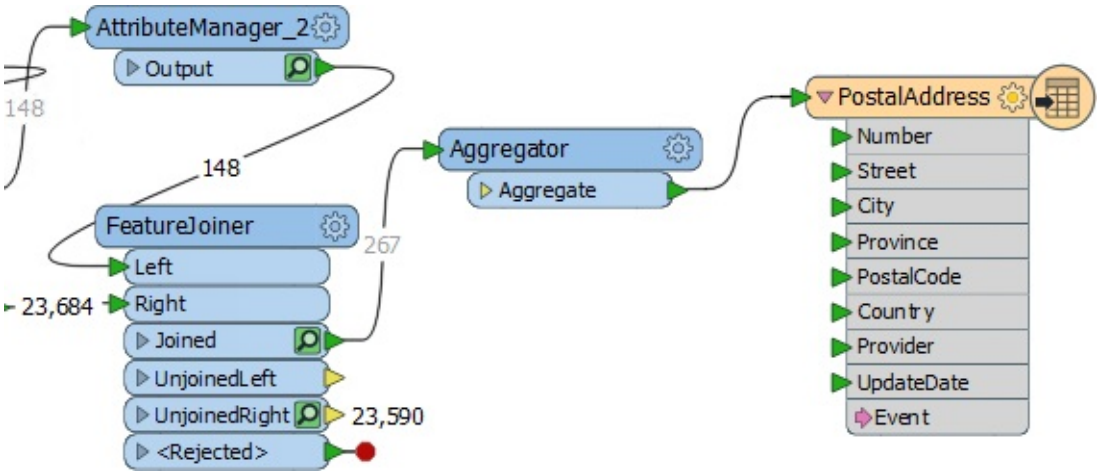
运行工作空间的该部分以查看此转换的结果。

8）添加**Aggregator**
有趣的是，尽管有148个地址进入FeatureJoiner，但却有267个地址出现：



那是因为每个区块有多个犯罪，并且数据有267个匹配。

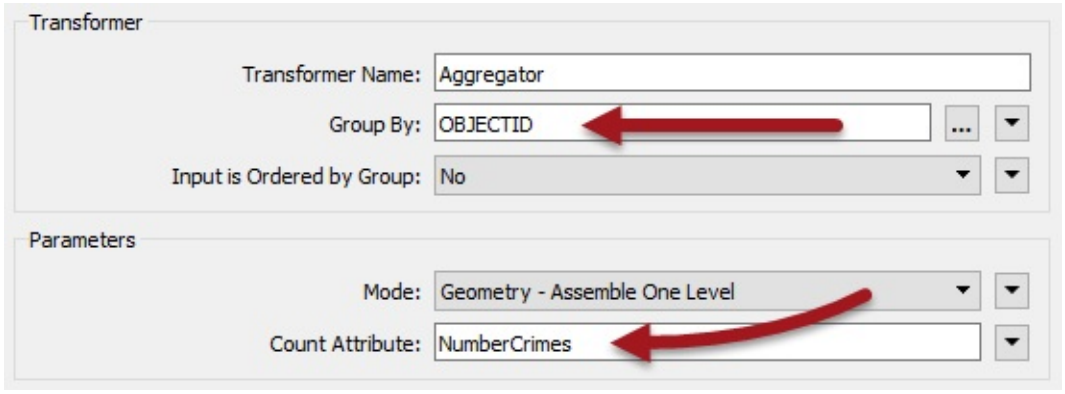
我们可以使用Aggregator转换器将这些数据聚合在一起。因此，在FeatureJoiner：Joined端口之后放置一个Aggregator转换器：



检查参数。我们需要通过选择将匹配回原始地址的属性来设置group-by参数。每个地址都没有ID，因为我们在上一步中删除了它们，因此：

- 返回到AttributeManager，撤消OBJECTID的Remove选项，并使用OBJECTID作为Aggregator分组
- 使用UpdateDate作为 Aggregator 分组（因为每个地址都将收到一个唯一的时间戳）

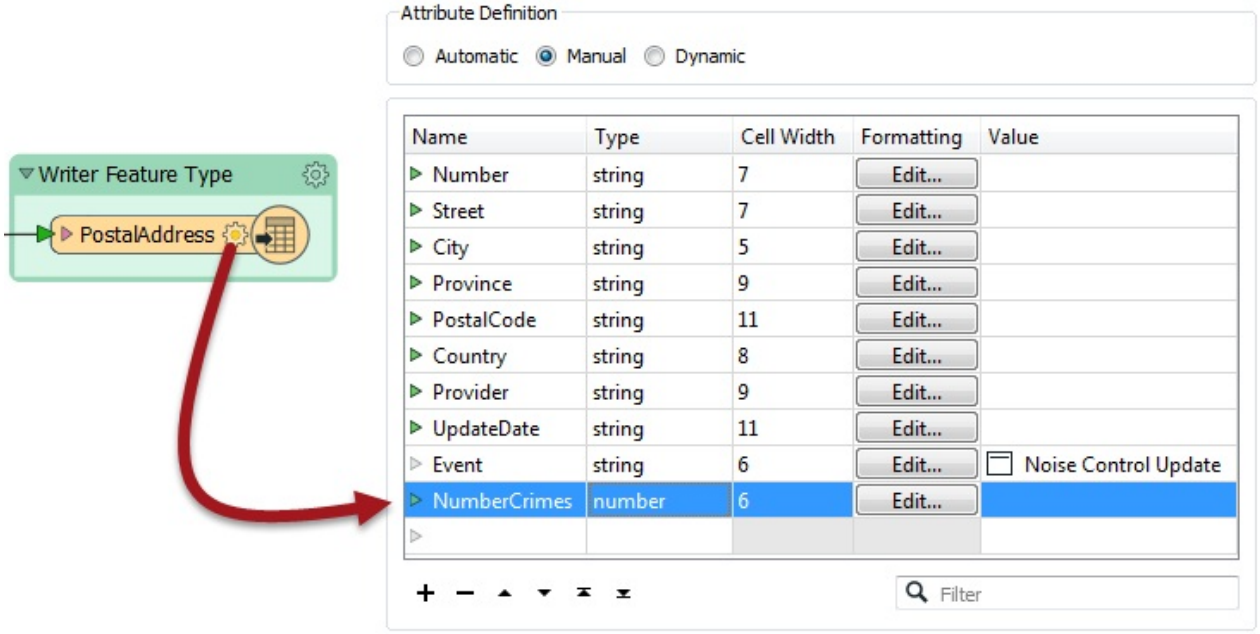
然后将Count属性设置为NumberCrimes的值：



如果展开PostalAddress写模块的属性，您会注意到没有出现NumberCrimes。当我们在之前的练习中编辑此写模块的用户属性时，我们将其属性定义从自动更改为手动。这意味着它不会自动更新以添加转换期间创建的新属性。因此，我们必须切换回自动（这会带来许多其他不需要的属性），或者只是在这里添加一个名为NumberCrimes的新属性。给它类型“数字”和单元格宽度6.来自Aggregator的数据现在将在写模块上有它的属性。

9）写入数据
现在我们知道工作空间正常运行，关闭要素缓存。没有必要为生产中的工作空间缓存所有内容。

现在，通过添加NumberCrimes属性来编辑PostalAddress属性模式：



最后，重新运行工作空间并在Data Inspector中检查结果。数据将包括犯罪数量，并且重新处理属性意味着单个地址也被匿名化了。这很重要，因为这些数据正在公开。

恭喜

通过完成本练习，您已学会如何：

- 预处理数据以获得具有匹配结构的连接键
- 构建一个用于FeatureJoiner的连接键
- 使用FeatureJoiner中的连接键加入非空间数据
- 使用一个Aggregator 转换器来合并连接并计算连接数

模块复习

该模块旨在向您介绍更广泛的FME转换器，以及更有效地应用转换器的技术。

你应该从这个模块中学到什么

以下是本次课程要学习的重点：

理论

- 除了转换数据属性或几何对象外，还有一些不同的转换器组可以工作
- 集成的编辑对话框允许作者用内置函数替换转换器
- 大部分最常用的转换器与属性处理有关
- 过滤是分割数据的行为。条件过滤是使用测试或条件划分数据的行为
- 数据连接由转换器执行，这些转换器从Workbench或外部数据源合并数据

FME技能

- 能够在不事先知道转换器的情况下定位转换器以执行特定任务
- 使用集成工具构建字符串和计算算术值的能力
- 能够使用通用转换器进行属性管理
- 能够使用转换器过滤和分割数据
- 能够使用转换器将数据合并在一起

进一步阅读

为了进一步阅读，为什么不浏览特定转换器的博客文章，例如**TestFilter**，**AttributeCreator**或**FeatureMerger**？

练习1答案

以下是本章问题的答案。

你的同事问道.....
问）Web类别中最常用的五大转换器是什么？ 答）在撰写本文时，它们是： 1. HTTPCaller 2. JSONFlattener 3. ParameterFetcher 4. Generalizer 5. JSONFragmenter 您可以通过Safe Software网站上的Transformers Gallery找到它。 尽管Generalizer似乎是一个奇怪的Web转换器，但它可用于减少几何对象中的顶点数量，使要素更小，更适合在Web上使用。

你的同事问道.....
问）对于每个字符串，“快速添加”(Quick Add)返回哪些转换器？ 答）它们是： 1. lineco：LineCombiner 出现在转换器的名字开始的字符串的示例 2 REAC：AreaCalculator 出现在转换器的名字中间的字符串的示例 3 RBIC：RasterBandInterpretationCoercer 表示转换器的驼峰命名法名称的字符串的示例 4. drape：SurfaceDraper 描述转换器执行操作 的字符串的示例 5. attributeexpl：AttributeExploder 太长的字符串的示例。“eexpl”会更有效率

你的同事问道.....

问）哪种转换器适用于这些情况？
答）在FME中执行任务的方法通常很多，但一般来说最好的转换器如下：
1。我们在文件中有一些文本行，并希望读取该文本并将其添加为属性。
AttributeFileReader

2.我们有一组矢量等高线，想要通过转置X和Z坐标来创建一个横截面。
CoordinateSwapper

3.我们有一个点云数据集，并希望通过重新采样来删除多余的点来减小其大小。
PointCloudThinner

4.我们有一个文本字符串，想要找出字符串包含的字符数。
StringLengthCalculator

5.我们有一组地址，并且每个地址都想找到最近的两个图书馆。
NeighborPairFinder

恭喜

通过完成本练习，您已学会如何：

- 使用在线转换器库(Transformer Gallery)
- 使用“快速添加”字符的各种组合来搜索转换器
- 使用Workbench转换器库和“快速添加”(Quick Add)搜索功能查找理想的转换器

Q + A答案

问题

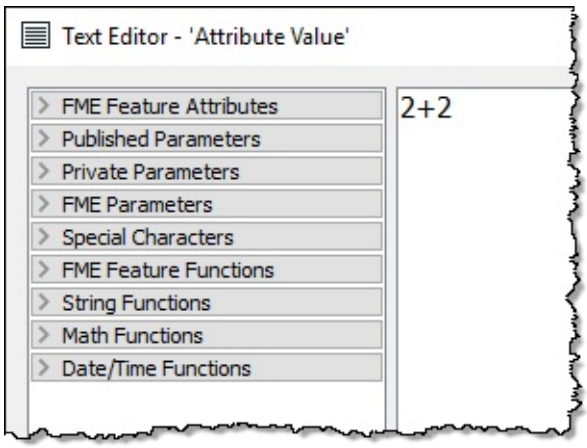
以下是本章问题的答案。

Vector小姐说**.....**
以下哪一项不属于转换器？ 1.属性 2.计算 3.数据质量 4.工作流程

Vector小姐说.....	
这里有四个转换器和四个类别。将转换器与正确的类别匹配。	
转换器	类别
Chopper	几何
Terminator	工作流程
Matcher	数据质量
DateTimeConverter	字符串
您是否通过转换器库查找这些？更快捷的方法是查看转换器的帮助页面。在我们讨论这个话题的时候，转换器可以属于不止一个类别。DateTimeConverter（例如）属于字符串和计算值。	

Vector小姐说.....

查看编辑对话框的屏幕截图，告诉我返回属性的值是：



1. 2 + 2
2. 4
3. 4.0
4.错误！

关键是要注意这个对话框的标题是“字符串编辑器(String Editor)”。因此，返回此属性的值将是文字字符串“2 + 2”。如果用户希望添加2 + 2来获得4，他们应该使用算术编辑器！

Vector小姐说.....

过滤和连接类别中有多少转换器出现在前30个最有价值转换器列表中？
1. 1
2. 4
3. 7
4. 10

截至2018年1月（FME 2018.0）有十个。它们是Tester（第1个），FeatureMerger（第4个），TestFilter（第8个），Aggregator（第12个），AttributeFilter（第13个），FeatureReader（第14个），SpatialFilter（第21个），GeometryFilter（第22个），DuplicateFilter（第27个）和Sampler （第28个）。因此，三分之一的顶级转换器来自过滤和连接类别！

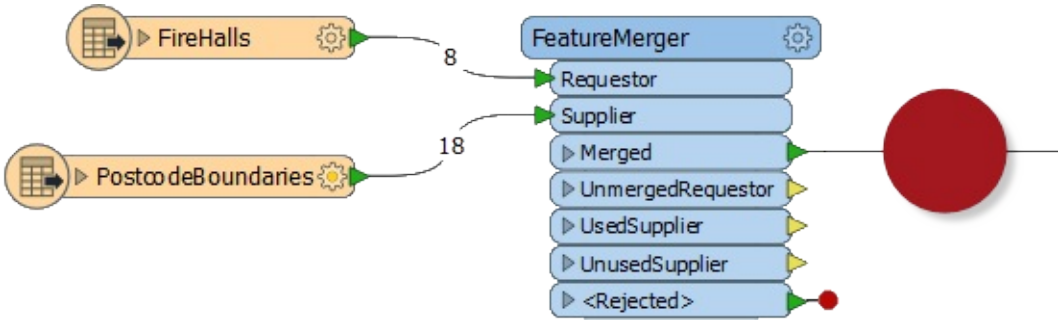
Vector小姐说.....

所以...为什么是Tester呢？为什么不使用AttributeFilter？

因为我们只需要以简单的Yes / No格式测试一个值。AttributeFilter更适合测试多个值。此外，AttributeFilter不允许我们进行"以...开始（Begins With）"测试。

Vector小姐说.....

看看下面的屏幕截图，然后回答输出连接中会出现多少要素.....



1. 8
2. 18
3. 26
4. 说不清楚

从截图中看出是不可能的，因为你不知道有多少属性值是匹配的。因为有八个消防站，它将从0到8，但这就是我们能了解的全部。事实上，如果它是一个FeatureJoiner转换器，那么如果每个消防站都与每个邮政编码边界相匹配，那么可能会有多达144个匹配！

Vector小姐说.....

为什么我们在道路数据上使用StringCaseChanger（将其更改为大写）而不是在犯罪数据上使用（将其更改为首字母大写）？

因为大写更容易匹配，并且没有坏数据的风险。如果在原始数据中错误地将块标记为W Georgia ST，则首字母大小写匹配将失败。不会发生大写转换。

另外，请确保不要将StringCaseChanger（更改属性值）与BulkAttributeRenamer（更改属性名称）混淆！

最佳实践

如果一个工作空间运行到完成并产生了您想要的输出，那么它不会是糟糕的，不是吗?嗯,是的。仅仅将一个有效的工作空间组合在一起是不够的; 以高效和可扩展的方式使用FME也至关重要。

什么是最佳实践？

一般而言，最佳实践意味着做某事的最佳方式; 换句话说，有效且高效地执行任务。

尽管有“最好”这个词，但我们并不认为这里的想法会满足每一个需要和场合。我听过的这个概念的最佳描述 - 以及一个适合这里的概念 - 是：

> “根据过去的经验和分析，在这种情况下考虑一个非常好的做法”

为何使用最佳实践？

FME的最佳实践可以帮助用户.....

- 创建自我记录的风格良好的工作空间
- 创建在正确位置使用正确功能的工作空间
- 创建高性能工作空间
- 以最有效的方式使用FME Workbench
- 如果工作空间不按预期方式工作，请对其进行调试
- 在基于项目的环境中使用FME

Workbench博士说.....
我是Workbench医生，获得了在FME工作空间上执业的许可。 当我在一组别人的工作空间里接受手术时，我以艰难的方式学习了最佳实践。他们的组织非常糟糕，整个行动花了我三倍的时间，破坏了我下午打高尔夫的计划！

在本章中，我们将介绍最佳实践的三个不同方面。

样式

本节是工作空间首选设计的指南。正确的样式使工作空间更容易解释，特别是在长期运行中作者可能在一段时间的不活跃之后重新使用它。

方法

本节介绍哪些技术可以有效地使用Workbench及其组件 - 哪些不能！使用Workbench以正确的方式提供更高效和高效的体验。

调试

本节介绍帮助识别和修复转换问题的工具和方法。

分析师女士说.....
FME最佳实践非常类似于软件开发最佳实践。例如，计算机程序员在其代码中添加注释以解释其操作，您应该在FME中执行相同的操作。

样式

> “一个外观漂亮，组织良好的工作空间让客户感觉你已经完成了高质量的工作。”

样式可能是FME最佳实践中最明显的组成部分。您可以一目了然地了解工作空间的样式是否合格，何时不是。正如上面的引用所暗示的那样，精心设计的工作空间展示了能力

但样式不仅仅是外观; 正确设计的工作空间提供了许多好处，因为它将来会进一步开发和编辑。

FME工作空间样式指南

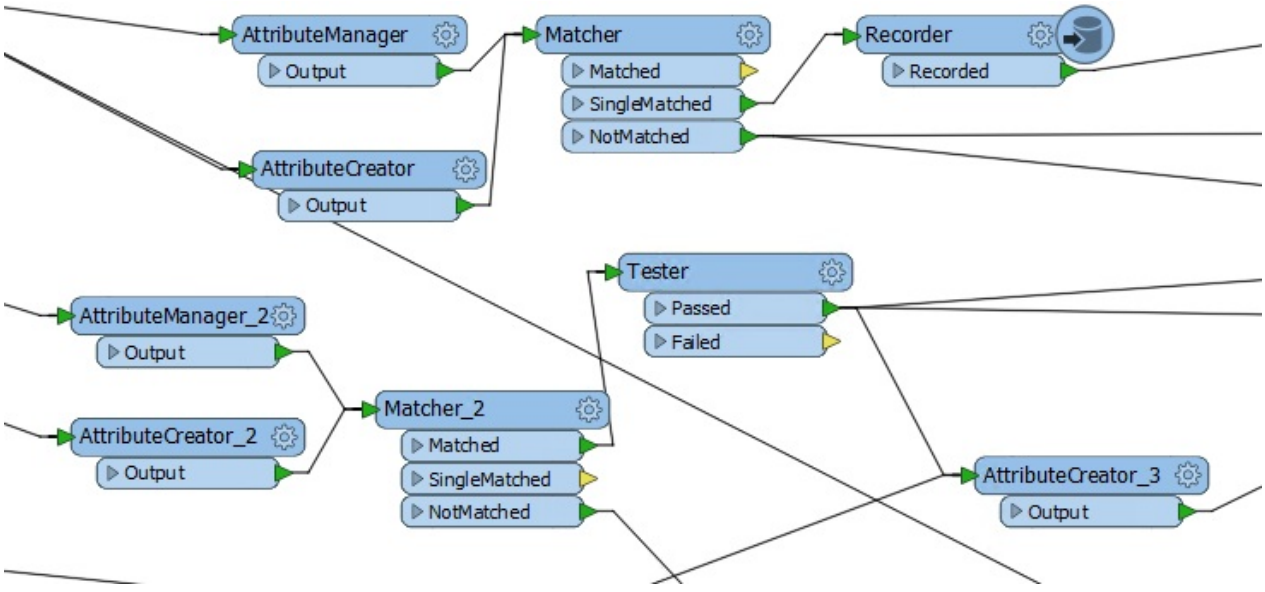
良好的设计样式使其更易于导航和理解现有工作空间。当工作空间可能需要由其他用户编辑时，或者您打算在以后自行编辑时，这一点很重要。

具体来说，一个好的样式可以帮助用户.....

- 明确定义工作空间的不同部分或组件
- 快速导航到指定的部分或特定的转换器
- 将工作空间传递给另一个用户进行编辑
- 使用更具说明性的标题重命名工作空间和内容

设计不佳的例子

你需要证明吗？那么，您是否希望被赋予编辑此工作空间的任务？你能说出这部分的内容，或者更重要的是 - 为什么？



Interop阿姨说.....

我是Interop阿姨。我写了一篇博客，帮助用户解决各种FME问题。 正如我常说的，大小无关紧要！您应该始终使用最佳实践，无论是小型工作空间还是培训练习，还是大型项目。养成这个习惯有助于你的小项目的可扩展性。 如果您从一开始就没有很好地设计工作空间，那么在您处理工作时，编辑将变得越来越难。

注释工作空间

注释是清晰易懂设计的关键方法。

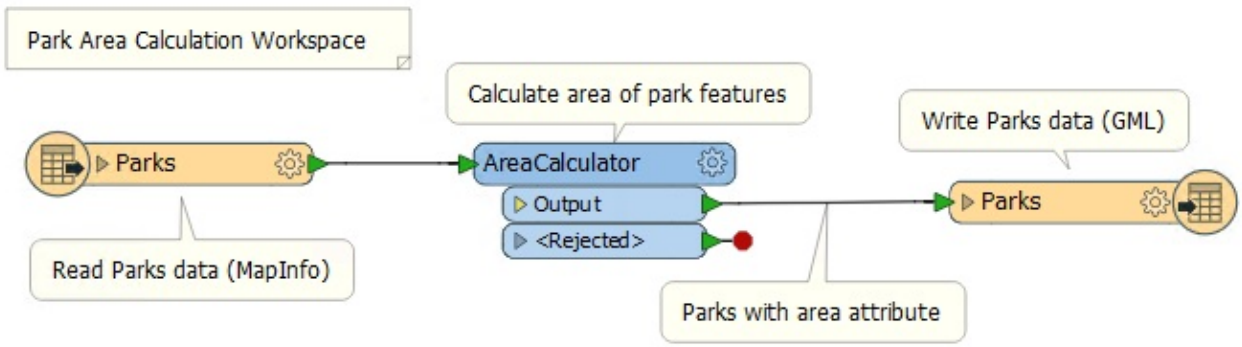
注释可以帮助其他用户理解转换中应该发生的事情，并在长时间间隔后返回工作空间时帮助创建者（从我这里获取这一点尤为重要！）

可以将两种不同类型的注释应用于工作空间。

用户注释

用户注释是用户创建的注释。它可以连接到工作空间对象（转换器或要素类型），可以连接到工作空间连接，也可以在工作空间内自由浮动。

[

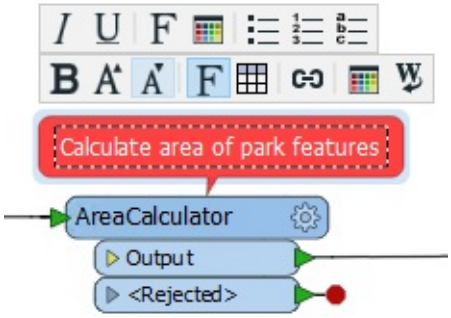


]

要创建附加的用户注释，请右键单击工作空间对象并选择“添加注释”，或在选择对象时使用快捷键Ctrl + K.

要创建浮动用户注释，请右键单击画布并选择“插入注释”，或在未选择任何内容时按Ctrl + K.

放置注释时，您可以更改字体样式，字体大小和背景颜色; 另外，您还可以添加超链接，项目符号和表格。

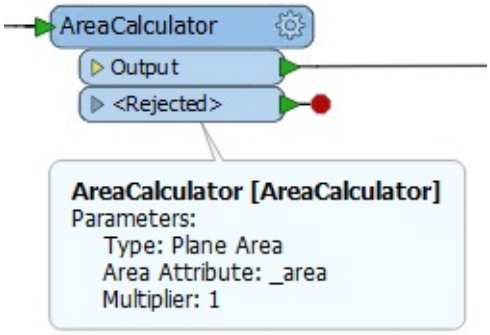


新知识
在FME2018中，用户注释现在可以直接附加到书签。

摘要注释

摘要注释是FME生成的注释，提供有关工作空间内任何对象的信息。此项可以是源或目标要素类型，也可以是转换器。

摘要注释始终显示为蓝色，以便将其与其他注释区分开来。它始终与其相关的项目相关联，并且无法分离。

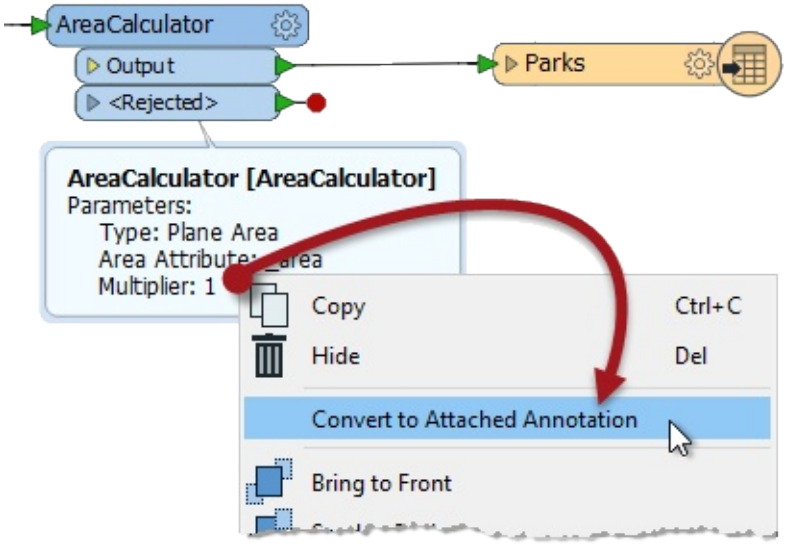


摘要注释的优点在于它会自动更新以响应更改。这使得它很快就可以快速检查转换器参数（或读/写模块模式）。它在通过向导设置参数并且更难检查的情况下特别有用（例如，SchemaMapper或FMEServerJobSubmitter转换器）。

技巧

一个好主意是使用总结注释显示哪些行为正在发生; 但是使用用户注释来阐明为什么要执行某个操作。

您可以使用此上下文菜单选项将摘要注释转换为用户（附加）注释：



这允许您从摘要注释中提取信息，但可以像编辑用户注释一样对其进行编辑。请注意，转换后的摘要注释不再自动更新！

书签

书签就像它的真实世界同名一样，是一种放下标记以便于访问的方法。

通过FME，书签覆盖了工作空间的一个区域，该区域通常执行特定的任务，因此用户可以从更广泛的转换器组中选择它并相对容易地移动到它。

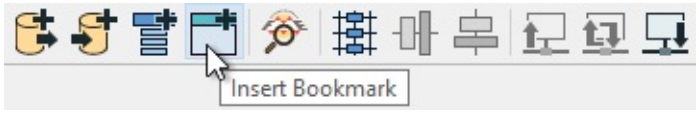
为什么要使用书签？

书签在样式良好的工作空间中起着重要作用，原因有很多，包括这些原因。

- 设计：作为细分工作空间和管理这些部分的一种方式
- 访问：作为快速访问工作空间特定部分的标记
- 编辑：作为一次移动转换器组的手段
- 性能：作为缓存数据时提高工作空间性能的一种方法

添加书签

要添加书签，请单击工具栏上的“书签”图标。

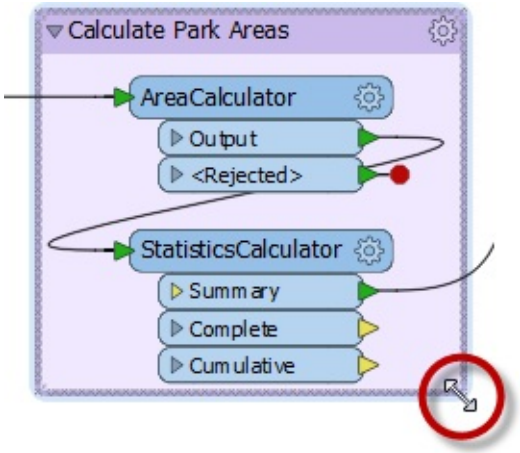


传统书签只标记书中的单个页面，而FME书签可以覆盖画布的大部分区域。通过应用多个书签，可以将单个工作空间划分为不同的部分。

技巧
如果在创建书签时选择了工作空间画布上的任何对象，则会自动扩展书签以包括这些项目。

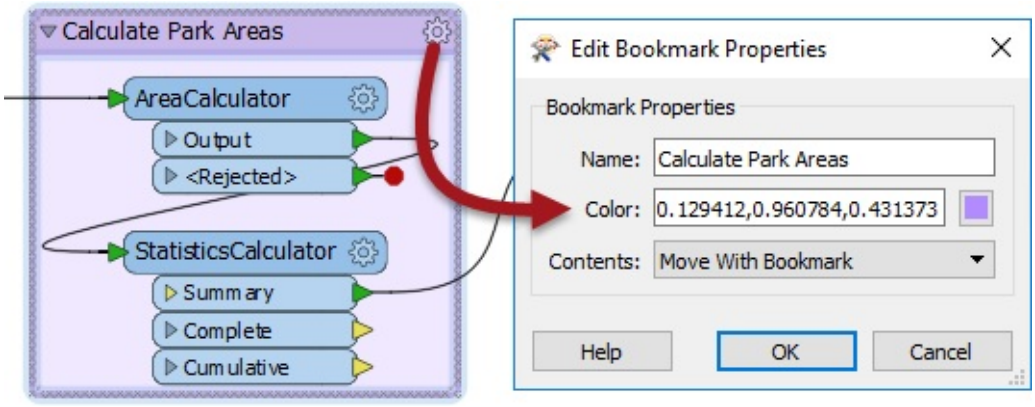
调整书签大小和编辑书签

要将书签大小悬停在角落或边缘上，然后拖动光标以更改书签大小或形状。



书签属性

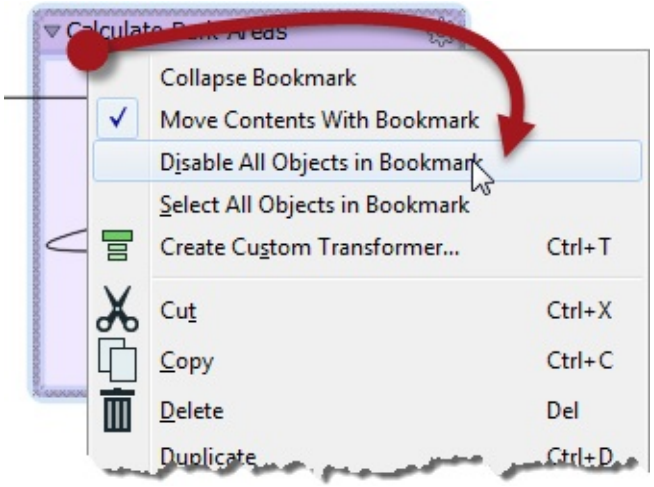
单击书签标题上的齿轮图标以打开书签属性对话框：



在这里，您可以更改书签的名称和颜色，并决定内容是否随之移动（稍后会详细介绍）。

技巧

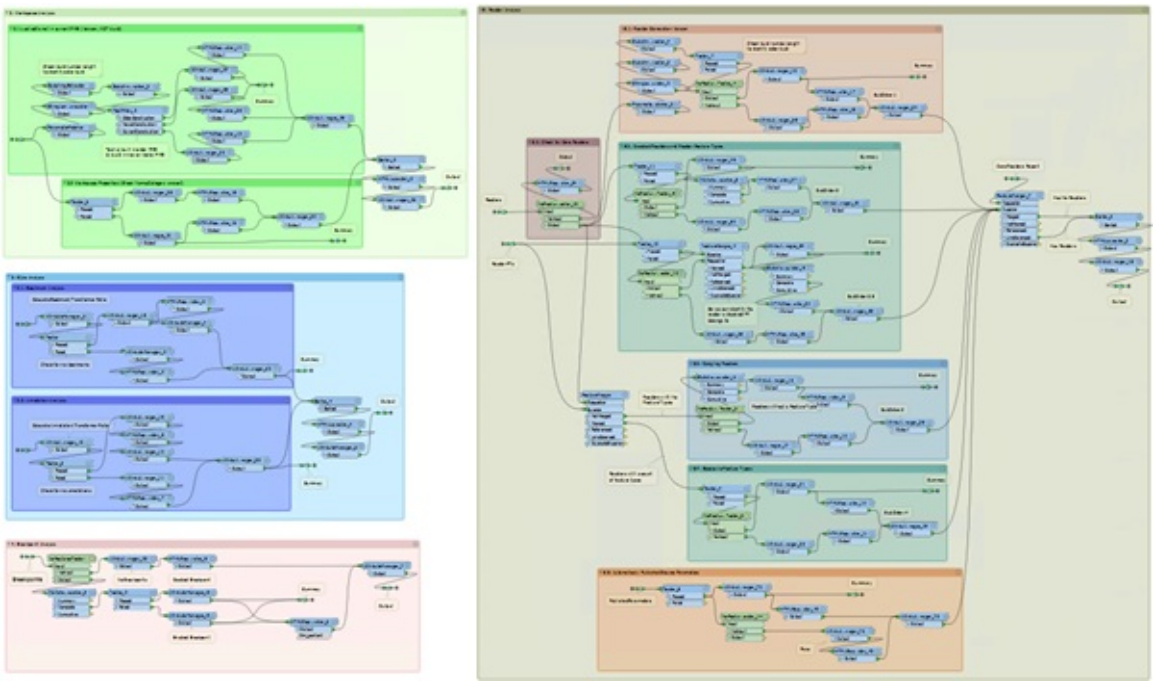
书签的上下文（右键单击）菜单显示选项以选择书签中的所有对象，或禁用所有这些对象，使其对测试有用：



书签用于设计

书签是一种很好的方式，可以指示工作空间中的特定部分是用于特定目的。通过以这种方式细分工作空间，布局通常更容易理解。

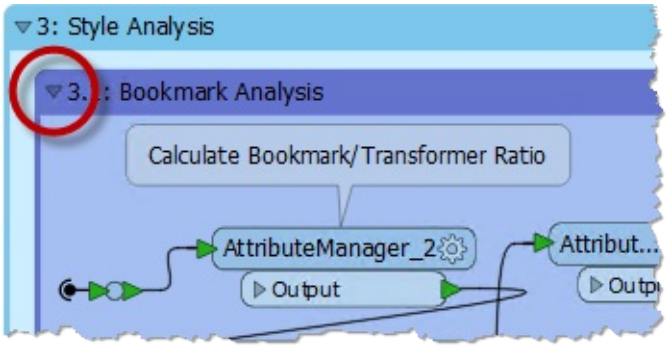
正如一位用户所说，书签就像工作区的段落！



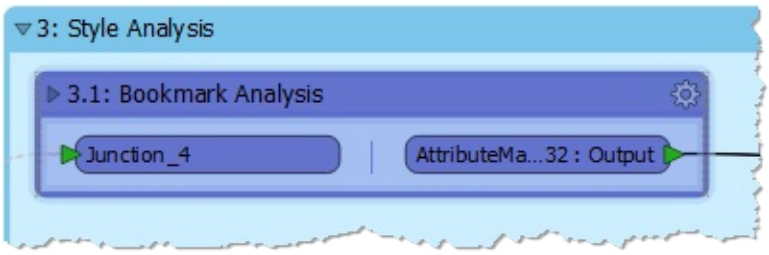
上述工作空间说明了如何使用书签标记工作空间的不同部分。如您所见，允许通过将一个书签嵌套在另一个书签中来进一步细分书签。

可折叠书签

每个书签的左上角都有一个小图标，可以折叠它：



折叠书签意味着它被压缩到单个转换器的大小，除了数据从哪里进入或退出书签之外，不显示任何内容：



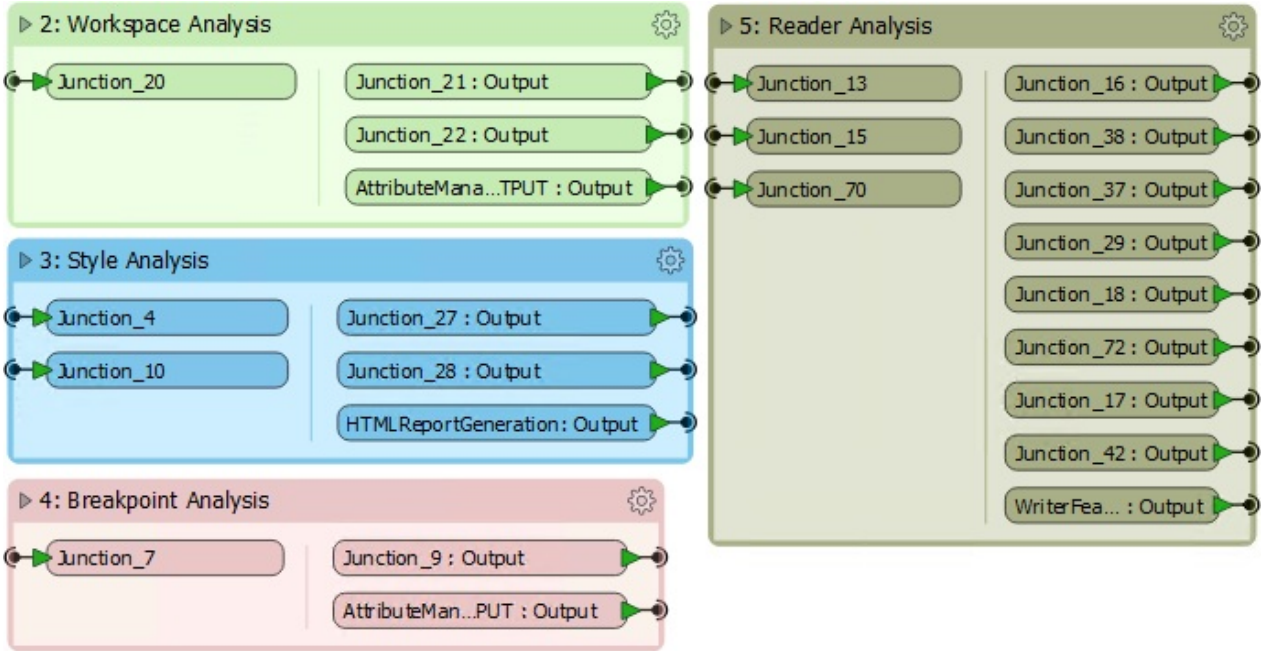
再次单击该图标会将书签重新打开为其先前的大小。

新东西

可折叠书签是*FME2018*的全新功能。

此功能允许在较小的区域中呈现较大部分的工作空间，并且仅在需要编辑时才打开。

例如，上面显示的工作空间部分可能会缩减为：



重新打开折叠的书签会调整工作空间的布局，将其他转换器或书签移开，使其内容显示时不会出现重叠。重新关闭书签会导致相反的情况发生。

例如，在上面的屏幕截图中，如果书签3（样式）被展开，则书签四和五被移动到一侧以适应它。当书签3再次折叠时，会发生相反的情况，以提供与以前相同的紧凑布局。

E. Dict先生（FME法律顾问）说...

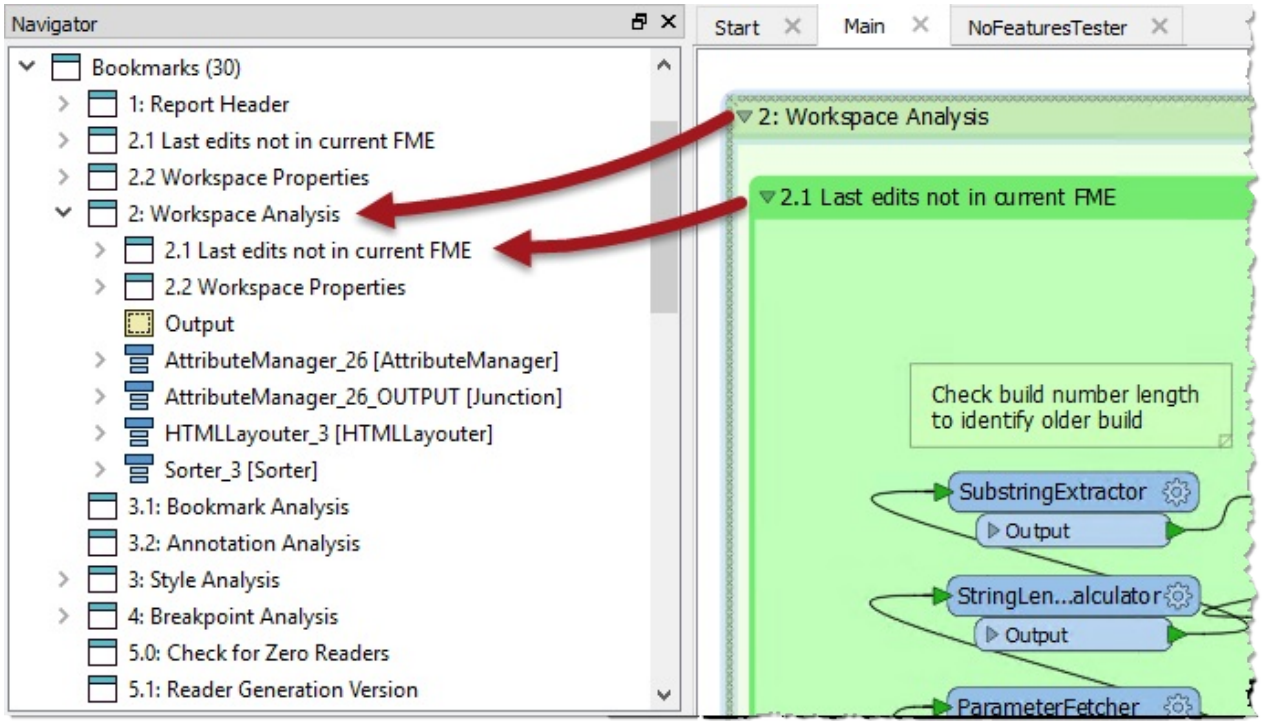
折叠书签上的输入和输出端口可以重命名，以帮助使进入和退出的数据更清晰：

这可以通过双击对象，按*F2*或使用上下文菜单上的重命名选项来实现。

书签用来访问

快速访问的书签

书签在Workbench Navigator窗口中列出。每个书签都被描述为一个文件夹，可以展开以显示其内容。它可能包括要素类型，转换器或其他 - 嵌套 - 书签：

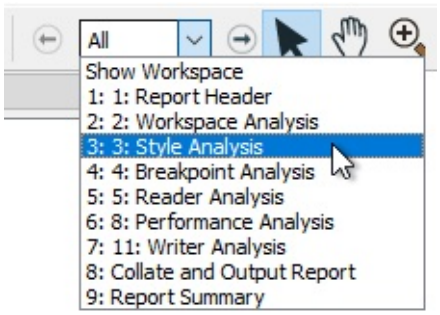


单击或双击导航器中的书签可选择该书签并将其显示在视图中。因此，当使用书签将工作空间划分为多个部分时，它们也可用于在该工作空间的不同部分之间导航。

通过这种方式，书签就像书中的章节标题！

工作空间演示

书签也可以显示在FME Workbench工具栏上：



除了作为快速访问书签的方法之外，该工具还可以用作FME Workbench中的“演示模式”。通过单击箭头按钮（或按键盘空格键），您可以使用动画从书签切换到书签，这在将工作空间显示为演示文稿的一部分时非常有用。

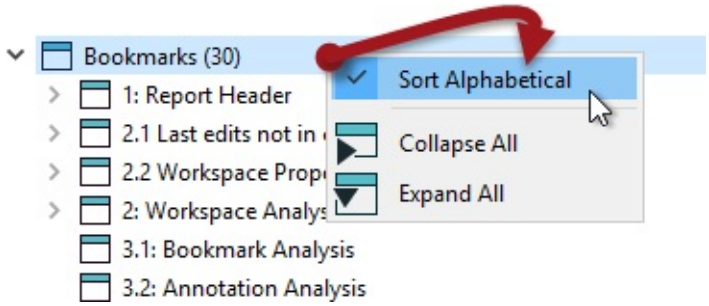
技巧

要访问所需的功能，请确保将其添加到工具栏中。您可以通过右键单击工具栏并使用自定义选项来完成此操作。

Workbench医生说...

该窗口中书签的顺序是按字母顺序排列的，可能并不总是与您希望呈现工作空间的顺序相同。

在这种情况下，右键单击“导航”窗口中的“书签”，然后将默认选项设置为“按字母顺序排序”。



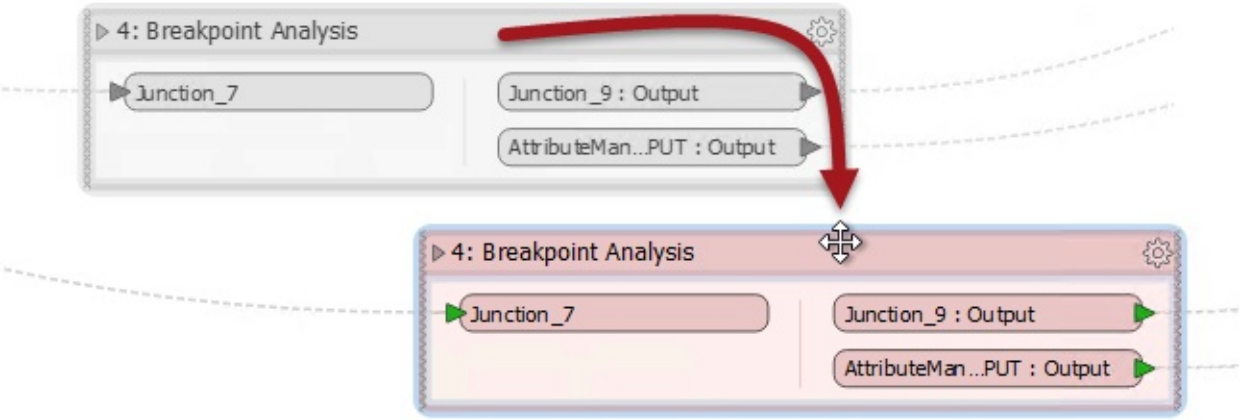
然后可以在“导航”窗口中上下拖动书签以提供正确的顺序。此外，书签属性对话框上的新选项允许您从书签导航器中排除特定书签。默认情况下会排除嵌套书签，因此必须将其打开才能包含在导航器中。

有关更多信息，请参阅安全软件博客上的 有关书签的文章。

书签用于编辑

书签定义包含许多对象的工作空间的一部分。编辑没有书签的工作空间时，通过选择一个或多个对象并将它们拖动到新位置来完成移动对象。

但是，当工作空间被书签分割时，只需将书签拖动到新位置即可移动对象。当对象位于书签内部时，它会像书签一样移动。



使用此技术，可以在工作空间画布周围移动大量对象，以创建更清晰的布局。无论书签是折叠还是展开，对象都会在书签内移动。

Workbench医生说...

请记住，其中一个书签属性标记为“内容”，其值为“使用书签移动”或“独立移动”。显然，前一个值会导致对象随转换器移动；后一个值会导致对象在移动书签时保持在适当位置。

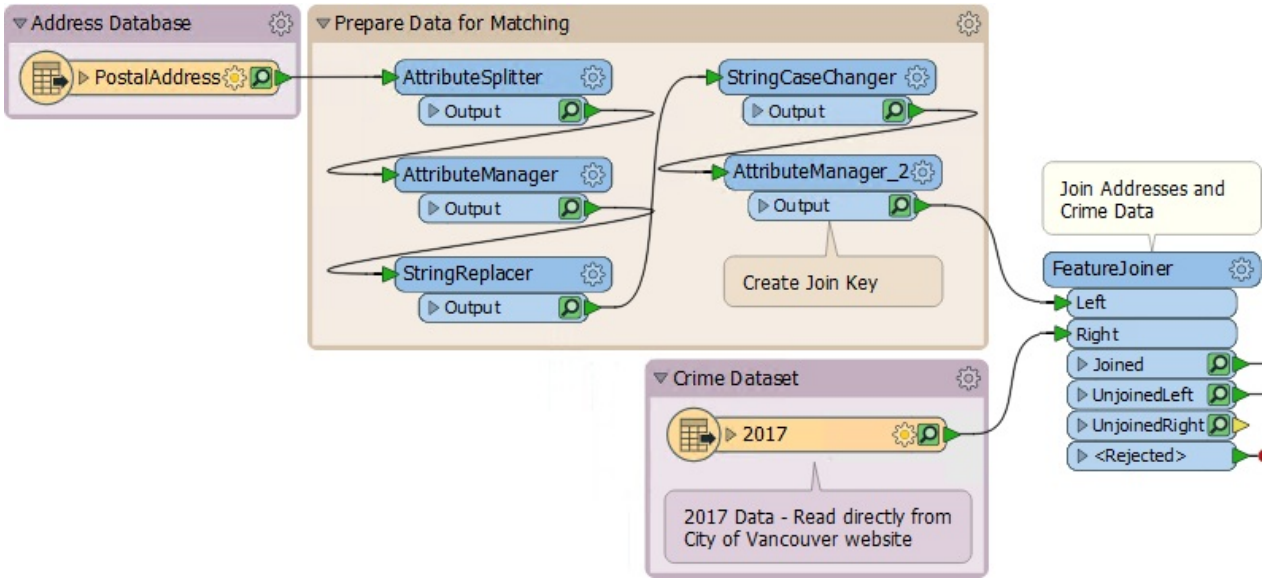
Vector小姐说...

这是关于书签的问题。

以下哪项不是创建书签的方法？
1.单击工具栏上的“插入书签”按钮
2.选择一个转换器，右键单击，选择“创建书签”
3.选择多个转换器，右键单击，选择“创建书签”
4.使用Ctrl + B快捷键

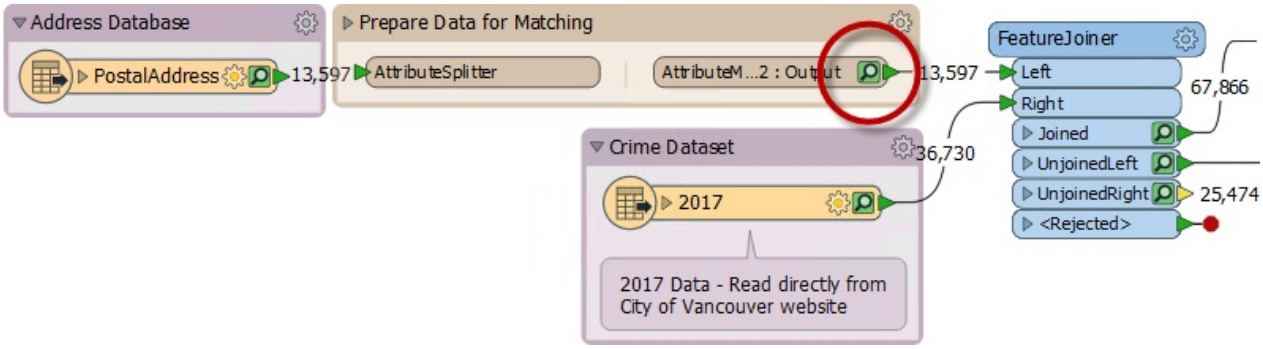
书签用于性能

在启用数据缓存的情况下运行工作空间时，会在每个转换器上缓存要素。可以想象，在较大的工作空间中，这会导致大量数据被缓存，有时不必要：

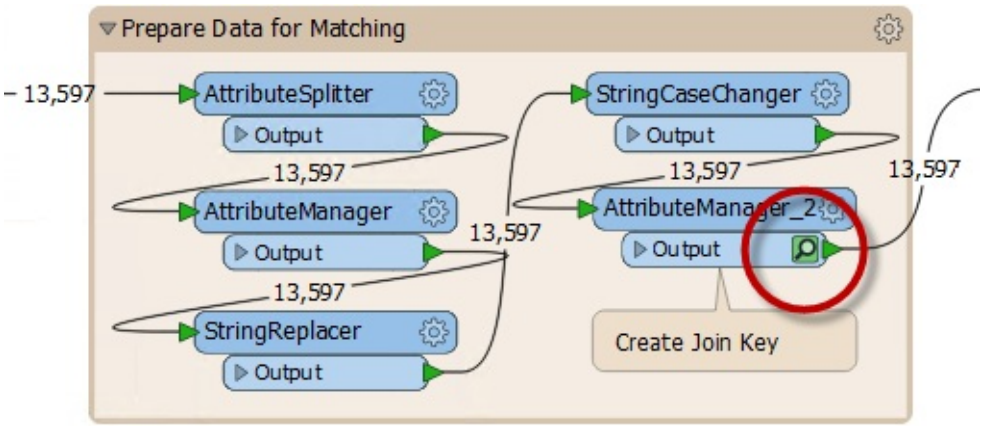


请注意，在上面的屏幕截图中，正在缓存“准备匹配数据”书签中的每个转换器。

但是，当折叠书签时，缓存仅发生在书签输出对象上：



此特性意味着仅为书签中的最终转换器缓存数据，从而节省大量时间和资源：



Workbench医生说...

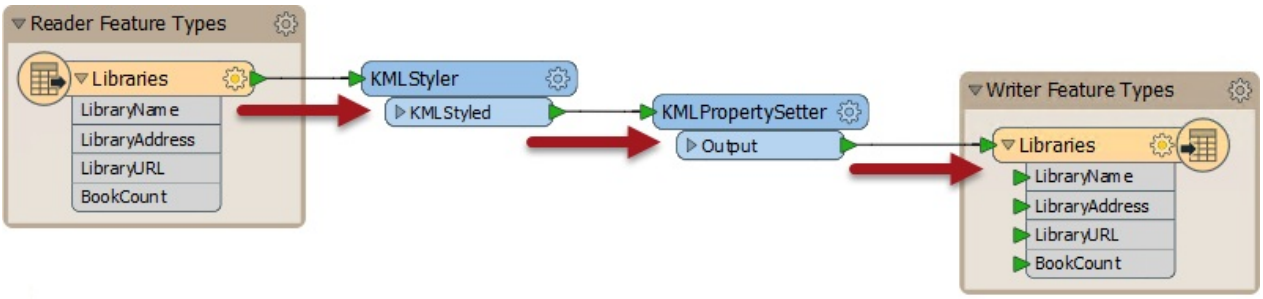
显然，无论您的书签是否已折叠，您都不希望在启用缓存时将工作空间投入生产。仅建议将此技术用于工作空间创建的设计，创作和测试阶段。

对象布局

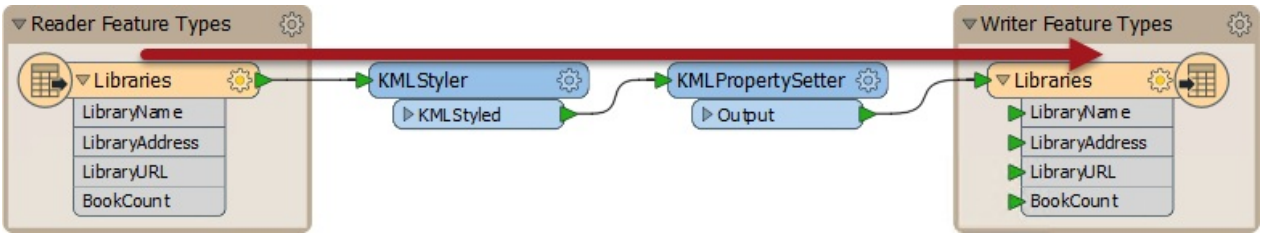
工作空间对象的定位以及连接它们时的注意力可以使设计不良的工作空间与视觉上有吸引力且高效的工作空间之间产生差异。

对象布局

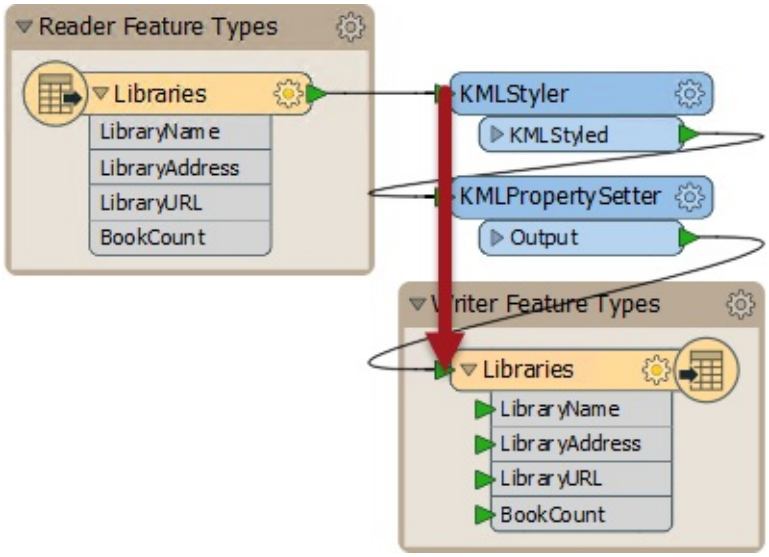
布局方法因用户而异。有些用户喜欢排列对象，以便所有连接都是水平的：



其他人更喜欢水平对齐的物体顶部，有角度的连接：



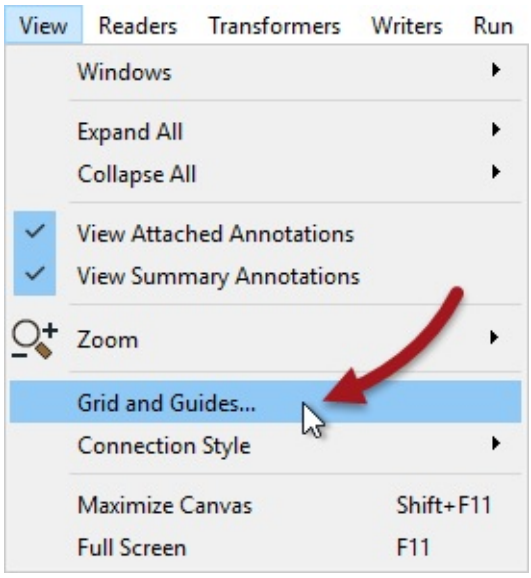
有些人喜欢垂直对齐对象边缘：



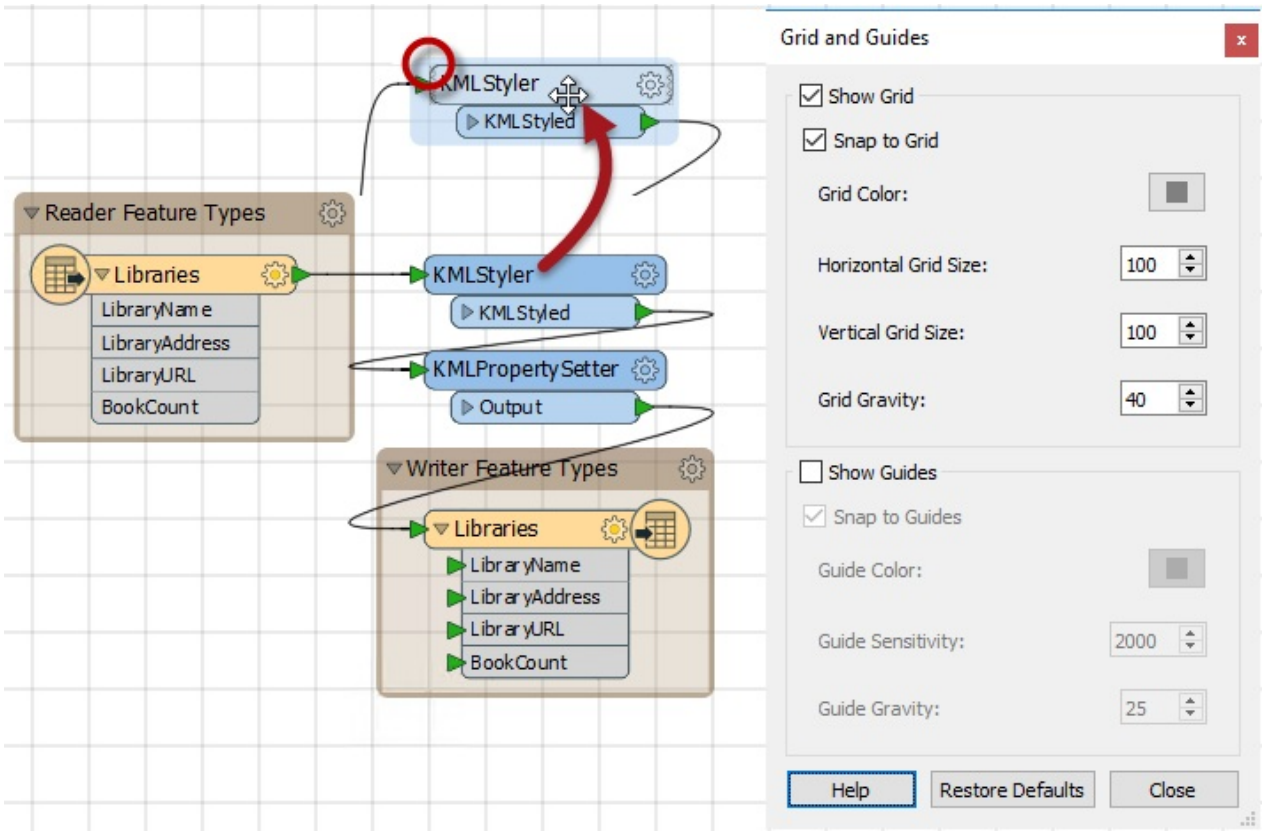
使用的样式更多的是个人偏好而不是明确的规则，但重要的是一致性。没有明显布局样式或不一致的工作空间不会激发对作者能力的信心！

格网和参考

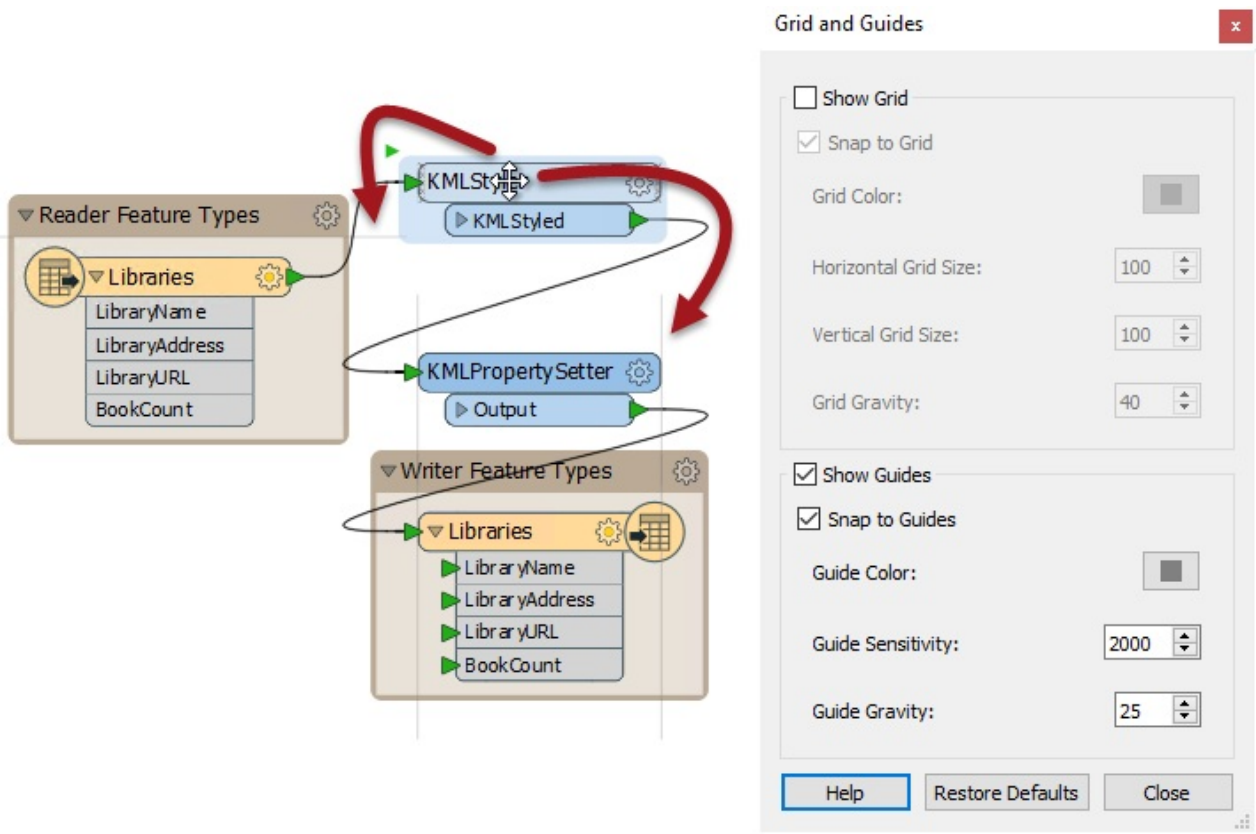
格网和指南是一种帮助整齐和整齐地对齐工作空间对象的工具。可以通过Workbench菜单栏上的View> Grid和Guides访问此功能。



显示格网会导致在Workbench画布上显示线条格网。“对齐格网”会导致所有对象（例如突出显示的摘要注释）在移动时捕捉到格网线的交叉点。通过这种方式，物体可以更容易排列。



“显示参考”会导致在移动对象时在Workbench画布上显示参考线，并将其大致排列到另一个画布对象。“捕捉到参考”允许将对象捕捉到突出显示的参考线上。

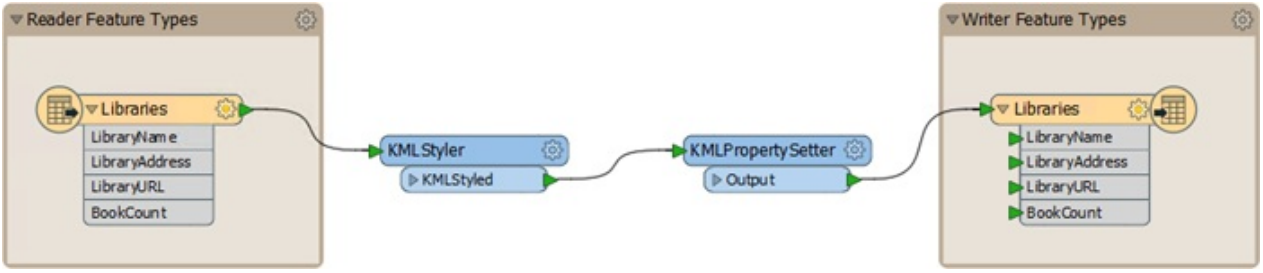


这两个工具使得工作空间对象以令人愉悦的方式对齐变得非常简单。

自动布局

自动布局工具显示在FME Workbench的工具栏上：

单击工具栏按钮将布局所有工作空间或仅显示当前选定的对象：



如您所见，自动布局倾向于使用水平图案，对象的顶部对齐。因此，在使用此工具时，最好一次选择转换器组，而不是尝试在单个操作中布置整个工作空间。

Workbench医生说.....
通常，自动布局算法是正常的...但它仍然无法与花费时间和精力来手动组织对象布局进行比较。

连接样式

值得注意的是，对象定位只是良好布局的一部分。另一个关键部分是连接样式。

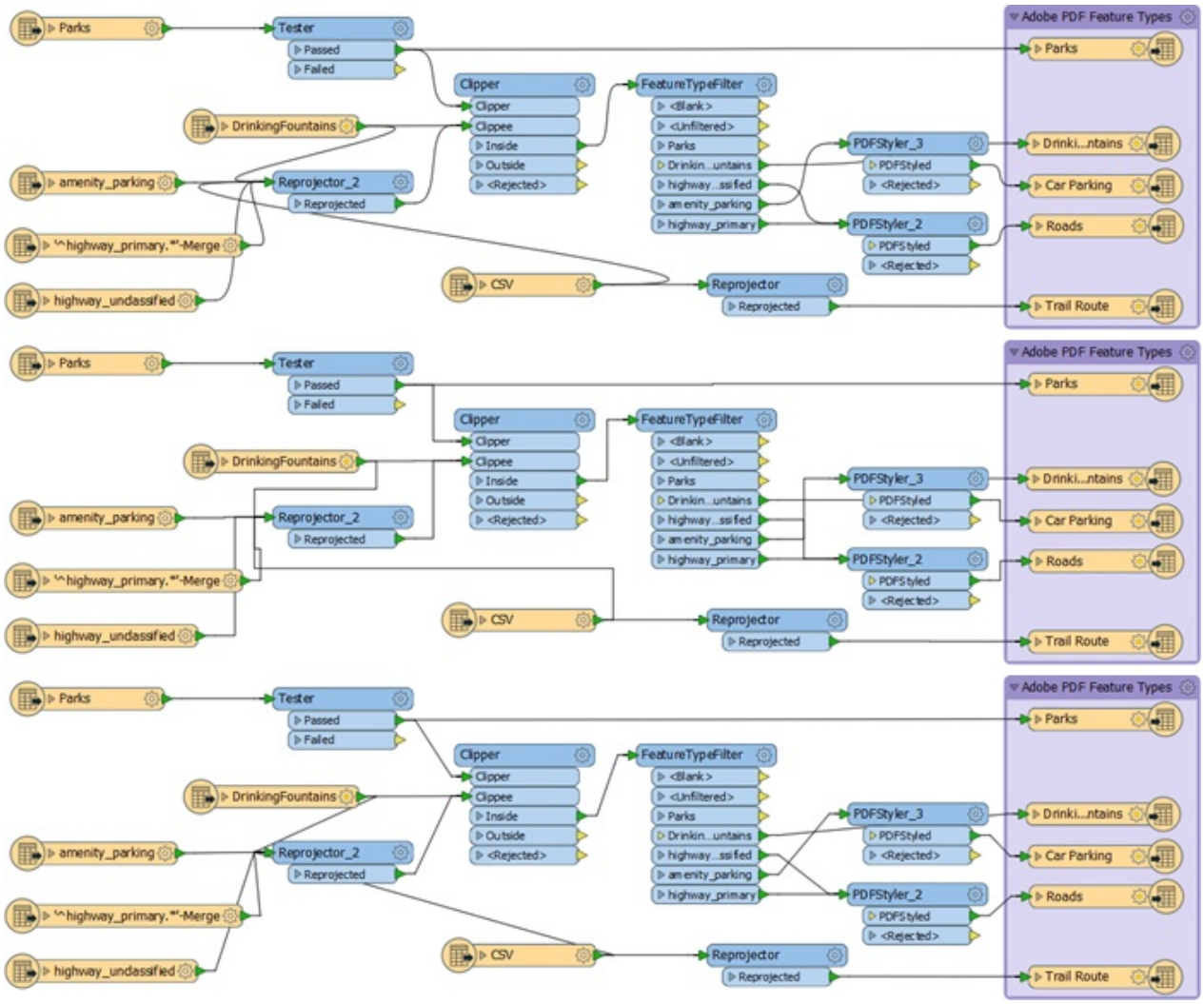
与工作空间对象的定位一样，连接它们时的注意力可以使设计不良的工作空间与视觉上有吸引力且高效的工作空间之间产生差异。

连接样式

连接是工作空间画布上对象之间的线。您可以在Workbench中创建三种不同的连接样式：

- 弯曲：具有曲线连接的默认样式
- 方形：通过方形连接唤起曼哈顿天际线的风格
- 直的：原始连接方式; 两个物体之间的直线

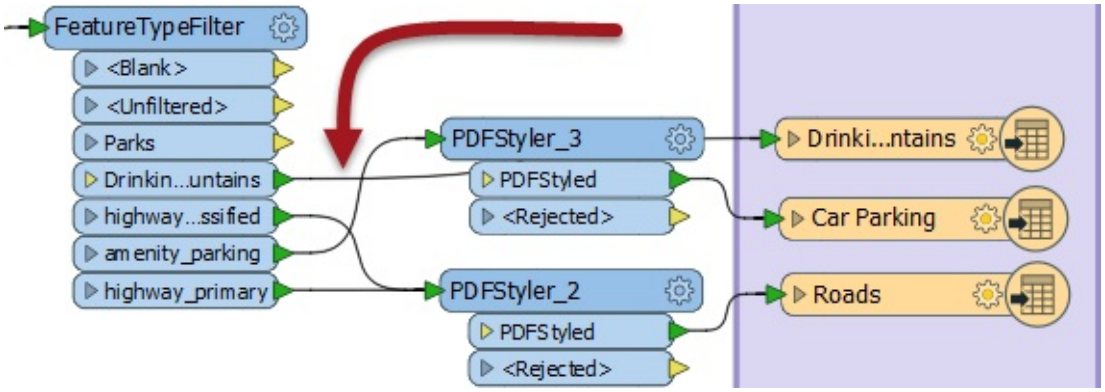
您可以使用“视图”菜单，“FME选项”菜单或快捷键Ctrl + Shift + C在样式之间切换。此图显示了三种样式的比较：



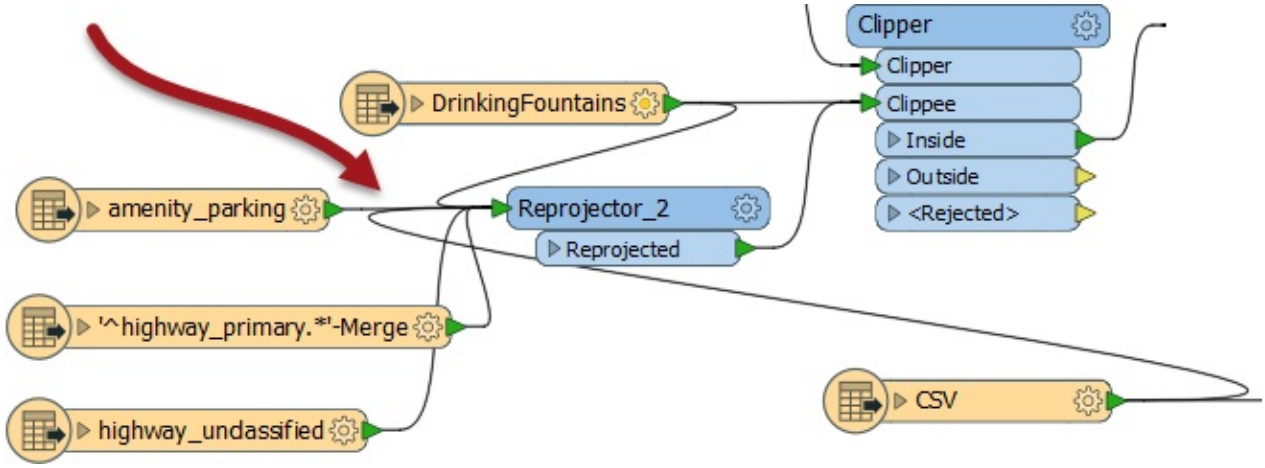
再一次，关于使用哪种样式没有正确或错误的选择; 这更像是个人偏好。但是，对象布局和连接方式是相关的; 最好的FME作者将根据用于避免重叠连接等问题的连接方式改变对象的位置。

重叠连接

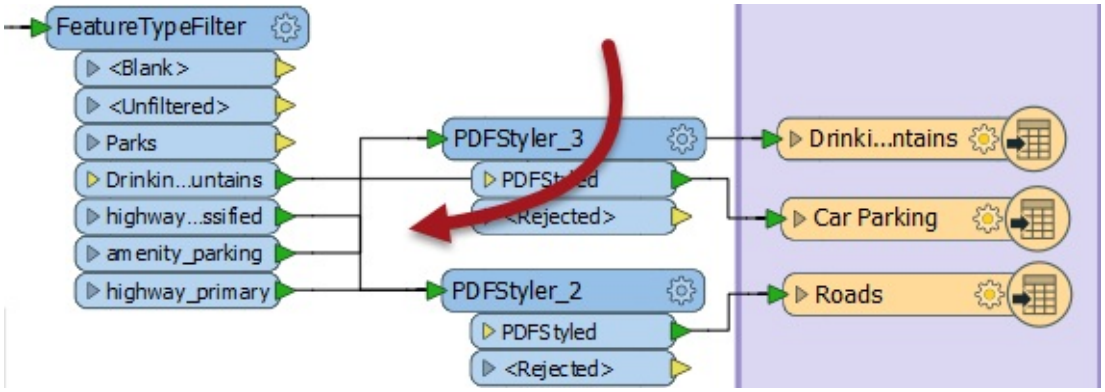
工作空间设计最明显的缺点之一是具有相互交叉的连接，例如：



当连接与画布上的另一个连接或另一个对象重叠时，连接的可见性和意图总是会受到损害。但是，连接方式的选择会影响重叠发生的可能性。例如，弯曲的连接往往比直的连接更多：



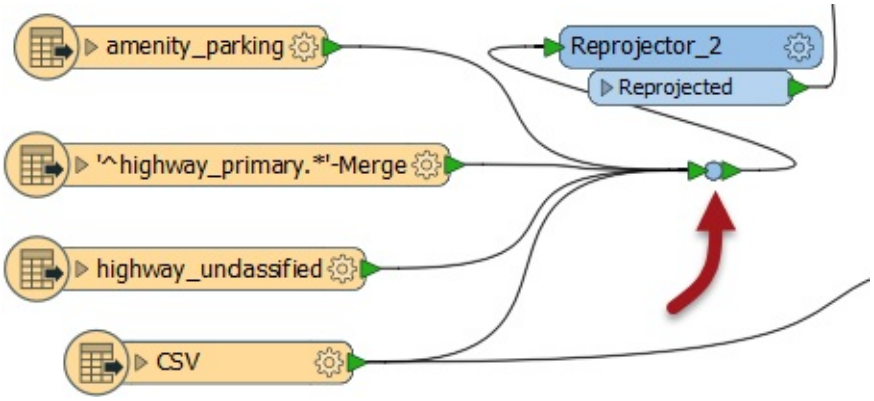
.....方形连接有时会以难以理解的方式交叉：



因为当您切换连接样式时会出现这些问题，所以选择特定的连接样式和布局技术并坚持下去是明智的。例如，在弯曲连接工作空间中，转换器可以更宽的间隔以避免重叠。

Junctions

FME Workbench中有一个转换器，旨在用于增强对象和连接的布局。此转换器叫做**Junction**。

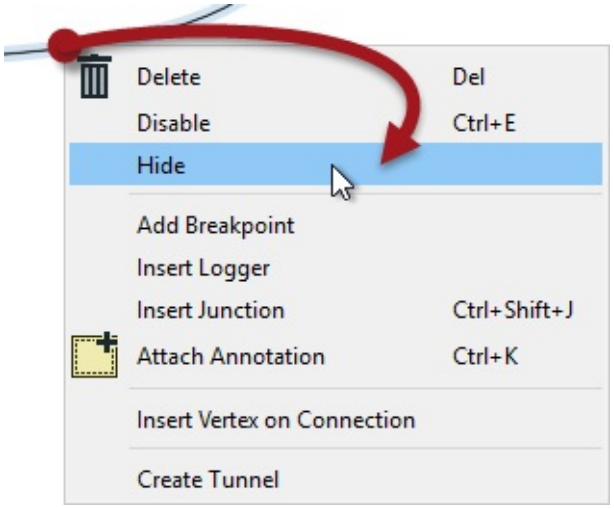


这个转换器是一个类似节点的小对象，它不对数据执行任何功能，而是用于整理工作空间内的连接 - 如上面的屏幕截图所示。这种特性使其成为最佳实践的绝佳工具。

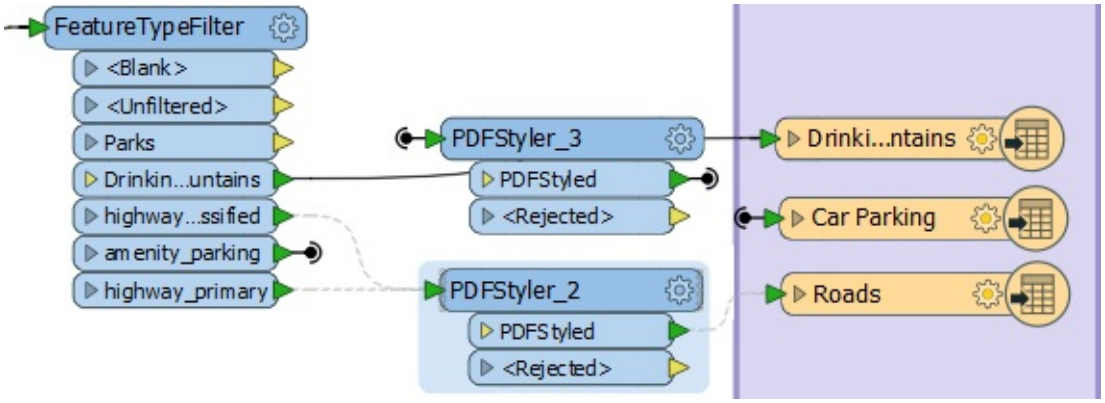
与任何其他转换器一样，结点可以连接到Inspector或Logger，并且可以附加注释对象。它还适用于“快速添加”，“拖动/连接”功能和“要素缓存”。

隐藏的连接和隧道

隐藏连接的能力对于避免重叠特别有用。要隐藏连接，请右键单击它并选择隐藏选项：

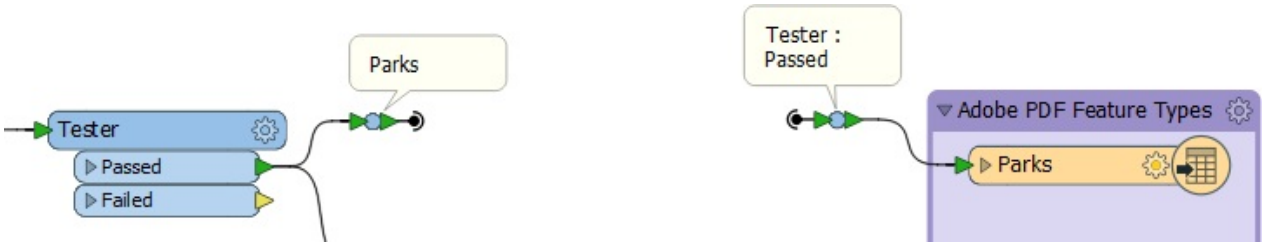


隐藏连接由“发射器”图标表示，或者当选择连接一端的对象时由灰色虚线表示：



此处必须选择对象（转换器或要素类型）才能使连接可见。

另一个可用选项是“创建隧道”。此选择会在每端添加带注释的结转换器时创建隐藏连接：



隧道使隐藏连接稍微明显，并允许在每一端进行注释。

重新审视隐藏连接

要查看隐藏的连接，请单击任一端的对象。连接突出显示为灰色虚线。

要将连接返回到视图，请右键单击与其连接的对象，然后选择“显示连接”。

有关隧道和连接点的更多信息，请参阅此博客文章。

练习：FME风格指南

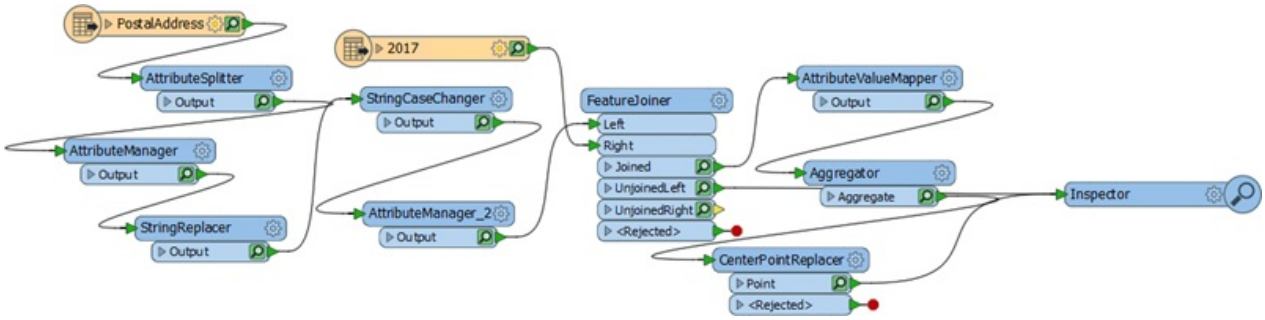
练习1	应用样式指南
数据	地址（Esri地理数据库） 犯罪数据（CSV - 逗号分隔值） 公园（MapInfo TAB）
总体目标	参与温哥华可行走性项目
演示	样式最佳实践
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\BestPractice-Ex1-Begin.fmwt
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\BestPractice-Ex1-Complete.fmwt

您刚被分配到一个项目来计算温哥华市每个地址的“步行性”。步行性衡量步行访问当地设施的难易程度。它将包括到最近公园的距离，一个区域内的犯罪数量以及其他类似指标的度量。

该项目的工作空间已经启动，犯罪信息已连接到每个地址，但作者不知道FME样式指南，因此，为了继续项目，必须在现有内容中添加一定量的样式。

1）启动**Workbench**
启动FME Workbench。

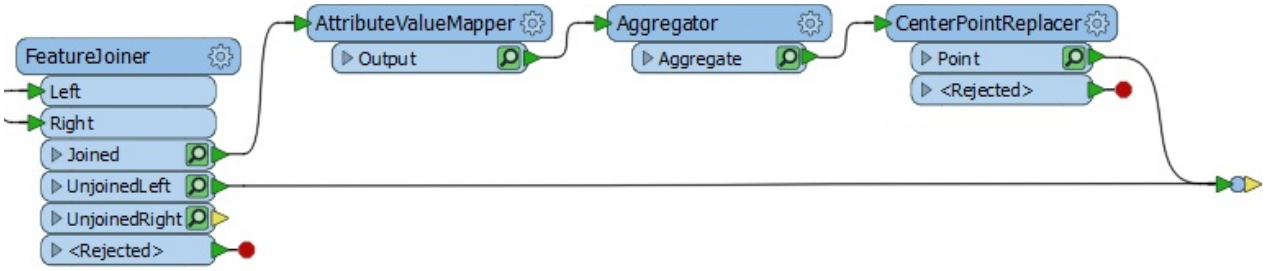
请注意，工作空间文件是一个模板（文件扩展名为.fmwt）。这很好，因为这意味着项目可能存在现有的缓存。确保已启用“使用要素缓存运行”并从C：\ FMEData2018 \ Workspaces \ DesktopBasic \ BestPractice-Ex1-Begin.fmwt打开工作空间模板。



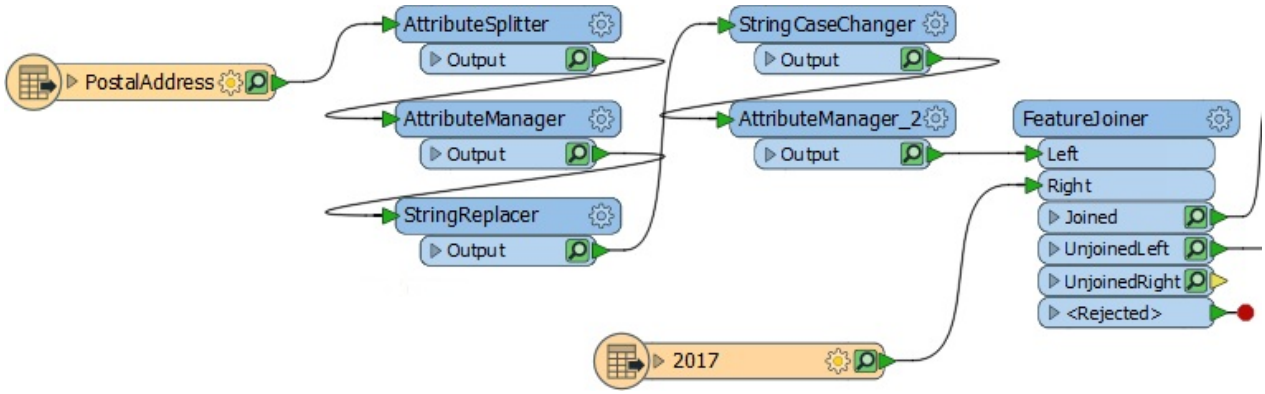
此工作空间需要以书签和注释的形式应用某种样式。

2）重新排列转换器
首先不需要Inspector转换器 - 我们可以更轻松地检查缓存 - 并且可以删除。但是，因为它有两个连接，我们想要跟踪它们是什么，一旦删除，用一个Junction转换器替换它。

然后清理它周围的转换器，以便没有重叠的连接：



现在重新排列工作空间的第一部分，避免重叠连接，并查找可能逻辑上分组在书签中的转换器：



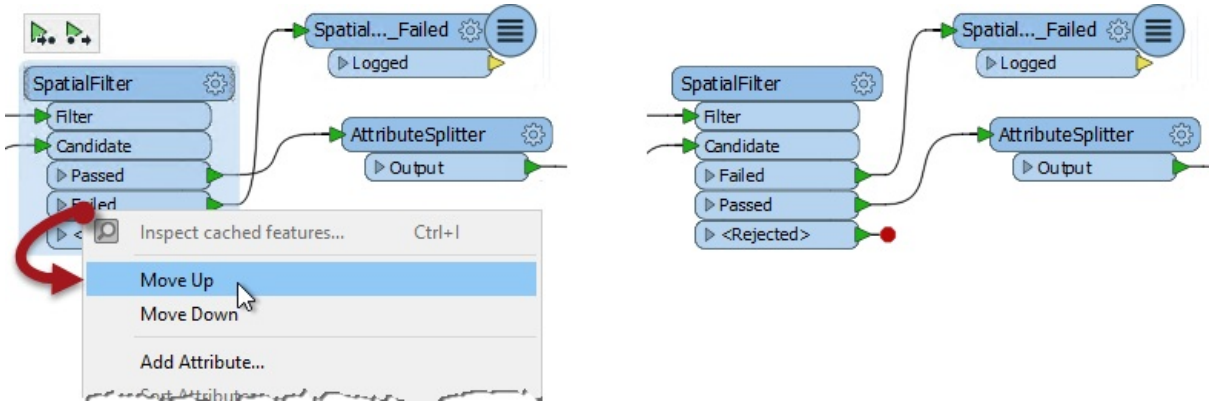
3）添加样式

重新排列转换器后，现在在适当的位置添加书签和注释。此步骤需要对转换器进行一些检查，以了解每个转换器的作用。但是你应该能够看到至少两组可以放入书签的转换器：

不要忘记使用注释和书签标题，以便工作空间的未来用户能够一目了然地了解工作空间应该做什么。

技巧

通过重新排列转换器输出端口的顺序，也可以清除重叠连接：



4）运行工作空间

如果需要，可以在新放置的Junction转换器上使用Run to This运行工作空间。然后，您应该能够单击Junction并按Ctrl + I来检查转换阶段的数据。

需要注意的一点是，CSV（犯罪）读模块直接从温哥华市网站上的Excel电子表格中读取数据。因此，保留缓存数据并且不必再次读取它是很重要的。

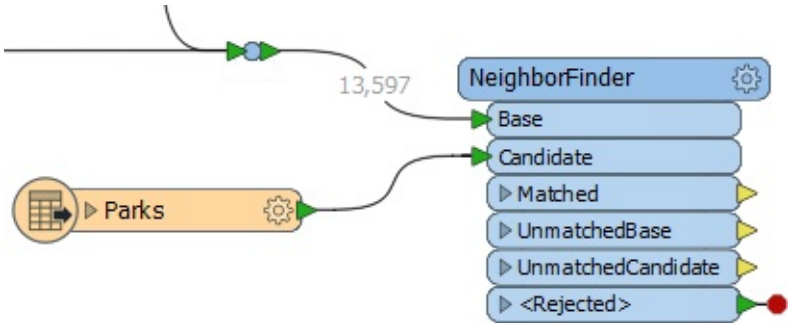
5）添加公园数据

现在让我们通过处理一些公园数据来继续项目。添加具有以下参数的读模块：

读模块格式	**MapInfo TAB（MITAB）**
读模块数据集	C：\ FMEData2018 \ DATA \Parks\ Parks.tab

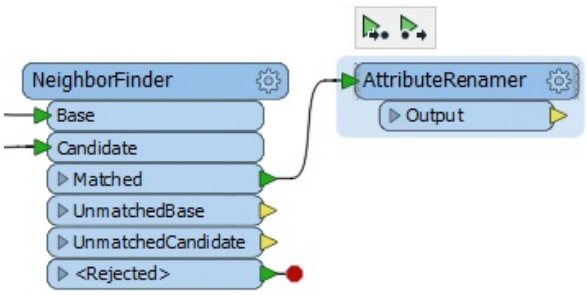
6）添加NeighborFinder
添加一个NeighborFinder转换器，我们可以用它来确定每个地址最近的停放地点，以及距离它有多远。

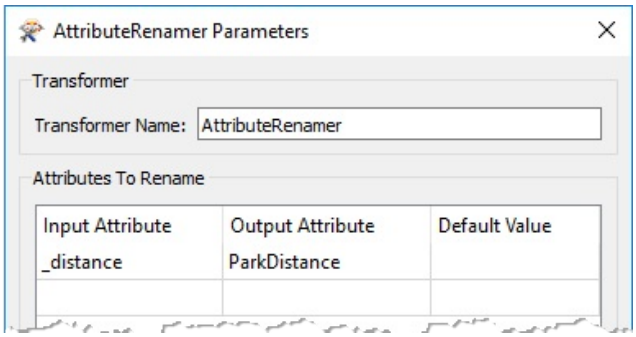
将Junction转换器连接到NeighborFinder:Base输入端口和Parks要素类型到NeighborFinder:Candidate输入端口：



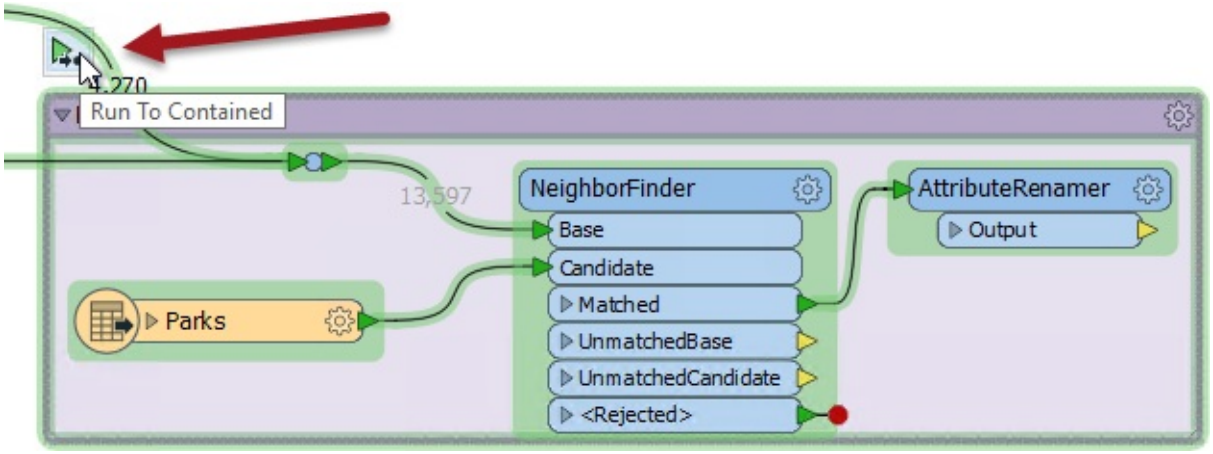
检查参数。我们只想找到一个邻居，没有最大距离。

7）添加AttributeRenamer
添加AttributeRenamer转换器以将属性*_distance*重命名为*ParkDistance*：



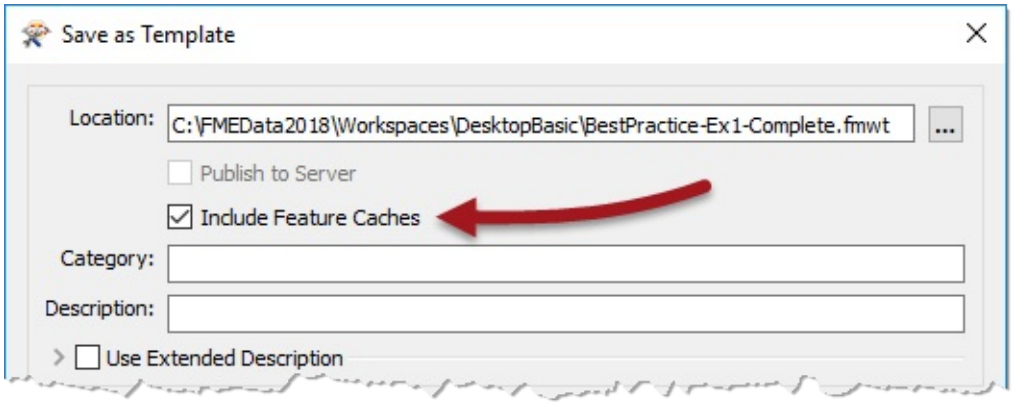


在处理公园的工作空间部分周围添加书签。通过选择书签并选择Run to Contained来运行工作空间的该部分：



检查输出以确保输出正确。

8）另存为模板
由于此项目可能会在以后继续，因此请保存工作空间，但使用“文件”>“另存为模板”。出现提示时，请确保选中“包括要素高速缓存”选项：



现在，如果我们稍后再回到这个项目，我们可以重新打开模板并准备好所有缓存数据。

恭喜
通过完成本练习，您已学会如何： • 将转换器重新排列为逻辑布局，将执行单个任务的那些组合在一起 • 使用注释来阐明工作空间中发生的过程 • 使用书签将单个工作空间转换为已定义的部分 • 避免糟糕的设计选择，如重叠连接 • 在工作空间中只运行一个书签 • 将工作空间另存为模板，包括缓存

方法伦

> “糟糕的方法会影响样式。糟糕的样式可能表明方法不佳。”

方法论是最佳实践的一个不太明显的组成部分而不是样式。它以两种主要方式表现出来：

- 维护
- 性能

在这两种情况下，实施一个好项目的第一步是了解哪种方法很差，以及如何避免它。

维护

维护是更新工作空间的能力，并将其扩展到更大的解决方案。

维护涉及设计模式 的概念。设计模式是执行特定任务的转换器的最佳实践。在维护或缩放工作空间时，糟糕的设计模式会导致复杂化。

通常可以以工作空间的方式检测到糟糕的设计; 因此，如果工作空间正确使用书签和注释，但仍然看起来不太好，则可能表明转换器模式存在缺陷。

性能

性能是使用最少资源执行转换的工作空间的效率。

性能对大多数用户来说很重要，但难以量化。如果工作空间在足够的时间内运行完成，则通常会忽略性能。当尝试增加项目规模以处理更多数据时，性能上的任何缺陷都会表现出来。

令人惊讶的是，糟糕的性能还体现在项目的风格以及所使用的设计模式中。

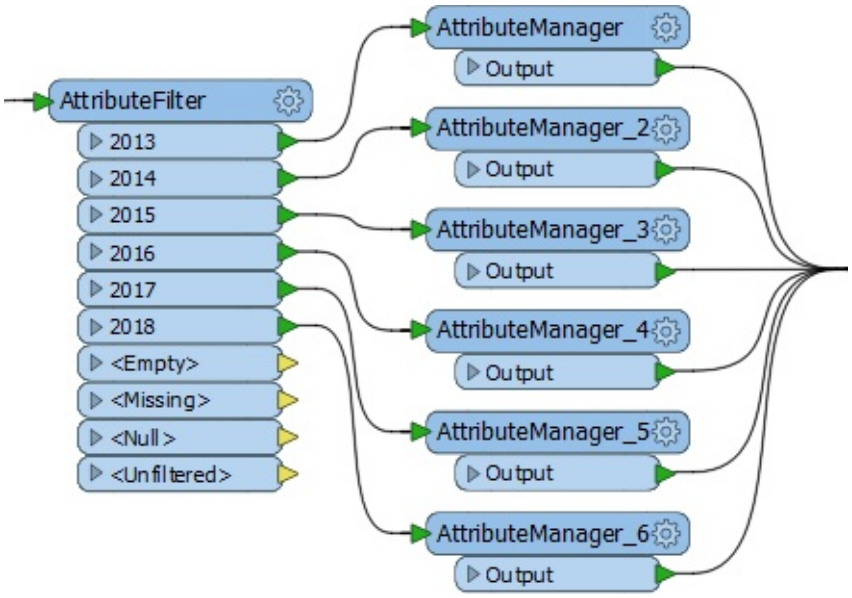
维护方法

当工作空间的设计不允许轻松更新或扩展时，维护方法很薄弱。一些关键指标表明了设计薄弱。

重复的转换器

一次又一次地复制相同的转换器 - 可能创建多个流来执行此操作 - 表示设计较弱。

例如，这里的多个AttributeManager转换器将导致维护问题：



第一个问题是每增加一年需要一台新转换器来处理它。第二个问题是AttributeManager中的一个小变化需要单独编辑每个转换器。

我们还可以说这种布局导致的样式不像应该的那样紧凑。

这里的解决方案是使用单个AttributeValueMapper转换器（或可能是SchemaMapper）替换所有上述内容，以允许使用最少的编辑次数进行扩展。

一般的解决方案是观察多个转换器的实例，并尝试评估是否有更好的设计。

复杂性

当工作空间设计比必要时更复杂时，工作空间设计很薄弱。复杂性意味着解决方案变得难以维护并且难以扩展，尤其是在由原作者以外的人执行时。

需要注意各种类型的复杂性：

- 多余的脚本：当等效转换器已存在时使用Python脚本
- 低级复杂性：当已存在等效转换器时使用FME功能和工厂
- 多个工作空间：当多个工作空间以不必要的方式链接时
- 工作空间样式：当工作空间样式如此复杂时，很难理解

在将项目投入生产之前，值得请一位同事评估它是否清晰。他们无法轻易理解的工作空间部分是未来作者也难以维护的要点。

技巧
对于具有强大脚本技能的新FME用户，Python转换器提供了特别诱人的快捷方式。但是，其他工作空间作者可能不熟悉Python，并且发现没有完整的Python开发环境就很难调试。

数据库连接

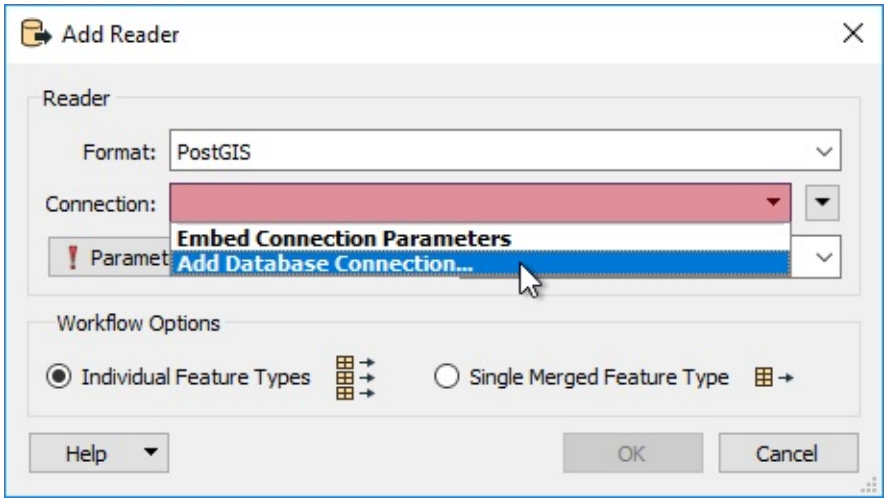
数据库格式需要连接参数。有时这些参数需要在同一工作空间中多次应用，有时在切换平台时需要更改参数（例如从测试到实时环境）。

可以在工作空间中嵌入数据库连接值，并将该信息添加到需要的任何位置。但是，更好的解决方案涉及FME中称为数据库连接的工具。

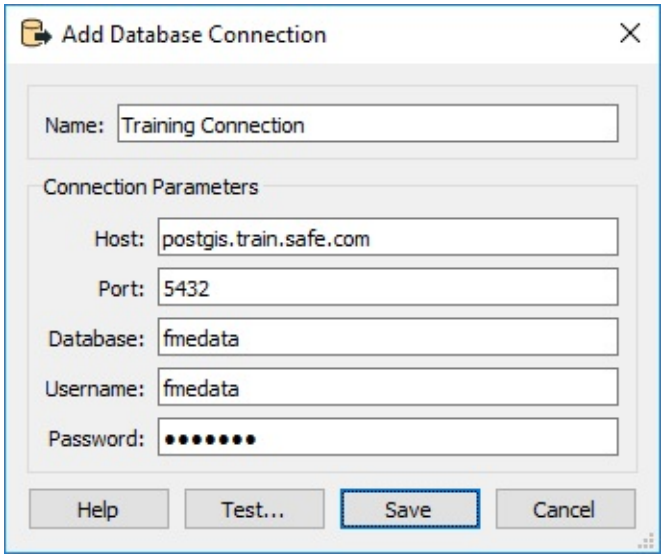
数据库连接是一组预定义的连接和身份验证参数，存储在单个名称下。创建后，将使用连接名称而不是实际参数。

创建数据库连接

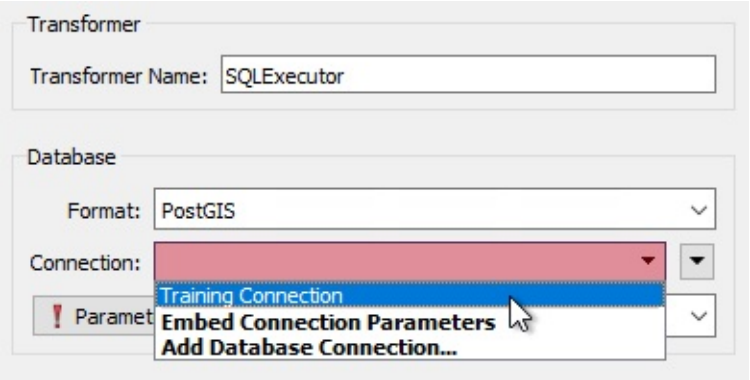
使用数据库格式并提示输入连接时，请选择添加连接的选项：



通过选择数据库类型，输入连接参数以及为连接指定名称来定义新连接：



现在，每当需要数据库连接时（此处在SQLExecutor转换器中），都会选择预定义的连接：



现在，重复使用数据库不需要重复输入相同的证书，如果证书更改，则只需要编辑一个连接。这两个功能都有利于工作空间的维护和扩展。

另一个好处是安全性。嵌入式连接信息存储在工作空间文件中，存在潜在的安全风险。数据库连接安全地存储在工作空间之外，并且其他人无法轻松访问。如果工作空间传递给另一个用户，则必须使用自己的参数设置连接。

消防员**Mapp**说.....
安全比抱歉更好。你可不想被糟糕的设计弄得焦头烂额。如果您对工作空间不确定，请咨询其他FME用户，或联系 Safe Software 支持团队获取建议。 另外，请务必查看有关FME和代码味道的博客文章。

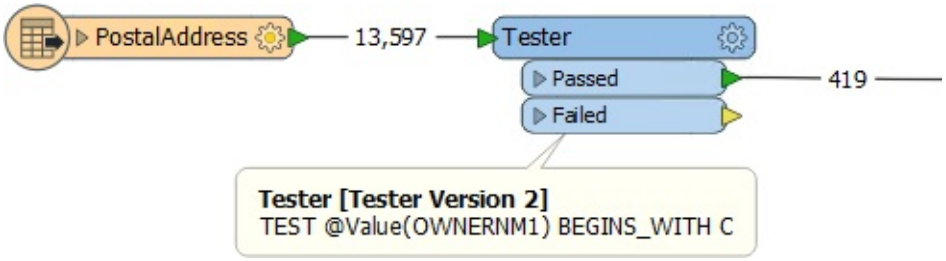
性能方法

当工作空间的设计导致工作空间使用比必要更多的系统资源（CPU和内存）时，性能方法很弱。与维护方法一样，一些关键指标表明设计缺陷。

性能方法意味着设置工作空间以便熟练地运行。

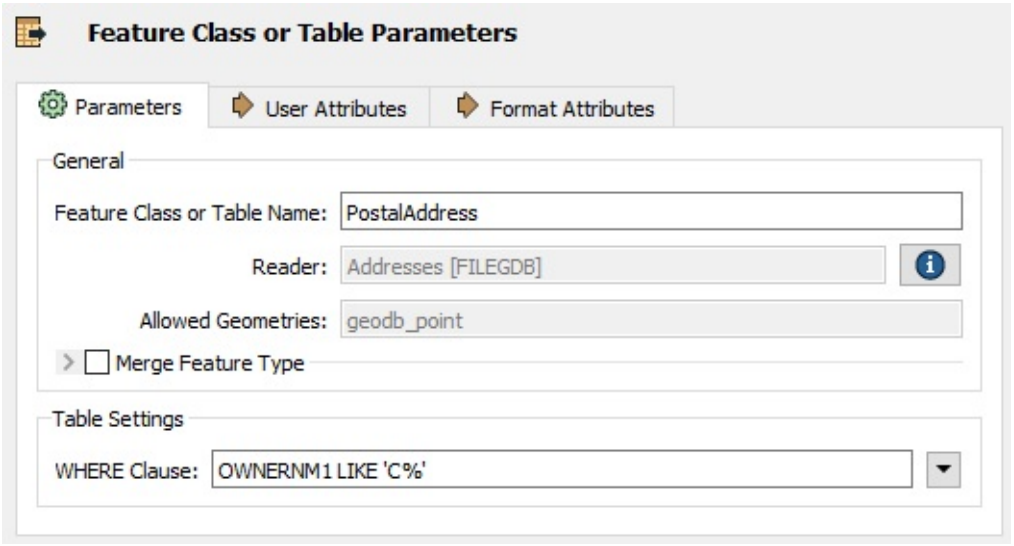
过滤输入

FME中的常见方案是将数据读入工作空间，然后过滤掉不需要的要素（记录）：



但是，当读取数据并立即丢弃时，用于读取该数据的资源会直接损失性能。

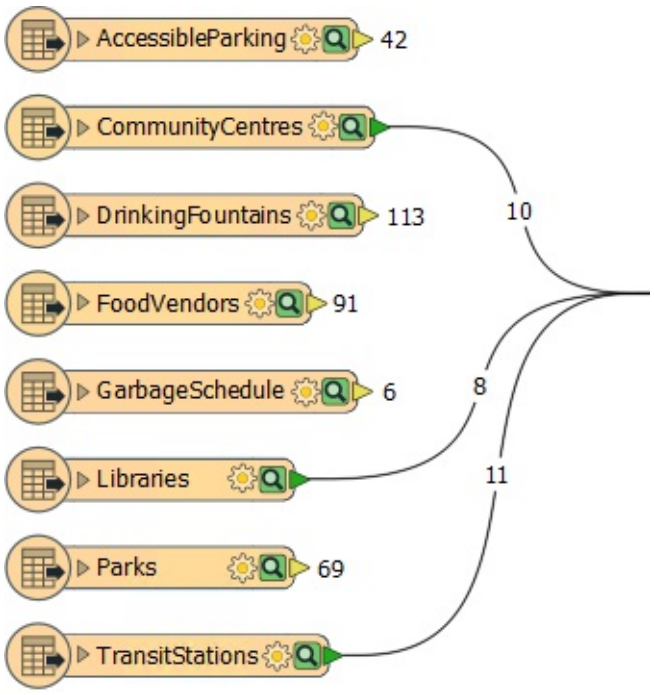
如果数据在读取时被过滤 - 而不是之后 - 性能要好得多，并且许多格式都有参数可以做到：



这里的'WHERE子句'参数将所需的过滤器直接应用于地理数据库读模块。仅读取与where子句匹配的数据并进入工作空间。

多余的要素类

源数据集的模式在FME画布上由要素类对象表示：



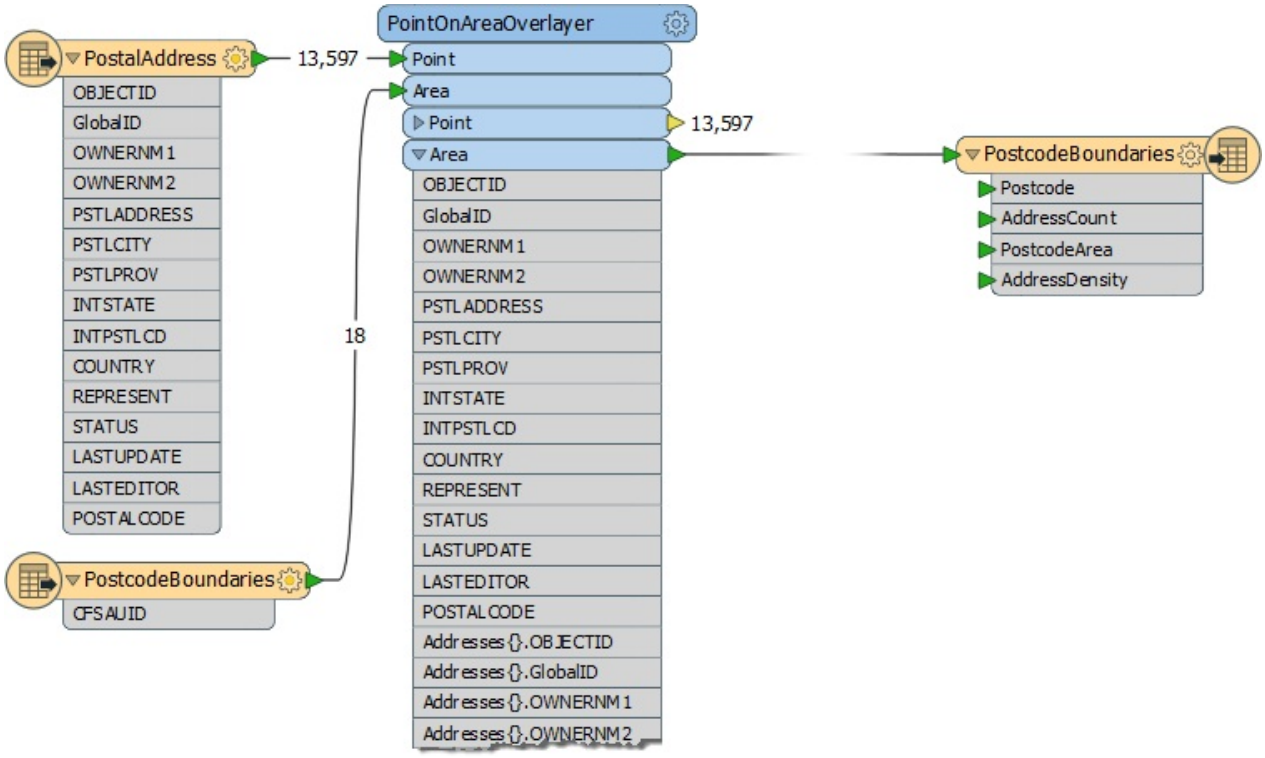
不将要素类连接到工作空间中的其他对象等同于读取和丢弃数据，同样对性能有害。

添加读模块时，FME会提示用户选择要添加到转换的要素类。您应该避免添加不需要的要素类，并删除已添加但未连接的要素类。

Workbenchy医生说...
多余的要素类不仅会减慢您的工作速度，还会使画布变得杂乱，并且难以保持干净整洁的样式。

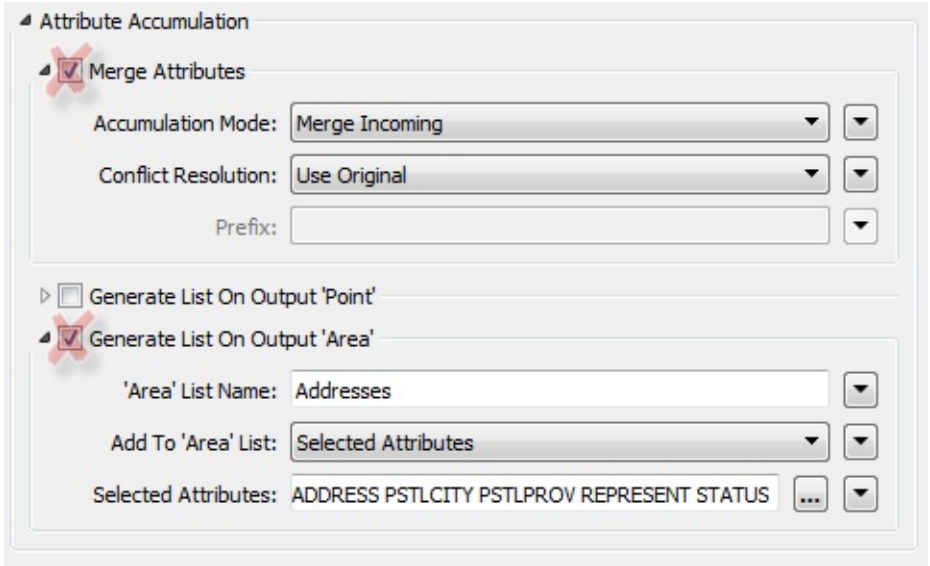
多余的属性和列表

将数据读入工作空间后，仍可缩小其大小以协助性能。例如，输出模式中未定义的属性对于工作空间不是必需的，可以删除：



这里工作空间作者正在计算城市的每个邮政编码（CFSAUID）中的地址数量。输出模式中不需要地址属性，但会将其复制到邮政编码要素，以及地址列表以及工作空间末尾的所有内容。

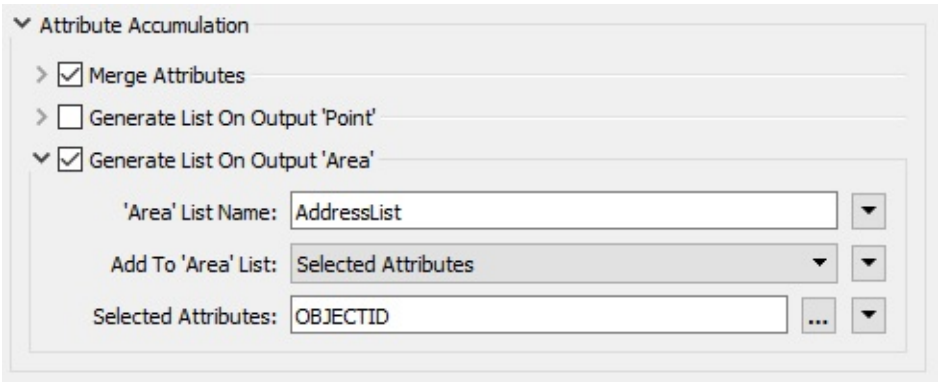
在这种情况下，作者应避免使用PointOnAreaOverlayer选项来复制属性和创建列表：



此步骤将减少运行转换所需的内存量，而不会影响其输出。

Workbench医生说...

列表是要保留的最糟糕的属性类型，没有任何理由，因为它们可以为每个记录设置多个值。许多连接转换器中的参数允许作者仅生成所需的列表属性：



错误陷阱

有时扩展/性能意味着使用更多不同类型和质量的数据集。如果不考虑数据质量，可能会影响未来的性能。

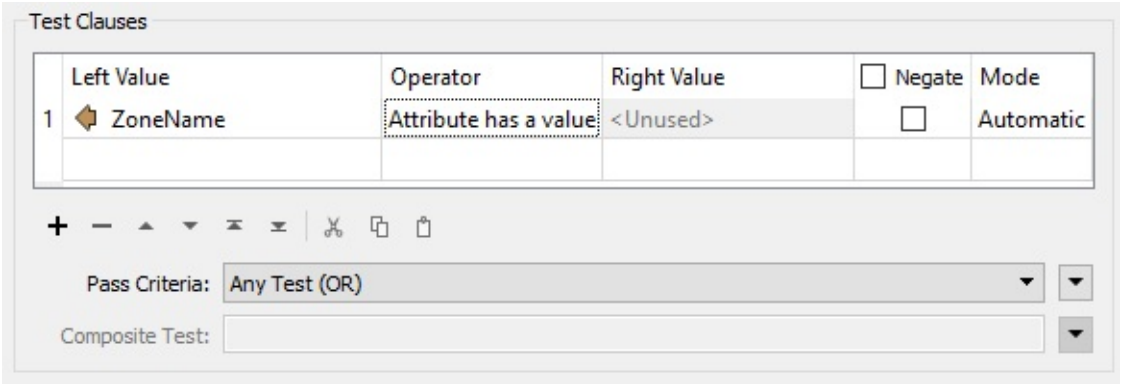
设计未来功能的一种方法是使用错误捕获。

错误捕获是一种设计工作空间的方法，使得意外数据不会导致工作空间失败。作者试图预见可能出现的数据问题并构建处理它们的方法。

错误捕获可以像添加测试或过滤器转换器一样简单，以清除不良要素，也可以更复杂，并包括根据具体情况以不同方式处理数据的方法。

技巧

Tester转换器有一个运算符，用于测试属性是否具有值：



这对于错误捕获非常有用，在尝试将其用作参数源之前测试属性是否具有值。

练习：设计模式

练习**2**	方法
数据	地址（Esri地理数据库） 犯罪数据（CSV - 逗号分隔值） 公园（MapInfo TAB）
总体的目标	参与温哥华可行走性项目
演示	方法最佳实践
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\BestPractice-Ex2-Begin.fmwt
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\BestPractice-Ex2-Begin.fmwt

继续上一个练习，您已被分配到一个项目，以计算温哥华市每个地址的“步行性”。步行性衡量步行访问当地设施的难易程度。初始工作空间分析了该地区的犯罪行为，并添加了一个部分来计算到最近的公园的距离。

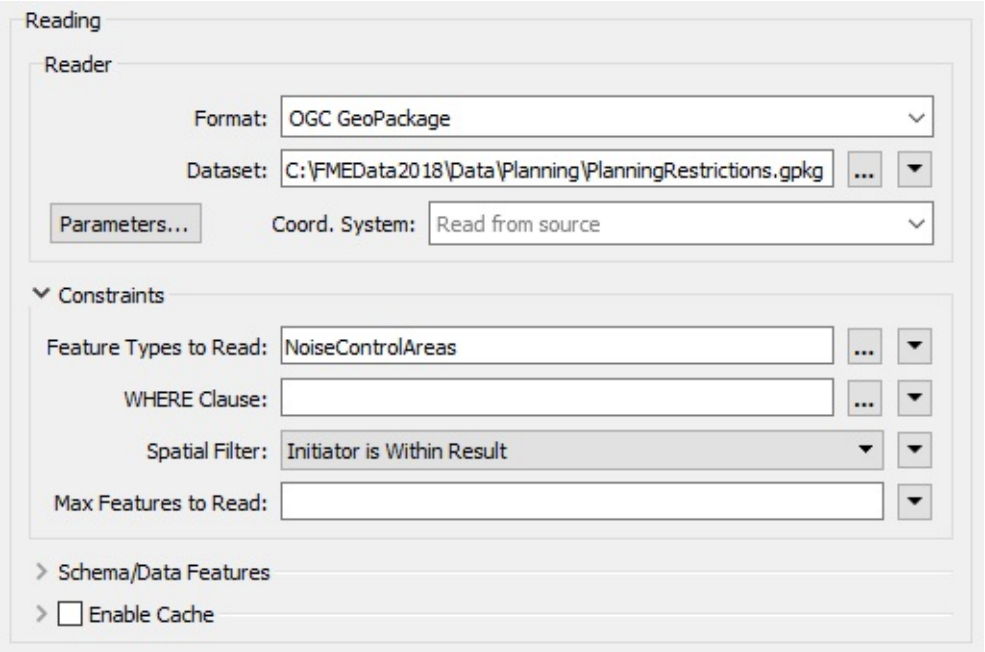
现在我们可以扩展项目，看看每个地址是否属于噪声控制区域。

1）添加读模块
我们的第一个任务是读取噪声控制区域数据。这可以在以下数据集中找到：

读者格式	**OGC GeoPackage**
读模块数据集	C:\FMEData2018\Data\Planning\PlanningRestrictions.gpkg

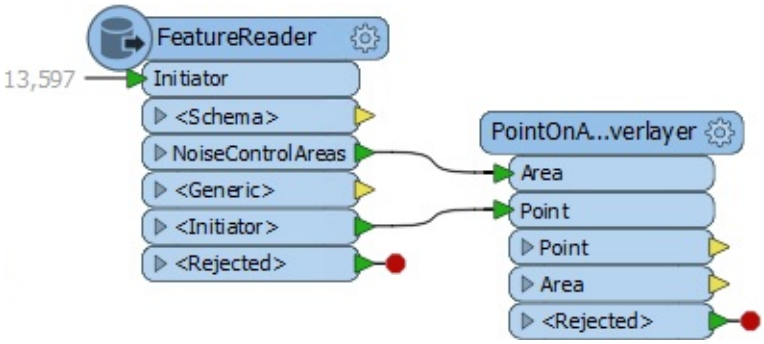
我们将尝试使用FeatureReader转换器来读取此数据，使用地址作为触发器来提高性能; 所以放置一个FeatureReader转换器并将其连接到AttributeRenamer。

检查转换器的参数。如上所述设置读模块参数。只读取NoiseControlArea要素类型并将Spatial Filter参数设置为Initiator is Result：



以这种方式读取数据可能有益于性能，因为我们只读取了我们知道地址落入其中的噪声控制区域。我们拭目以待...

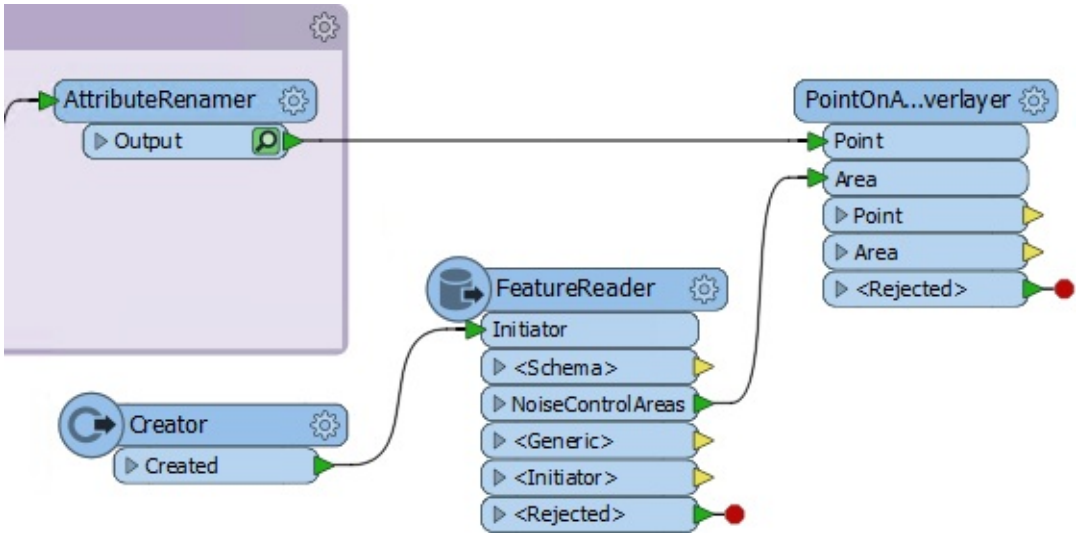
2）覆盖地址和噪声控制区域
要进行覆盖，请添加PointOnAreaOverlayer转换器。将FeatureReader：<Initiator>连接到Point输入端口，将FeatureReader：NoiseControlArea连接到Area输入端口：



检查参数并启用“合并属性”选项。使用Run to This运行工作空间; 遗憾的是，工作空间需要很长时间才能运行，因此请按“停止”按钮取消转换。

3）添加**Creator**
很明显，当前设置不是最佳解决方案。FeatureReader为每个重叠的地址输出一个面要素，导致PointOnAreaOverlayer减速。我们的赌博失败了。

因此，添加一个Creator转换器来触发FeatureReader，并将其与AttributeRenamer断开连接，保持其他连接不变：



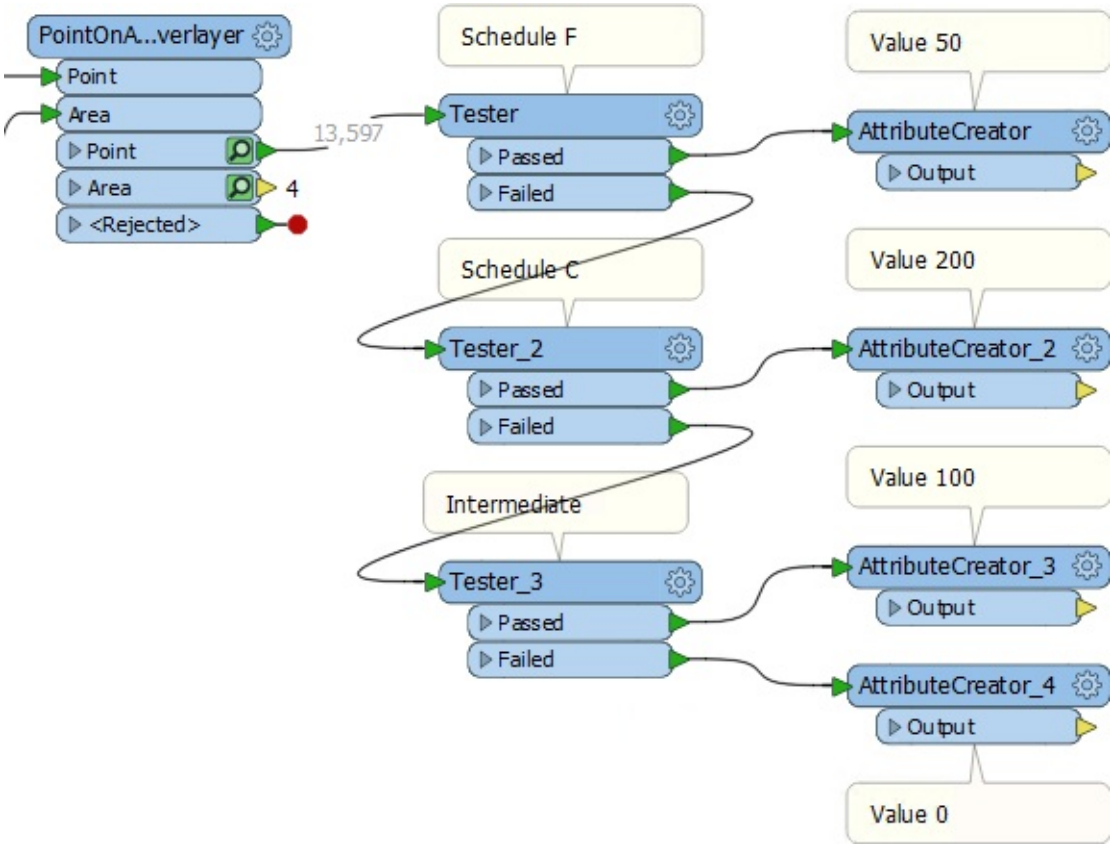
检查FeatureReader参数并将Spatial Filter参数重置为<无空间过滤No Spatial Filter>。

现在重新运行工作空间的这一部分并检查输出以确保某些地址要素正在接收AreaID，AreaName和AreaDescription属性。

4）设置数值
为了帮助评估步行性，我们将为每个不同的区域提供一个数值，相对于在城市中行走的居民所能获得的福利，：

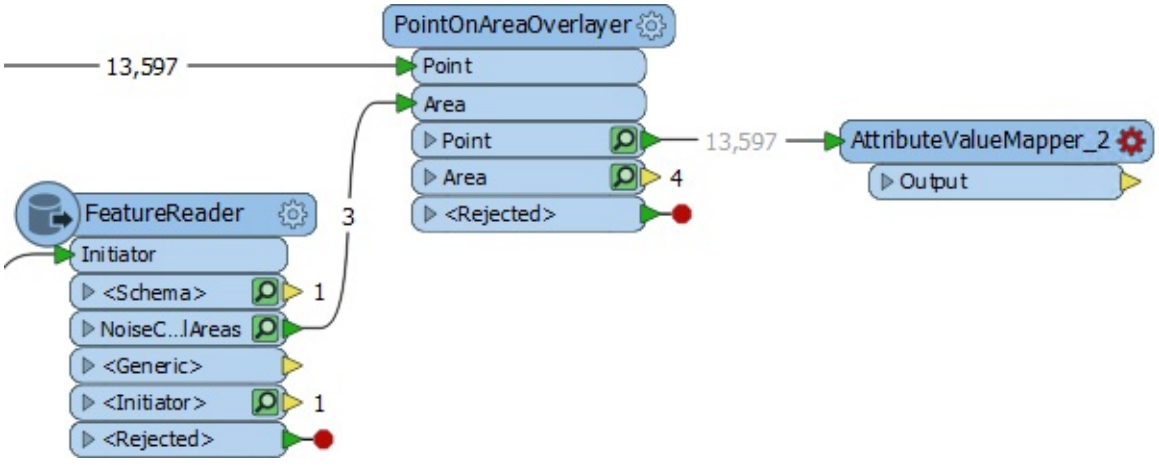
计划**F.**	**50**
计划C	200
中间	100

一种技术是使用Tester和AttributeCreator转换器，如下所示：



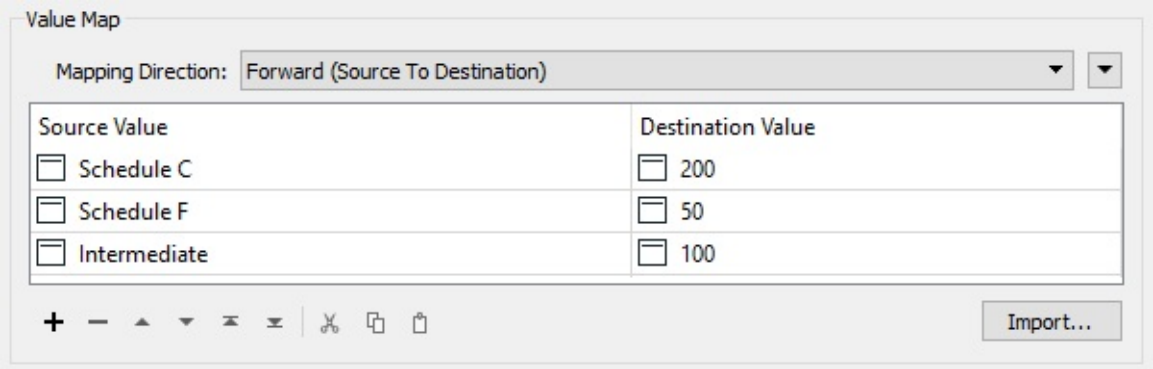
然而，这显然是一种重复。这种方法很难维护; 如果添加了额外的计划表，则需要为每个表添加一个额外的Tester / AttributeCreator组合。

因此，替代为放置一个AttributeValueMapper转换器连接到PointOnAreaOverlayer:Point输出端口：



检查AttributeValueMapper的参数。将其设置为从AreaName映射到NoiseZoneScore，默认值为零（0）：

现在使用上面表格中的值设置映射：



您可以手动设置（通过键入值）或自动设置（通过使用“导入”按钮）。

在工作空间的此部分周围放置一个书签并运行它。检查缓存的结果以证明地址被赋予正确的值。

5）清理输出
因为有太多多余的属性使显示混乱，检查输出变得困难。这些属性很难帮助工作空间的性能 - 即使在开发过程中使用缓存可以减轻这种影响。

检查模板文件BestPractice-Ex2-Begin.fmwt的大小。你会发现它的大小接近46mb，这对于模板来说相当大。立即保存工作空间，文件可能会更大。拥有一个大型模板文件并不是一个问题，但它确实表明正在缓存大量数据，这可能会影响工作空间的性能。

数据的一个方面是属性和列表的数量。要删除一些这些多余的信息，请将AttributeRemover转换器放在工作空间的末尾。

检查其参数并将其设置为删除您怀疑输出中不需要的任何属性和列表。

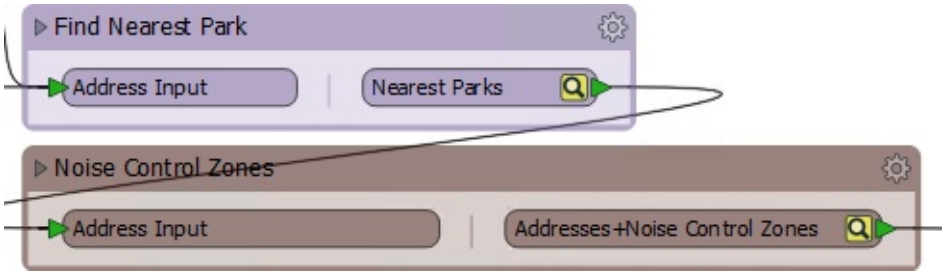
一个有意思的项目是一个名为CrimeList{}的列表属性，，对于此转换的任何部分而言似乎都不需要它。按Ctrl + F并搜索该短语来追踪其来源。您会发现它是由Aggregator转换器创建的。

检查Aggregator转换器的参数并关闭Generate List参数，以防止创建列表。此步骤将导致许多缓存变得陈旧，但我们将很快重新运行工作空间以解决此问题。

5）折叠书签
过度缓存的另一个来源是转换器产生我们不需要检查的输出。可以通过在折叠的书签中隐藏这些转换器来防止这些。

因此，通过工作空间折叠书签，我们知道其中的转换器正在产生正确的输出：

您可能希望重命名书签上的某些端口以阐明它们的作用：



6）重新运行工作空间
现在通过单击AttributeRemover并选择*Run to This*重新运行工作空间。

工作空间将运行，数据将被缓存，但仅限于书签的输出点。此外，还将删除输出不必要的属性。

将工作空间另存为新模板，并选中包含缓存的选项。这样，工作空间就可以在以后继续使用了。

检查新模板的文件大小。它应该相当小（大约14mb）。

恭喜
通过完成本练习，您已学会如何： • 评估读模块的表现以及什么时候FeatureReader不是一个好的选择 • 避免重复转换器以提高维护和可扩展性 • 删除不必要的属性以提高性能 • 跟踪不必要的列表并将其删除 • 通过折叠书签来提高性能，以防止过多的缓存

调试

即使是熟练的FME用户也很少生成零缺陷的新工作空间。因此，所有用户都必须了解FME中可用的调试技术。

调试方法

通常，FME中的调试包括首先确定问题是否已经发生，然后追踪问题的根源（即，工作空间中的位置是否发生）。一旦确定问题出现在哪里，就可以调查问题的性质。

FME中提供了各种调试技术，以正确的顺序使用它们非常重要。例如，当简单的日志消息已经解释了问题时，更改参数并重新运行工作空间并不是特别有用！

逻辑顺序是：

- 检查是否有任何问题
  - 解析日志中的警告和错误
  - 检查输出数据集
- 找到问题所在
  - 查看连接要素计数
  - 检查转换关键阶段的数据
- 确定问题
  - 在故障点检查读模块，写模块或转换器参数
  - 如有必要，请在要素调试模式下运行工作空间

Workbench医生说...
在软件开发领域，在完成后调试一个进程称为“事后调试”！我们正试图找出导致失败的原因。 如果我们无法通过事后检查确定原因，我们将重新启动该过程，并启用各种跟踪选项。我们可以称之为“交互式调试”。

日志记录和日志解析

FME日志包含转换中所有阶段和过程的记录。因此，内容对于调试目的至关重要。

日志消息类型

日志窗口中显示了不同的消息类型，包括：

错误：一个错误，在日志中用红色文本和术语**ERROR**表示, 代表有问题导致FME终止处理。例如，由于用户权限不正确，FME无法写入输出数据集。

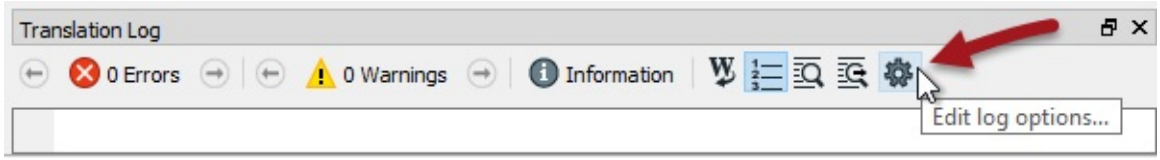
警告：一个警告，由蓝色文本和术语**WARN**表示，意为处理问题。这个问题小到足以让FME完成转换，但输出可能会受到不利影响，应该检查。例如，FME无法写入要素，因为它们的几何对象与写模块格式不兼容。要素将从转换中删除，并在日志中发出警告。

信息：信息消息，由术语**INFORM**表示代表可以帮助用户确定他们的转换是否已被正确处理的一条信息。例如，FME有时会记录特定数据集参数的确认，例如坐标系。

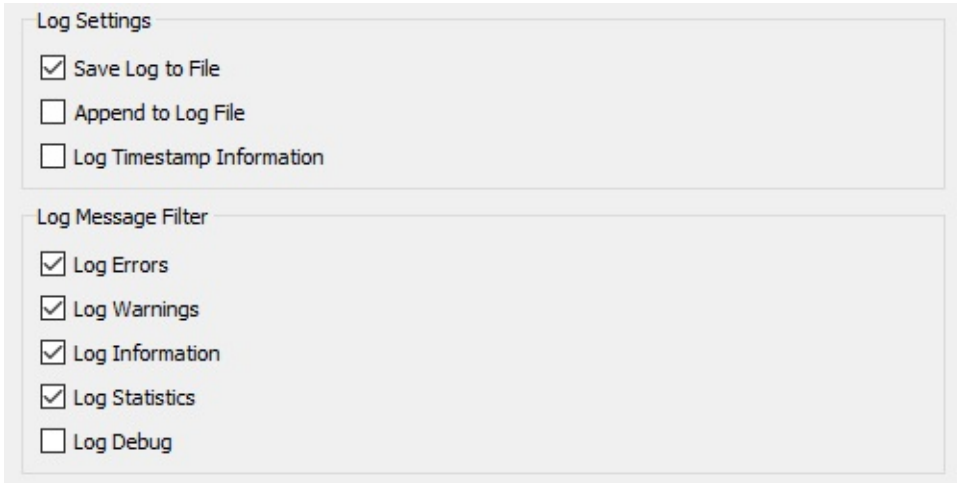
统计：统计信息，由术语**STATS**表示，提供与转换有关的各种数字的信息; 例如，从源数据集读取的要素数量，以及执行此操作所需的时间。

日志文件选项

日志文件选项对于设置（在运行工作空间之前）非常重要，这样我们就可以在我们需要的结构中接收信息。可以通过日志窗口上的齿轮图标访问这些选项：



日志选项允许我们控制数据的呈现方式和格式：



在调试工作空间的性能时，“日志时间戳信息”选项至关重要，因为它会添加有关每个步骤所用时间长度的信息。它还会导致显示消息类型（ERROR，WARN等）。

过滤消息的能力并不是特别重要，因为可以在日志窗口中直接过滤消息类型。

Log Debug选项使FME在FME引擎中显示更深层次的消息。这个选项不应开启用作一般用途，但只有当工作空间已经显示为失败时，并且需要额外的信息。否则，可能会被正常过程中的异常消息混淆。

Intuitive修女说.....
在跟踪与HTTP相关的问题时，Log Debug特别有用。

空间日志文件

除了将日志写入文本文件（<工作空间名称> .log）之外，FME还会写入空间日志：

空间日志是日志中提到的要素（以FME要素存储格式）的数据集 - 由于来自FME的警告或使用Logger转换器。

可以在FME Data Inspector中打开数据集，以检查要素并识别导致它们被拒绝的任何问题。

解析日志窗口

日志窗口应该是检查转换完成时的第一个位置。它会告诉用户是否有任何相关的错误或警告。

错误

如果发生ERROR，则可能会暂停转换。会有很多红色文本和一些终止语句，例如：

> 程序终止
>
> 转换失败。

可能有几个ERROR消息，因此向上滚动日志窗口以尝试识别其中的第一个，这可能是问题的原因。例如这条消息：

> ERROR |连接到PostgreSQL数据库时出错（host ='postgis.train.safe.com'，port ='5432'，dbname ='fmedata'，user ='fmedata'，password ='***'）：'致命：用户“fmedata”的密码验证失败致命：用户“fmedata”的密码验证失败

...验证数据库连接是一个明显的问题。

警告

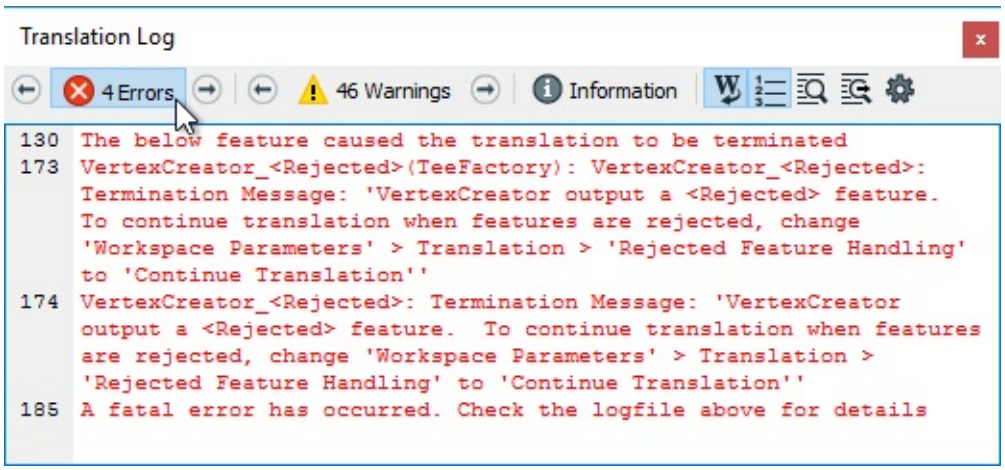
即使转换成功，检查日志以获取以下注释也很重要：

> 转换成功与X警告（s）

如果有任何警告（即，如果X> 0），则使用搜索选项查找单词WARN。任何警告消息都可能对输出数据的质量产生重要影响。

新知识

FME2018引入了在日志窗口中以交互方式按类型过滤消息的要素。在这里，用户单击了一个按钮，仅查看工作空间内的ERROR消息：



这对于调试非常有用，特别是对于色觉缺陷的用户而言，他们无法轻易区分红色错误和蓝色警告。

检查输出

即使工作空间在没有警告或错误的情况下运行完成，这并不意味着输出与预期的或需要的匹配。无论出于何种原因，工作空间可能以错误的方式生成数据。我们可以通过检查转换输出来确定这一点。

检查输出仅仅是在FME Data Inspector中或在要使用数据的应用程序中查看输出的情况。

如本手册中所述，可以检查数据的许多不同方面，包括以下内容：

- 格式: 数据是否为预期格式？
- 模式: 数据是否细分为正确的图层，类别或类？
- 几何: 几何对象是否在正确的空间位置？几何类型是否正确？
- 符号: 每个要素的颜色，大小和样式是否正确？
- 属性: 是否存在所有必需属性？是否遵循所有完整性规则？
- 数量: 数据是否包含正确数量的要素？
- 输出: 转换流程是否按预期重组了数据？

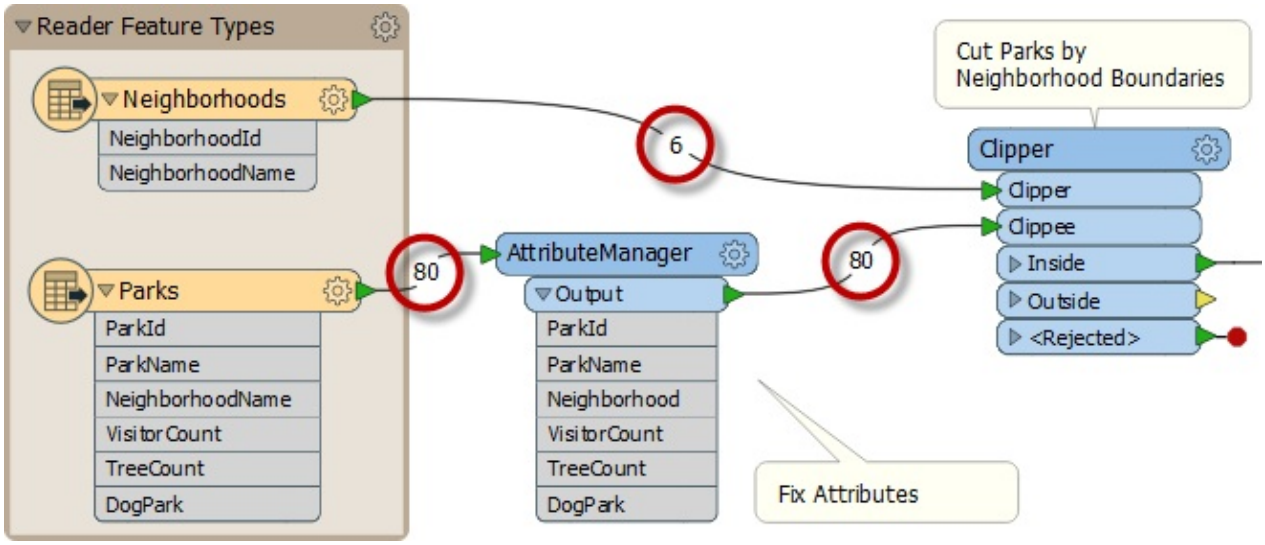
检查数据集并查看其中的任何组件是否不正确应该很简单。

Workbench医生说...

这个阶段仅仅是确定是否有任何问题。 如果没有问题，那么我们可以满意转换是成功的。 如果有问题，我们应该继续确定问题发生的位置。在这一点上不要急于下结论是很重要的。输出不正确的事实并没有告诉我们这个问题是在哪里引起的。

要素计数

工作空间要素计数是指转换完成后每个连接上显示的数字：



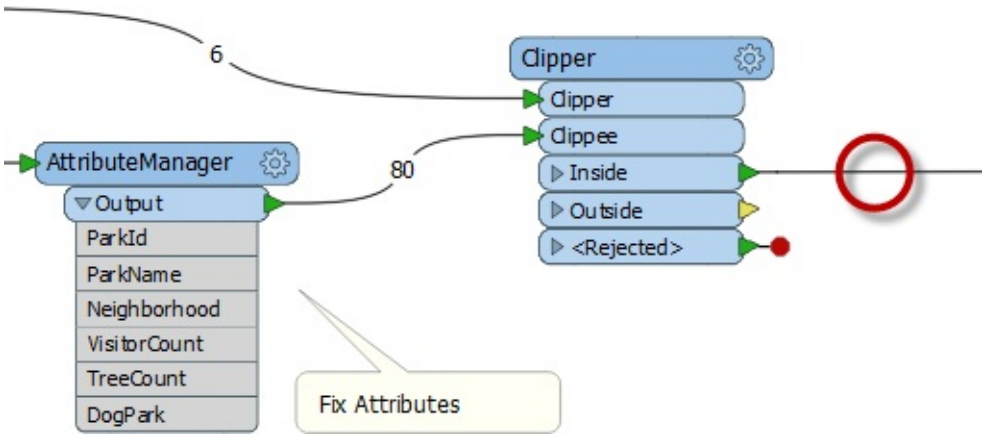
一旦错误或问题被确定为存在，要素计数帮助我们找出哪里这个问题的发生。

在上面的屏幕截图中，如果Clipper输出不正确，那么您将检查先前的要素计数以查看是否有任何计数看起来错误（可能您知道有七个街区，但要素计数仅显示六个）。

输出不正确

当输出要素的数量不正确时，有几件事要检查。

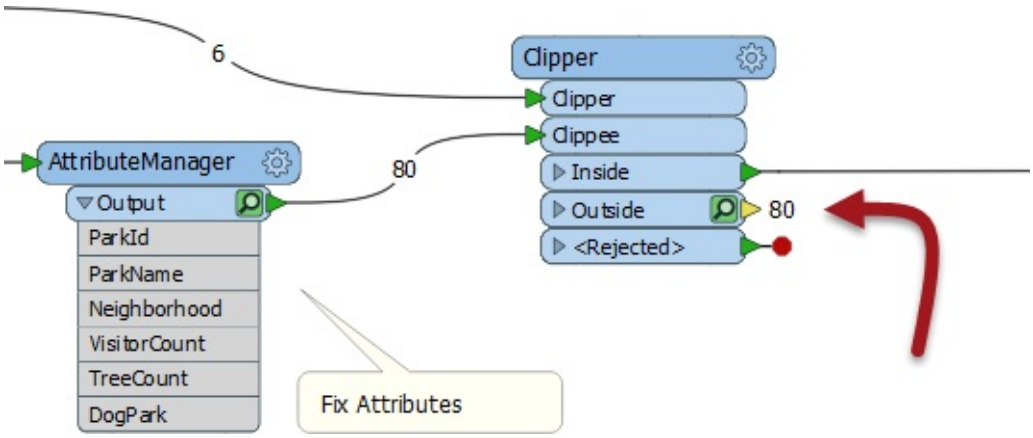
如果输出为零，并且要素计数显示所有要素都已进入转换器，但没有出现，那么您可以相信转换器是导致问题的原因：



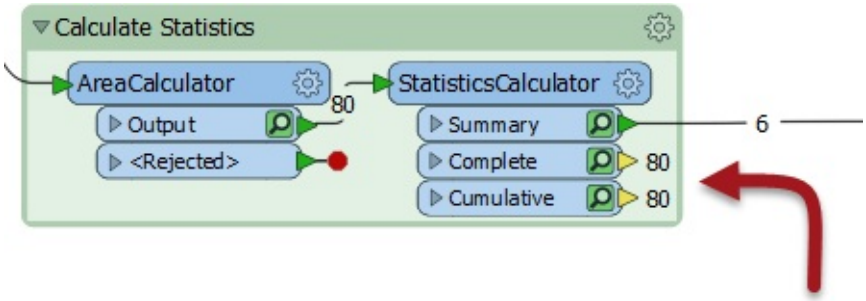
例如，在这里，80个要素进入Clipper转换器（将被剪切为单个边界）但没有出现。Clipper转换器几乎可以肯定是输出错误的原因。

数据不会被视为无效; 它只是没有通过预期的测试。Clipper和Clippees可能不占用相同的坐标系; 因此，一个没有落在另一个里面。

启用要素缓存有助于确认是这种情况：



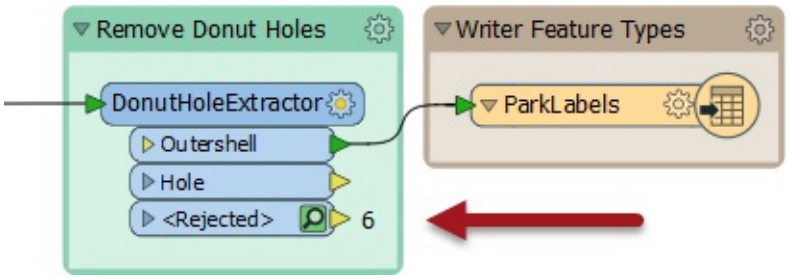
或者 - 这是缺少要素的常见原因 - 作者连接了错误的输出端口！例如，此用户已连接StatisticsCalculator Summary输出端口，其实他们真正想要连接是Complete端口时，：



要素缓存再次为我们提供了我们应该连接的端口的线索。

被拒绝的要素

有时当要素丢失时，它们会被转换器拒绝。许多转换器包含一个<Rejected>端口来输出这些无效要素：



请记住，即使关闭了要素缓存，要素也会自动计数并存储在<Rejected>端口上。

作为额外的好处，被拒绝的要素通常包括拒绝代码属性：

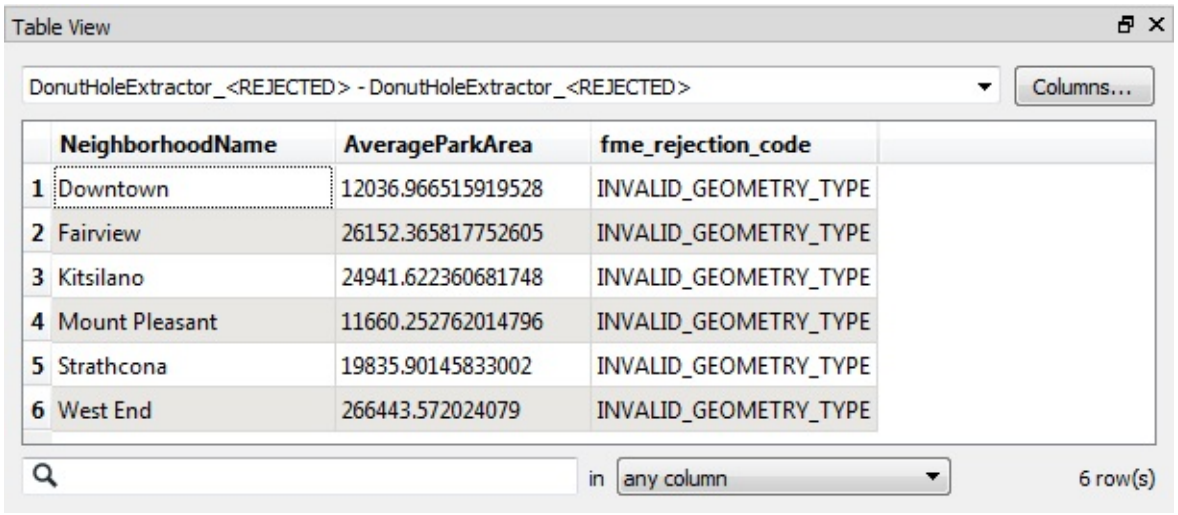


Table View
DonutHoleExtractor_<REJECTED> - DonutHoleExtractor_<REJECTED>
Columns...
NeighborhoodName
AverageParkArea
fme_rejection_code
1 Downtown 12036.966515919528 INVALID_GEOMETRY_TYPE
2 Fairview 26152.365817752605 INVALID_GEOMETRY_TYPE
3 Kitsilano 24941.622360681748 INVALID_GEOMETRY_TYPE
4 Mount Pleasant 11660.252762014796 INVALID_GEOMETRY_TYPE
5 Strathcona 19835.90145833002 INVALID_GEOMETRY_TYPE
6 West End 266443.572024079 INVALID_GEOMETRY_TYPE
in any column
6 row(s)

检查关键阶段

如果要素计数无法确定问题的位置，则下一步是检查转换关键阶段的数据。

通常，输出数据集中的问题是由于以下原因之一而发生的：

- 数据读取错误
- 数据被错误地转换
- 数据写错了
- 另一个应用程序错误地解释了数据

例子

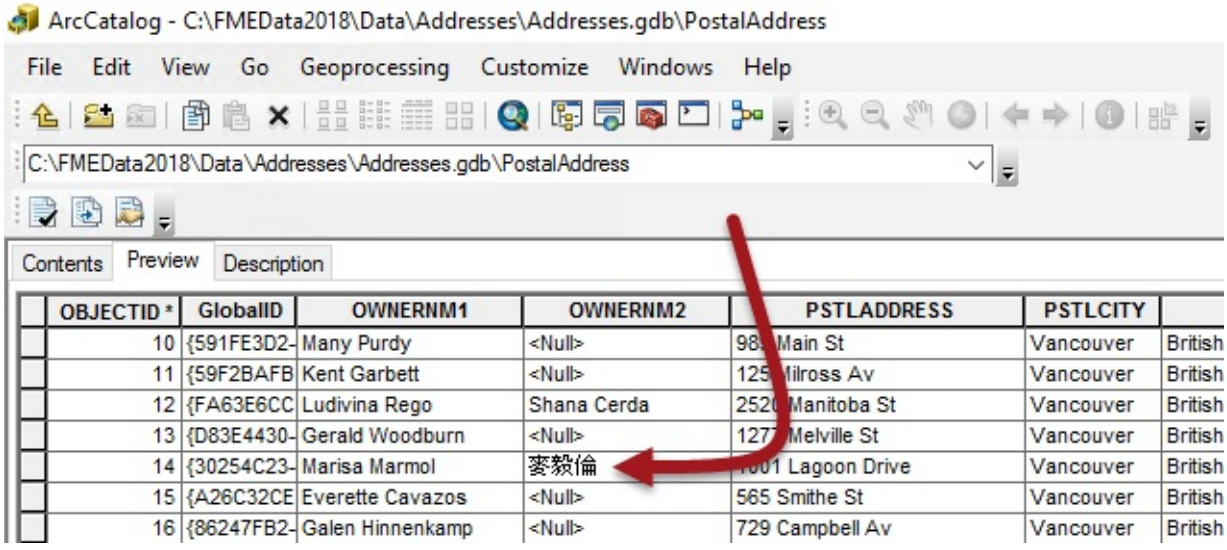
从地址的文件地理数据库到CSV格式的转换会在输出中错误地呈现属性值。例如，名称“麦毅伦”呈现为“???”。

显然，编码问题已经发生，但在哪里呢？数据在另一个应用程序中显示不正确的事实并不能真实地指出问题的引入位置。

在这种情况下，作者应检查关键阶段的数据，以确定数据最后看起来是否正确。

在读取之前检查源数据

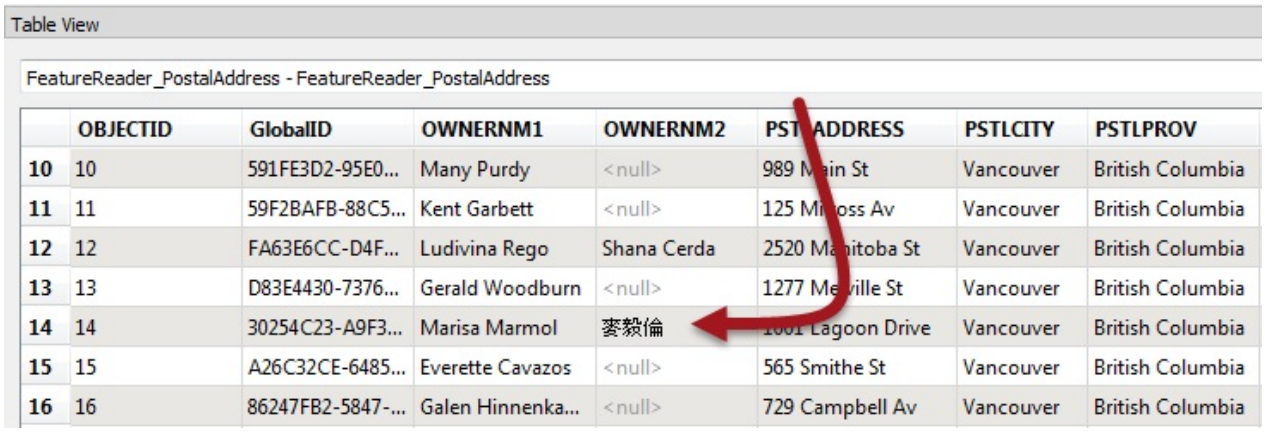
无需检查即可轻松假设源数据正确无误。因此，如果可能，应在其本机应用程序中检查源数据：



显然，如果源上的数据不正确，那么转换输出就不太可能是正确的。但是在我们的示例中，它在ArcGIS中是正确的，因此转换应该有效。

读取后检查源数据

如果源数据在其本机应用程序中是正确的，则使用FME进行检查。可以直接在FME Data Inspector中打开数据，或者 - 如果在启用缓存的情况下运行工作空间 - 从FME Workbench中打开它：



	OBJECTID	GlobalID	OWNERNM1	OWNERNM2	PST ADDRESS	PSTLCITY	PSTLPROV
10	10	591FE3D2-95E0...	Many Purdy	<null>	989 ain St	Vancouver	British Columbia
11	11	59F2BAFB-88C5...	Kent Garbett	<null>	125 Mi oss Av	Vancouver	British Columbia
12	12	FA63E6CC-D4F...	Ludivina Rego	Shana Cerda	2520 Ma itoba St	Vancouver	British Columbia
13	13	D83E4430-7376...	Gerald Woodburn	<null>	1277 Me ville St	Vancouver	British Columbia
14	14	30254C23-A9F3...	Marisa Marmol	麥毅倫	1001 Lagoon Drive	Vancouver	British Columbia
15	15	A26C32CE-6485...	Everette Cavazos	<null>	565 Smithe St	Vancouver	British Columbia
16	16	86247FB2-5847-...	Galen Hinnenka...	<null>	729 Campbell Av	Vancouver	British Columbia

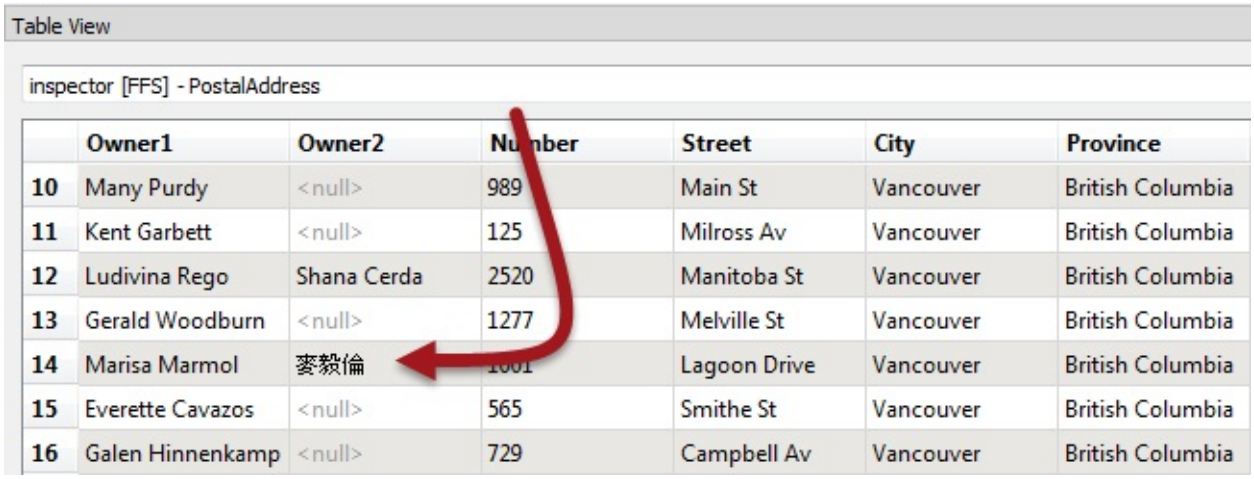
如果此时数据不正确，那么使用FME读取数据的过程就出错了。同样，在我们的示例中，数据是正确的，因此我们必须继续诊断问题。

写入前检查数据

现在我们应该检查转换和写入之间的数据。

如果在启用缓存（运行>使用要素缓存运行）的情况下运行工作空间，则数据已可用于检查。否则，请启用此选项（或写模块> 重定向到FME Data Inspector）并重新运行工作空间。

在将数据写入输出之前检查数据以查看它是否正确：



	Owner1	Owner2	Number	Street	City	Province
10	Many Purdy	<null>	989	Main St	Vancouver	British Columbia
11	Kent Garbett	<null>	125	Milross Av	Vancouver	British Columbia
12	Ludivina Rego	Shana Cerda	2520	Manitoba St	Vancouver	British Columbia
13	Gerald Woodburn	<null>	1277	Melville St	Vancouver	British Columbia
14	Marisa Marmol	麥毅倫	1001	Lagoon Drive	Vancouver	British Columbia
15	Everette Cavazos	<null>	565	Smithe St	Vancouver	British Columbia
16	Galen Hinnenkamp	<null>	729	Campbell Av	Vancouver	British Columbia

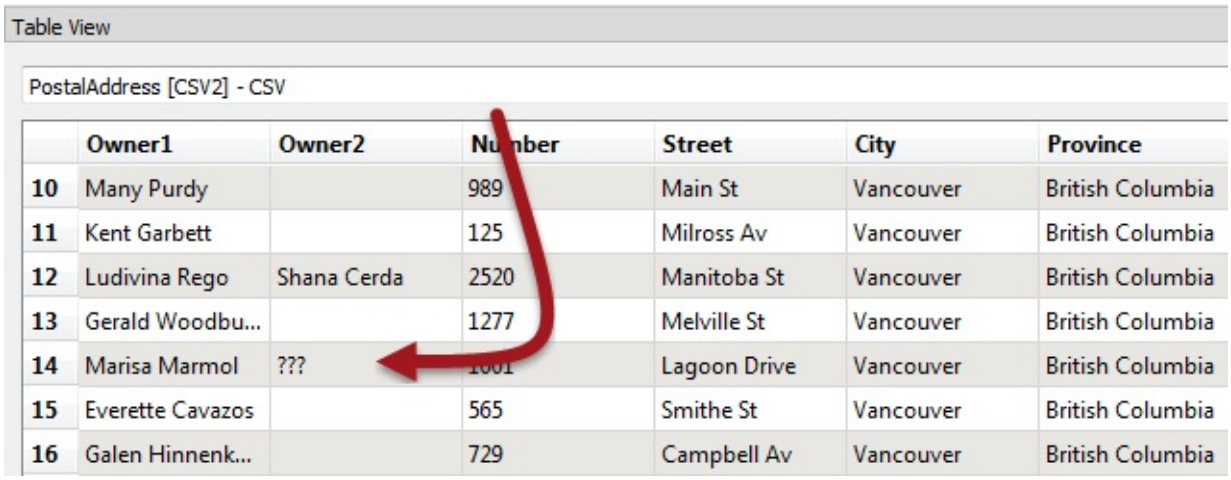
如果Data Inspector此时显示 字符，那么我们可以假设在写入之前数据转换中出现问题。您可以使用要素缓存来检查过程的每个步骤，以找出出现问题的转换器。

在我们的示例中，数据仍然是正确的，因此我们应该测试下一个输出数据集。

检查输出数据集

如果在写入之前数据是正确的，那么可能是它被错误地写入了。

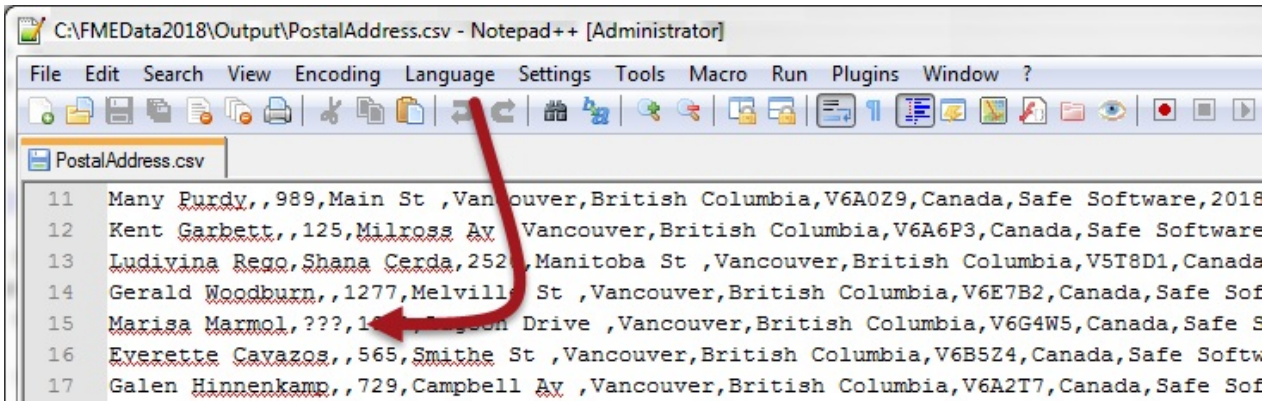
在FME Data Inspector中打开输出数据集。此步骤将显示FME写入的数据（当然，请将其读回）。如果这里的数据不正确，那么在写入数据期间可能会出现问题：



	Owner1	Owner2	Number	Street	City	Province
10	Many Purdy		989	Main St	Vancouver	British Columbia
11	Kent Garbett		125	Milross Av	Vancouver	British Columbia
12	Ludivina Rego	Shana Cerda	2520	Manitoba St	Vancouver	British Columbia
13	Gerald Woodbu...		1277	Melville St	Vancouver	British Columbia
14	Marisa Marmol	???	1001	Lagoon Drive	Vancouver	British Columbia
15	Everette Cavazos		565	Smithe St	Vancouver	British Columbia
16	Galen Hinnenk...		729	Campbell Av	Vancouver	British Columbia

在上面的示例中，如果Data Inspector此时显示 字符，则数据在写入时会被破坏。这似乎就是这种情况。

另一项检查是在文本编辑器中打开数据。由于显而易见的原因，不可能对每个数据集（例如二进制文件或数据库）执行此操作，但对于基于文本的文件，它可以提供此时数据是否正确的明确证据：

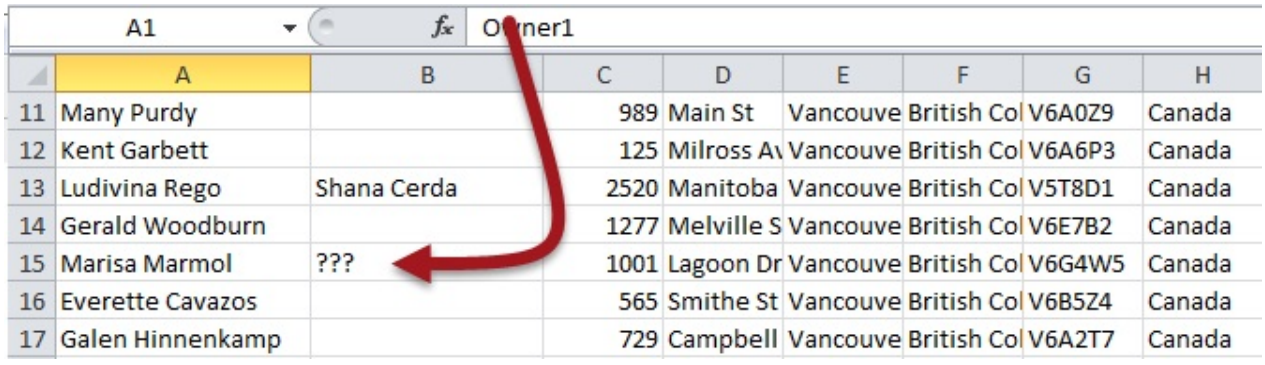


对于我们的示例，这确认了在写入数据时出现问题。但是，情况可能并非总是如此。

检查另一个应用程序中的输出数据集

如果FME（和文本编辑器）可以显示输出数据，那么可能是预期的应用程序没有正确地解释数据。

因此，打开要在其中使用的应用程序中的输出数据集。如果FME可以正确读取数据，并且在文本编辑器中看起来正确，则问题更可能出在最终应用程序如何解释数据：

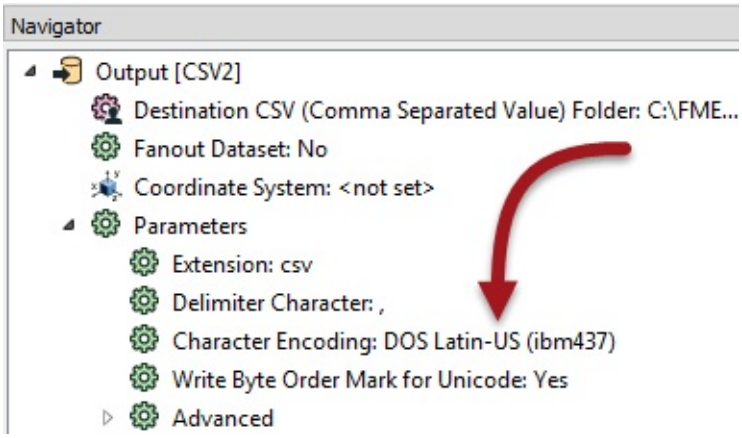


	A	B	C	D	E	F	G	H
11	Many Purdy		989	Main St	Vancouve	British Col	V6A0Z9	Canada
12	Kent Garbett		125	Milross Av	Vancouve	British Col	V6A6P3	Canada
13	Ludivina Rego	Shana Cerda	2520	Manitoba	Vancouve	British Col	V5T8D1	Canada
14	Gerald Woodburn		1277	Melville S	Vancouve	British Col	V6E7B2	Canada
15	Marisa Marmol	???	1001	Lagoon Dr	Vancouve	British Col	V6G4W5	Canada
16	Everette Cavazos		565	Smithe St	Vancouve	British Col	V6B5Z4	Canada
17	Galen Hinnenkamp		729	Campbell	Vancouve	British Col	V6A2T7	Canada

如果格式对于该应用程序是非原生的，那将尤其如此; 例如，在Esri产品之外读取地理数据库。

所有这些技术都缩小了可能发生错误的位置，但并不总是指定原因。例如，不正确的输出可能意味着FME在该写模块中具有限制，或者工作空间作者设置了不正确的参数。或者，也许一个应用程序使用不同的默认编码。

在这种情况下，知道写模块有问题，我们可以检查写模块参数并找到一个似乎设置错误编码的参数：



简而言之，这些技术首先确定了调查的位置，但不会自己提供绝对的答案。

技巧
编码对我们来说是一个很好的例子，但它也是一个你应该检查你的计算机是否能够查看这些数据的例子！如果您的计算机设置在错误的地区，那么您甚至可能无法使用它。

练习：调试工作空间

练习**3**	调试工作空间
数据	地址（Esri地理数据库） 犯罪数据（CSV - 逗号分隔值） 公园（MapInfo TAB） 游泳池（OSM - OpenStreetMap）
总体目标	参与温哥华可行走性项目
演示	调试最佳实践
启动工作空间	C:\FMEData2018\Workspaces\DesktopBasic\BestPractice-Ex3-Begin.fmwt
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\BestPractice-Ex3-Complete.fmwt

继续上一个练习，您已被分配到一个项目，以计算温哥华市每个地址的“步行性”。步行性衡量步行访问当地设施的难易程度。该工作空间目前正在评估犯罪，公园和噪音控制区域，但并未对步行性进行全面衡量。

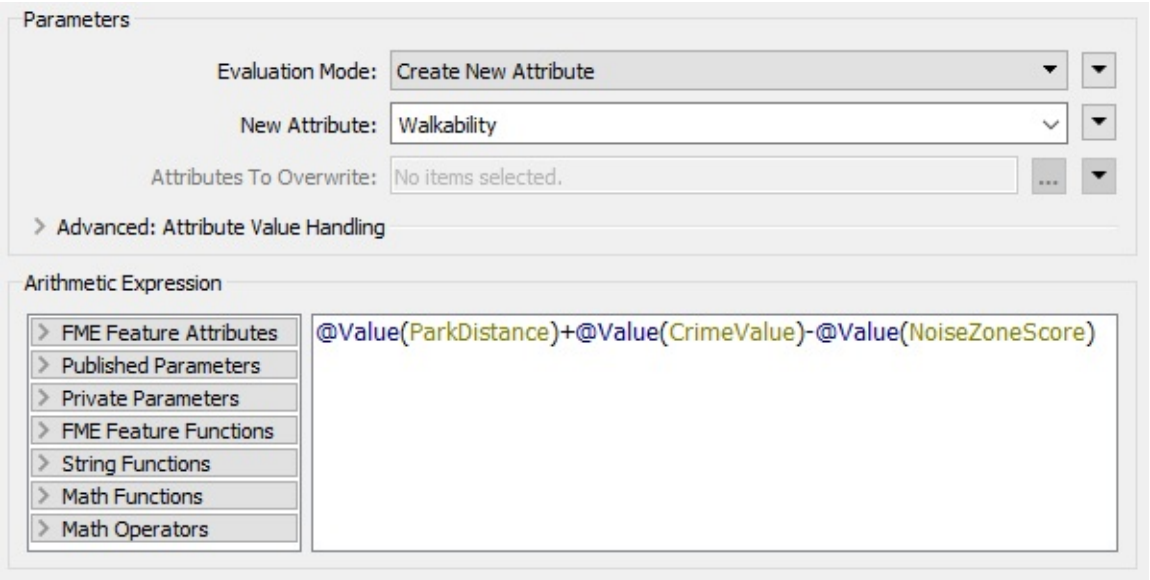
所以，让我们这样做，然后看看是否还有其他方面可以包括。

1）添加**ExpressionEvaluator**
我们可以创建一个walkability度量，它使用ExpressionEvaluator转换器组合所有当前值。

因此，将ExpressionEvaluator转换器添加到工作空间的末尾。

检查其参数。将其设置为创建一个名为Walkability的新属性，该属性为：

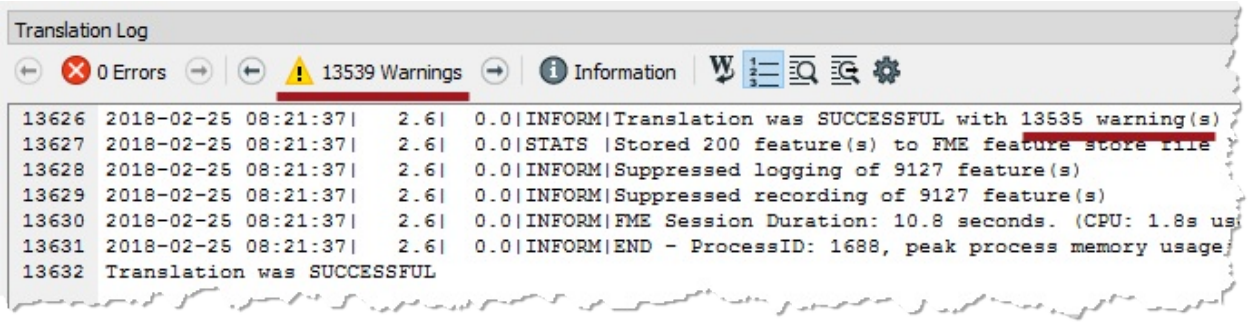
```
ParkDistance + CrimeValue  -  NoiseZoneScore
```



使用此表达式，结果越小越好。在ExpressionEvaluator上使用Run from This运行工作空间。

2）评估结果
让我们评估转换结果是否正确。

首先检查日志窗口是否有错误和警告。没有错误，但有超过13,000个警告，这不是一个好兆头：



警告说：

```
ExpressionEvaluator：无法计算表达式'@ real64（560.3272250455418+ <null> -0）'。
结果设置为空
```

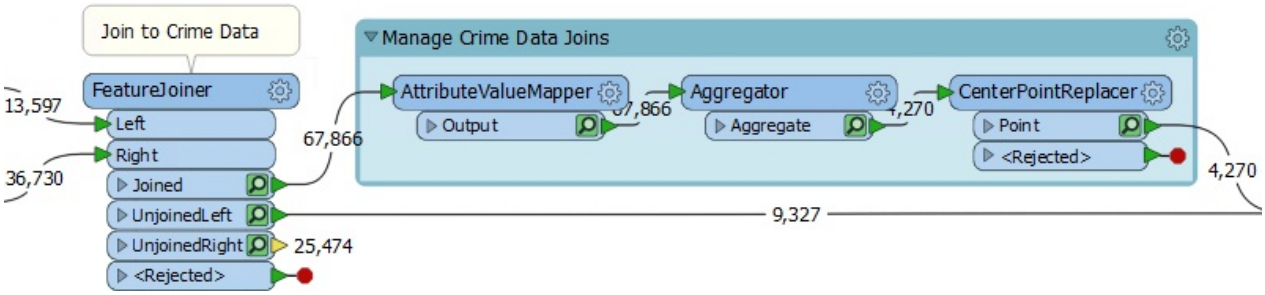
检查输出缓存，某些地址确实具有<null>的Walkability值。

所以我们知道存在问题，让我们试着弄清楚问题出在哪里以及问题发生的原因。

3）定位问题
我们可以告诉ExpressionEvaluator发出警告，但这并不一定意味着问题所在。计算失败，因为中间值为<null>。如果表达式是：

```
ParkDistance + CrimeValue  -  NoiseZoneScore
```

然后我们知道必须是CrimeValue才是问题。展开执行犯罪计算的书签。然后在FeatureJoiner转换器上执行Run From This，以便我们在书签中为转换器提供缓存，并为所有连接提供要素计数：



我们知道这里的一些要素正在获得<null>结果。首先检查AttributeValueMapper的缓存。这就是设置值的地方，所以也许那里出现了空值？

技巧
一个有用的测试是右键单击表视图窗口中的CrimeValue，然后按升序排序。这将把任何空值放在表的顶部。

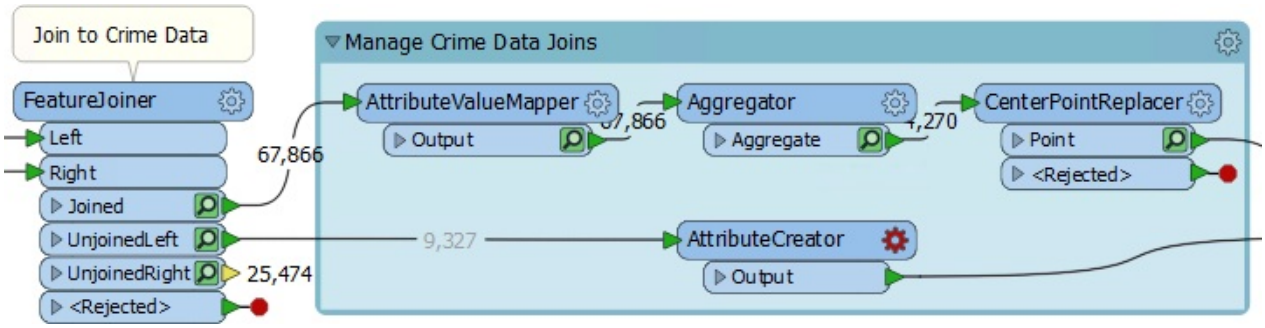
检查时，我发现其中没有CrimeValue属性的<空>值。Aggregator和CenterPointReplacer缓存都没有空值。

因此，请检查每个连接的要素计数。有4,270个要素标记有犯罪，9,327个要素没有。这总共给出了13,597，这是正确的。

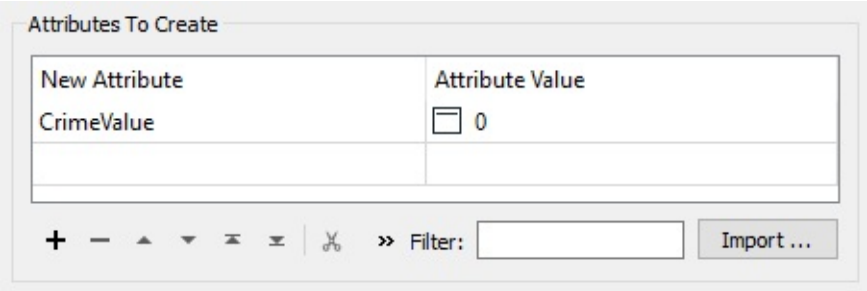
哦。你看到了吗？没有标记犯罪的9,327个要素：他们获得了什么CrimeValue？检查FeatureJoiner的UnjoinedLeft输出，您将看到它们没有值。这就是ExpressionEvaluator说存在空值的原因。

技巧
为了确认这一点，我将日志复制到文本编辑器中并搜索短语“ExpressionEvaluator：无法评估表达式”。 它出现了9,327次，与退出UnjoinedLeft端口的要素数量相同。巧合？

4）修复问题
如果这些要素没有CrimeValue属性，那么我们应该给它们一个。为此，请将AttributeCreator转换器添加到工作空间，并展开“管理犯罪数据连接”书签以将其括起来：



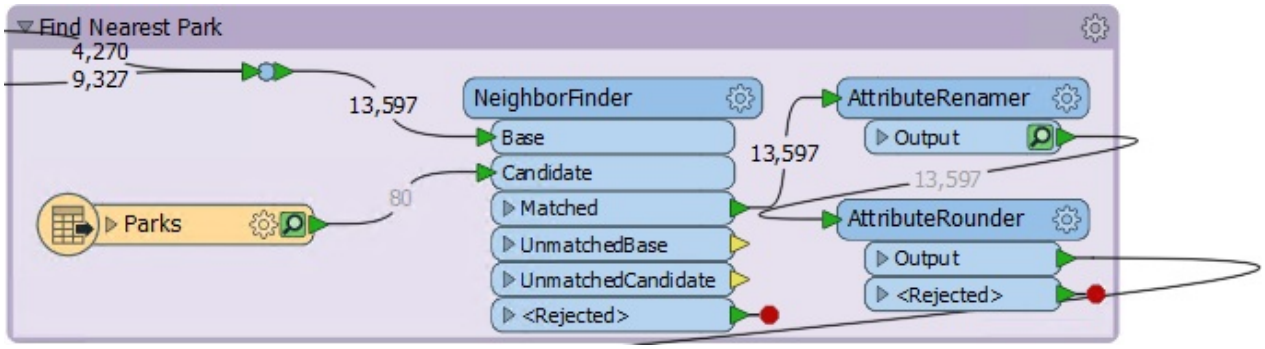
打开其参数并创建一个名为CrimeValue的属性，其值为零（0）。



折叠书签并重新运行工作空间（使用ExpressionEvaluator上的“运行到此Run to This”）。您现在应该发现警告更少，输出中不包含<null空>值。

5）四舍五入属性
数据中的一个问题是结果被测量到多个小数位。此步骤不是必需的。快速浏览一下数据就会发现，ParkDistance结果才是罪魁祸首。

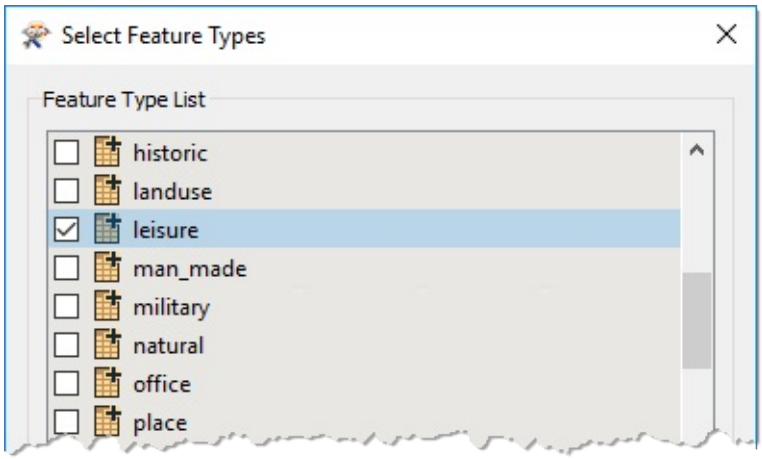
因此，找到ParkDistance值的创建位置，并添加一个AttributeRounder转换器，将其四舍五入到零位：



6）添加游泳池
由于我们现在能够计算步行性值，因此请将游泳池纳入计算中。为此，请添加以下内容：

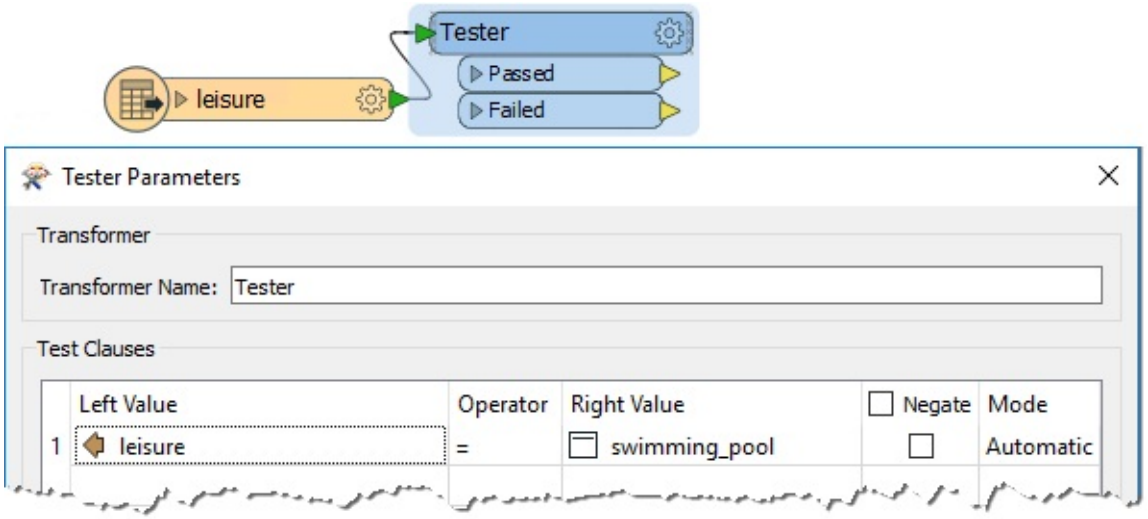
读模块格式	**OpenStreetMap（OSM）XML**
读模块数据集	C:\FMEData2018\Data\OpenStreetMap\leisure.osm

出现提示时，仅选择leisure要素类型：



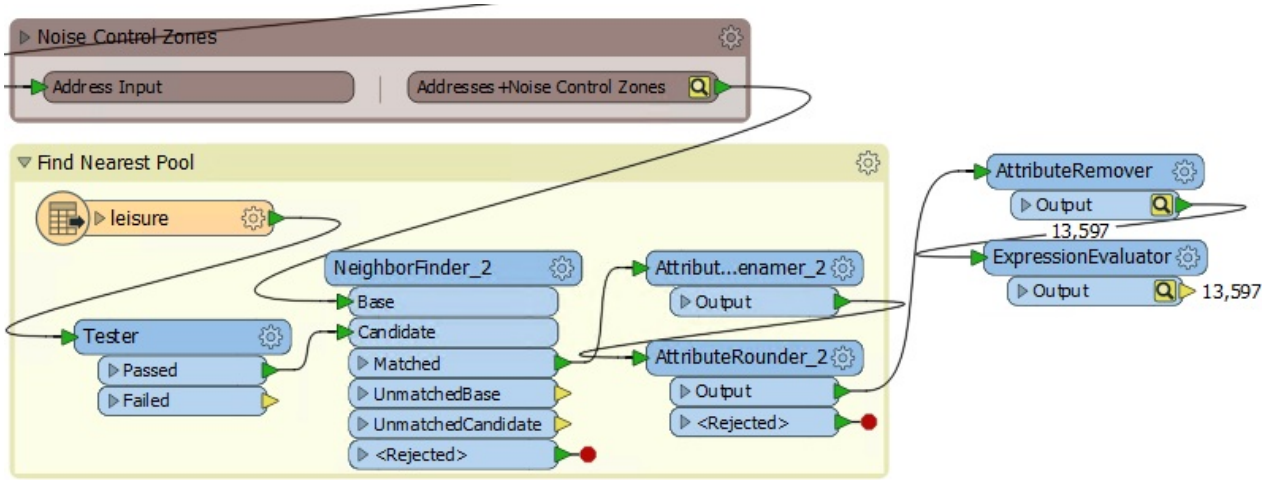
7）过滤休闲（**Leisure**）数据
如果您检查休闲数据，您会注意到有各种类型的休闲设施，其类型记录在休闲*leisure*属性中。

因此，设置一个Tester转换器来测试leisure = swimming_pool



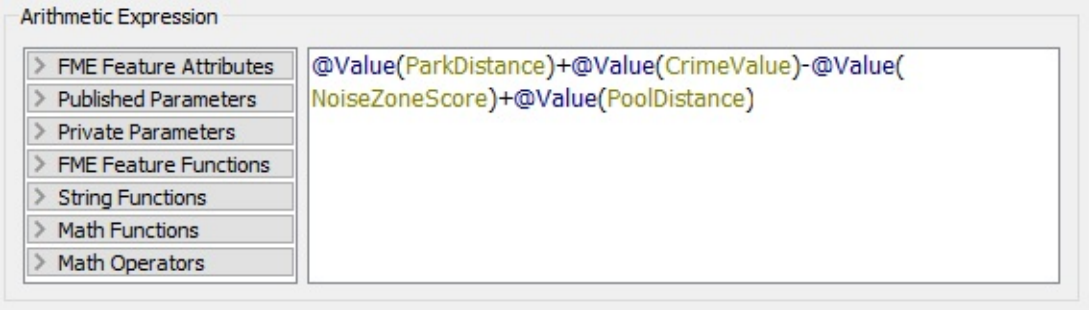
8）找到最近的游泳池
找到最近的游泳池的技术与找到最近公园的技术相同。因此，展开最近的公园书签，并从里面复制/粘贴转换器。

在它周围放一个书签，并以与park部分相同的方式连接它：



检查新粘贴的转换器的参数。NeighborFinder转换器没有任何需要设置，但AttributeRenamer和AttributeRounder需要使用PoolDistance而不是ParkDistance。

9）更新**ExpressionEvaluator**
现在只需更新ExpressionEvaluator以考虑新的PoolDistance属性：



重新运行工作空间。检查日志中是否有警告和错误，然后检查ExpressionEvaluator缓存。

请注意，由于PoolDistance，可步行性得分突然非常大。出了点问题，但是什么？

10）找到问题
PoolDistance是问题的根源。没有相关的日志消息提供线索，并且要素计数数字看起来正确。

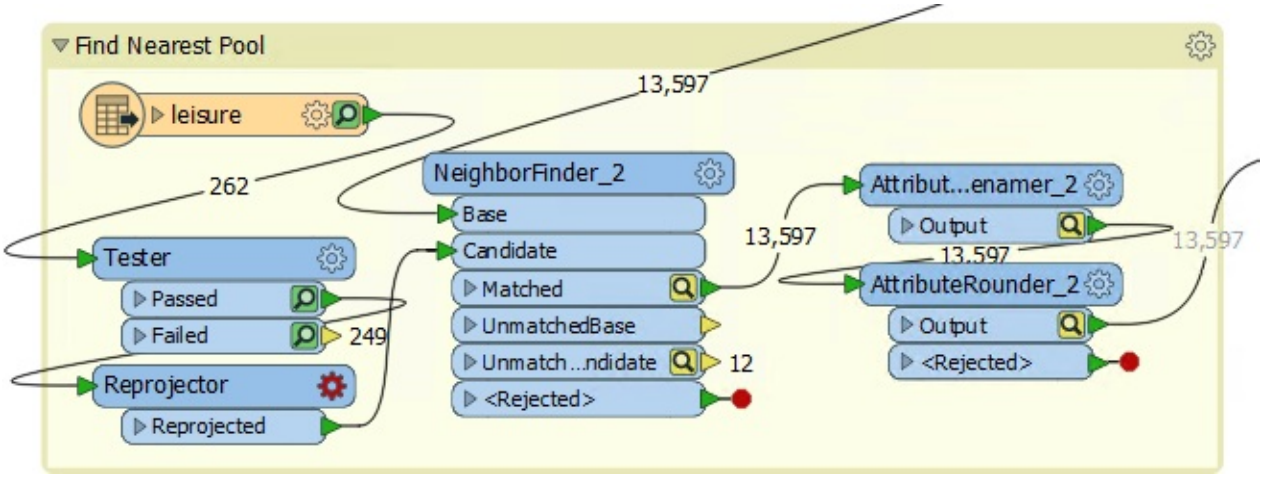
因此，右键单击“查找最近的泳池”书签，然后选择“选择书签中的所有对象”。现在按Ctrl + I检查所选对象的缓存。

Data Inspector中的显示窗口显示两个数据小斑点，相距很远。该结果是坐标系不匹配的典型结果。

查询一些要素，您将看到主数据的坐标系为UTM83-10，而OSM的休闲数据的坐标系为LL84。

这种差异是为什么每个地址的“最近”泳池是如此高的距离。

11）修复坐标系问题
显而易见的解决方案是将池重新投影到正确的坐标系。因此，添加一个Reprojector转换器，以便在它到达NeighborFinder之前重新显示休闲数据：



检查其参数并将其设置为从LL84重新投影到UTM83-10

折叠所有书签并重新运行工作空间的相应部分。检查日志窗口并检查ExpressionEvaluator缓存。

现在每个地址都有一个步行分数，数字越低越好，数字越高越差。

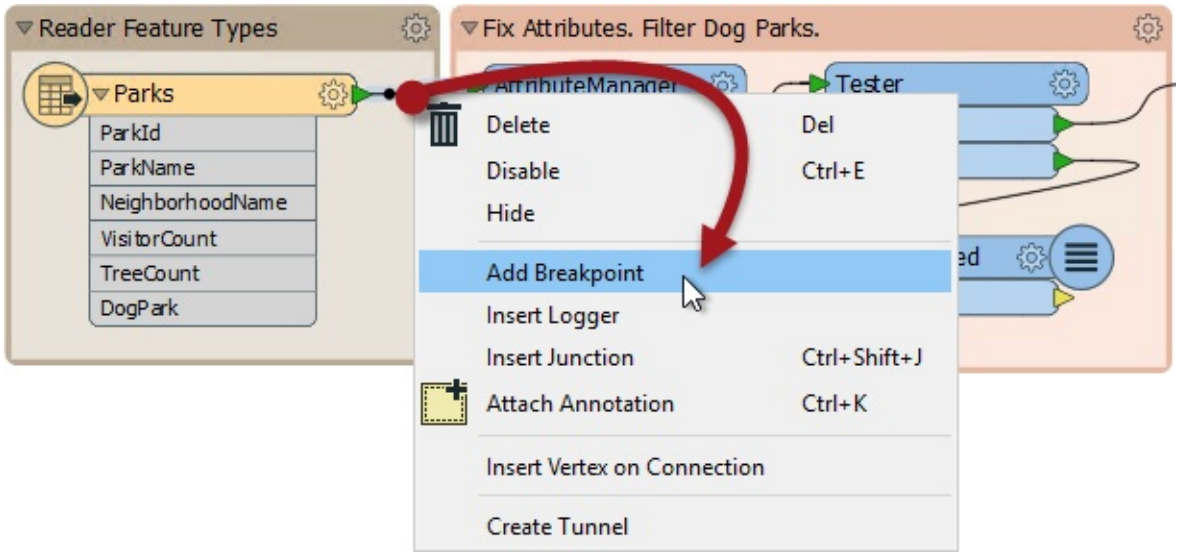
恭喜
通过完成本练习，您已学会如何： • 使用ExpressionEvaluator转换器 • 检查日志窗口是否有错误和警告 • 通过使用要素计数和数据检查来查找问题 • 识别并修复工作空间中的问题

要素调试

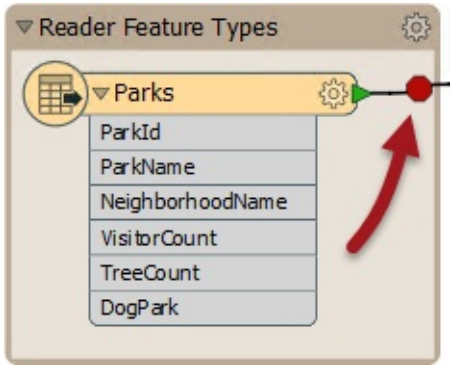
要素调试是一种允许在翻译期间检查各个要素的工具。它不同于检查特定位置的数据，因为它一次检查一个要素，并允许作者通过工作空间跟踪该要素的进度。

当一个问题被确定为在转换期间，但是失败点未知时，这是最有用的。

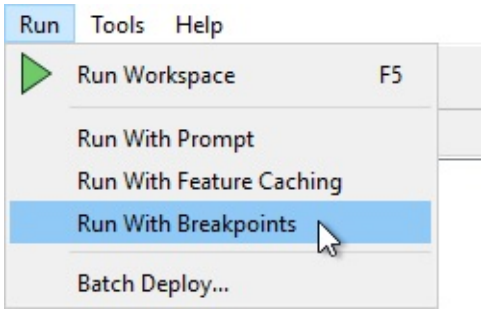
要素调试由“断点”触发; 工作空间连接，由用户标记为应检查要素的位置：



连接以深黑色突出显示，带有红色“停止”符号，表示其新状态：

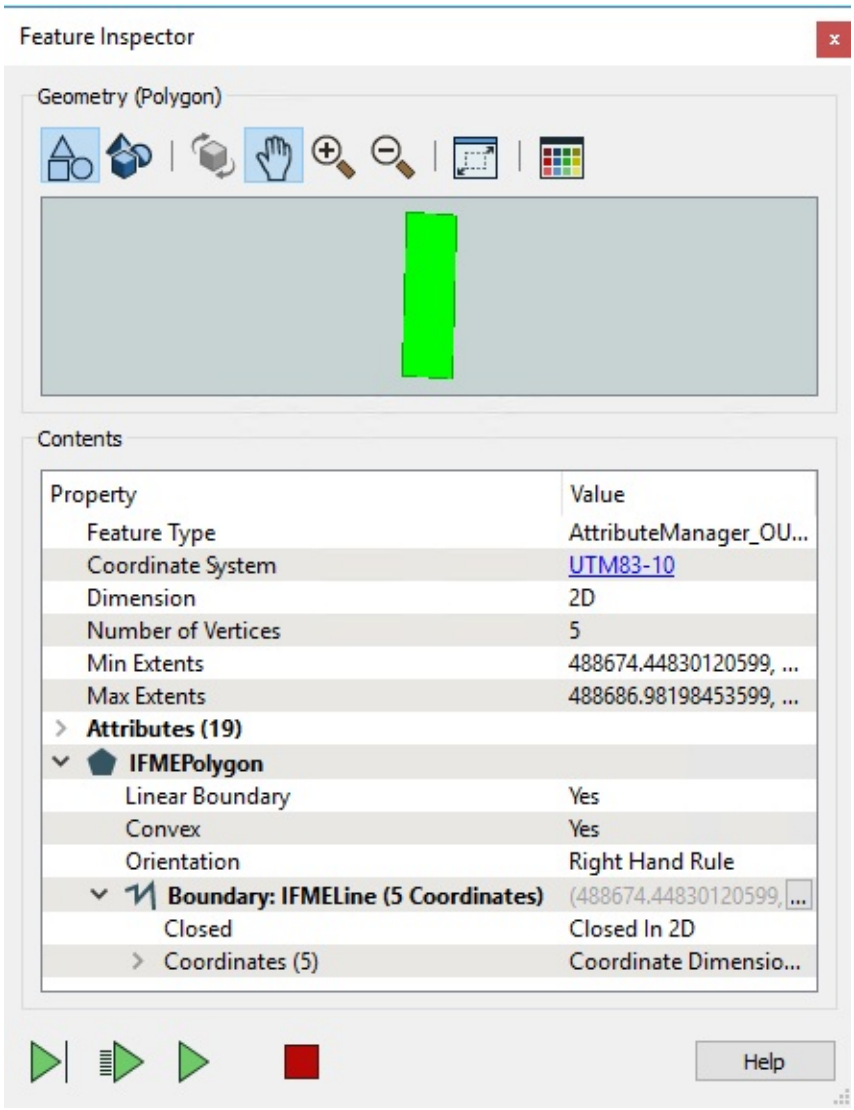


现在，工作空间运行时打开了“Run with Breakpoints”选项：



当第一个要素到达断点时，临时暂停转换，并在“要素检查器”窗口中显示有关该要素的信息。

窗口的上半部分显示了该要素的图形表示; 下半部分列出了要素类型和坐标系等属性; 加上属性和几何对象信息。



Feature Inspector窗口底部有四个按钮：

按键	操作	描述
	进入到下一个连接	此工具一次一个转换器，一步步通过工作空间，显示处理时的要素状态。
	进入到下一个断点	此工具重新启动转换，在下次要素到达检查点时停止。
	继续转换	此工具重新启动转换，忽略所有其他断点。
	停止转换	此工具会停止转换。

当前活动的连接以红色突出显示，表示它是当前暂停转换的位置。

模块复习

该模块旨在帮助您有效地使用FME Workbench，有效管理FME相关项目，并确保这些项目具有可扩展性和可移植性。

您应该从这个模块中学到什么

以下是本次课程要学习的重点：

理论

- 最佳实践是以有效的方式使用FME的行为，但其他FME用户也很容易理解。
- FME的最佳实践可分为样式，方法和调试。
- 样式是一种实践，可以更轻松地导航和理解现有工作空间。
- 方法是一种实践，它定义了用于维护和性能目的的最优转换器。
- 调试是在工作空间中定位和修复缺陷的行为。

FME技能

- 能够使用书签和注释创建精心设计的工作空间样式。
- 能够创建更易于维护，编辑和扩展的工作空间。
- 能够创建高效执行的工作空间。
- 解析FME日志文件的能力。
- 检查输出和要素计数以查找错误的能力。
- 能够使用要素调试来跟踪各个要素。

进一步阅读

如需进一步阅读，为什么不在我们的博客上浏览 标记为**Best Practice**的文章?

Q + A答案

以下是本章问题的答案。

Vector小姐说...

以下哪项不是创建书签的方法？

1.单击工具栏上的“插入书签”按钮
2.选择一个转换器，右键单击，选择“创建书签”
3.选择多个转换器，右键单击，选择“创建书签”
4.使用Ctrl + B快捷键

Vector小姐说...

除了连接之外，还可以禁用其他对象。你可以在Workbench中找出哪些对象（可能有多个）可以被禁用？

1. 转换器
2. 要素类
3. 注释
4. 书签

OK, 从技术上讲，您无法禁用书签，但您可以选择禁用其中的所有对象。

练习：最短路线黑客马拉松

练习**4**	**FME**黑客马拉松
数据	道路（Autodesk AutoCAD DWG和/或PostGIS）
总体目标	找到从黑客马拉松到意大利咖啡馆的最短路线
演示	数据转换，变换和最佳实践
启动工作空间	无
结束工作空间	C:\FMEData2018\Workspaces\DesktopBasic\BestPractice-Ex4-Complete.fmw

一个区域GIS小组正在举办FME Hackathon，并且您已被邀请参加。

您已获得一组源数据，并被要求从中创建有用的项目。您决定制作制作一个工具来绘制出从黑客马拉松场地到一家咖啡厅的路线，在那里有个团体聚会将在当晚举行，这将是有趣的。

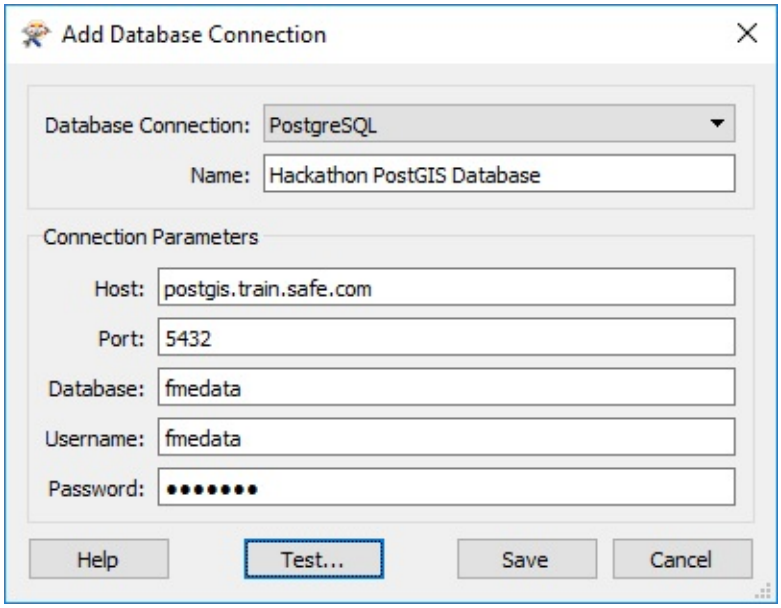
因此，您的任务是使用可用的数据来计算从会议中心到咖啡馆的最佳路线，并将该数据写出为GPX格式，以便民众可以在他们的GPS /移动设备中使用它。

1）创建数据库连接
源数据已在PostGIS数据库中提供; 因此，我们的首要任务应该是创建一个连接。这样我们就可以使用连接而不必输入连接参数。

在Web浏览器中访问http://fme.ly/database - 这将显示在Amazon RDS上运行的PostGIS数据库的参数。

在Workbench中，从菜单栏中选择工具> FME选项。

单击“数据库连接”类别的图标，然后单击[+]按钮以创建新连接。在“添加数据库连接”对话框中，首先选择PostgreSQL作为数据库类型。然后输入通过Web浏览器获取的连接参数。



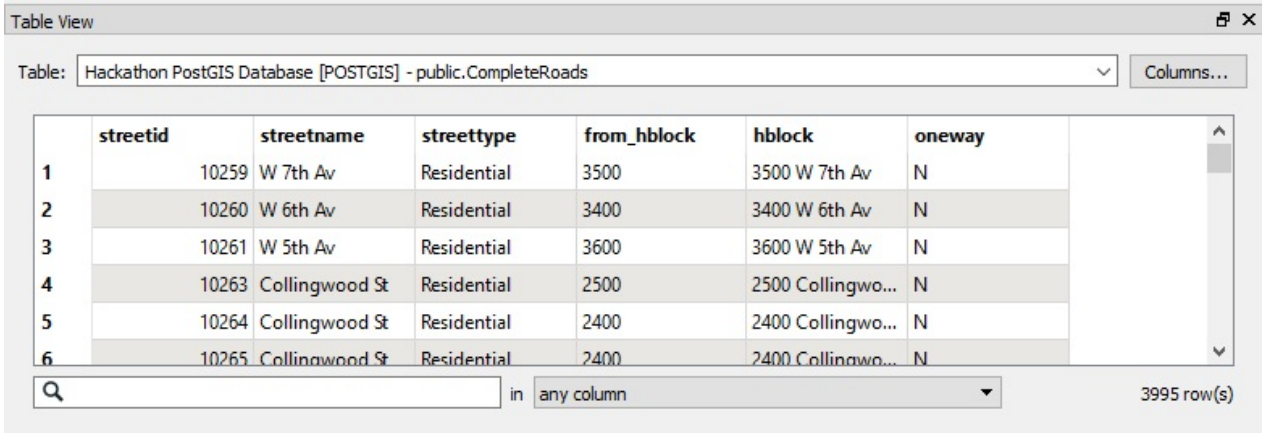
为连接指定名称，然后单击“保存”。然后单击“确定”关闭“FME选项”对话框。

技巧
本练习的已完成工作空间使用名为**Hackathon PostGIS**数据库 的数据库连接 如果要打开/使用此工作空间，则应使用相同名称创建连接。这样，当您打开工作空间时，它将自动找到匹配的连接。 这很好地说明了数据库（和Web）连接的命名约定的重要性。

2）检查数据
启动FME Data Inspector以检查我们将使用的数据集。选择“文件”>“打开数据集”，然后在出现提示时输入以下内容：

读模块格式	**PostGIS**的
读模块数据集	Hackathon PostGIS数据库
参数	表格列表：public.CompleteRoads

单击“确定”关闭对话框并打开数据。您将看到一组道路要素，每个道路要素都有一组属性。一个属性指定要素是否表示单行道。如果我们想要计算一条实际合法的路线，这一点很重要！



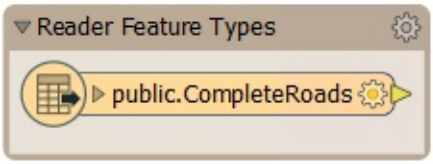
注意：如果您在使用*PostGIS*数据库时遇到任何问题 - 例如防火墙的连接问题 - 那么以下*AutoCAD*数据集可以替换，只需要很少的更改:

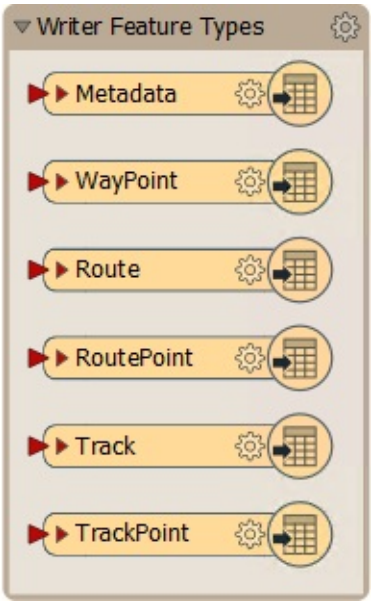
读模块格式	**Autodesk AutoCAD DWG / DXF**
读模块数据集	C:\FMEData2018\Data\Transportation\CompleteRoads.dwg

3）启动**Workbench**
启动Workbench并使用选项创建工作空间。

读模块格式	**PostGIS**
读模块数据集	Hackathon PostGIS数据库
参数	表格列表：public.CompleteRoads
写模块格式	GPS交换格式（GPX）
写模块数据集	C:\FMEData2018\Output\Training\Route.gpx

工作空间将如下所示：





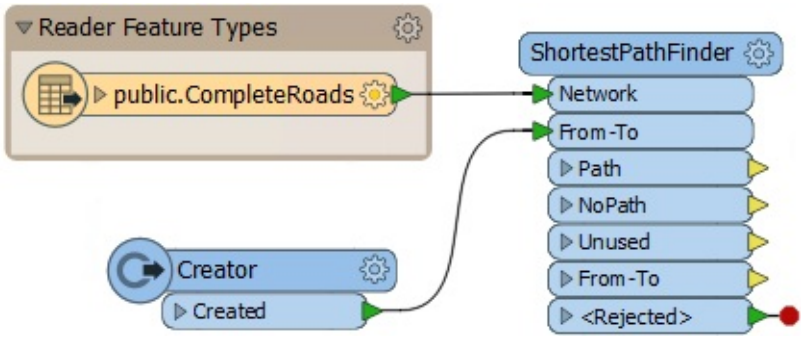
请记住，GPX是一种固定模式格式，因此会自动创建六种不同的写模块要素类型。

4）添加**ShortestPathFinder**
现在我们需要开始计算路线。显而易见的第一步是添加ShortestPathFinder转换器，这是我们计算路线的方法。

因此，添加一个ShortestPathFinder转换器。将public.CompleteRoads连接到Network端口。

5）添加**Creator**
ShortestPathFinder上的另一个输入端口用于From-To路径（我们旅程的起点和终点）。有很多方法可以创建它 - 甚至接受来自Web地图的输入 - 但是在这里我们将使用Creator转换器手动创建一个要素。

因此，添加一个Creator转换器并将其连接到From-To端口：

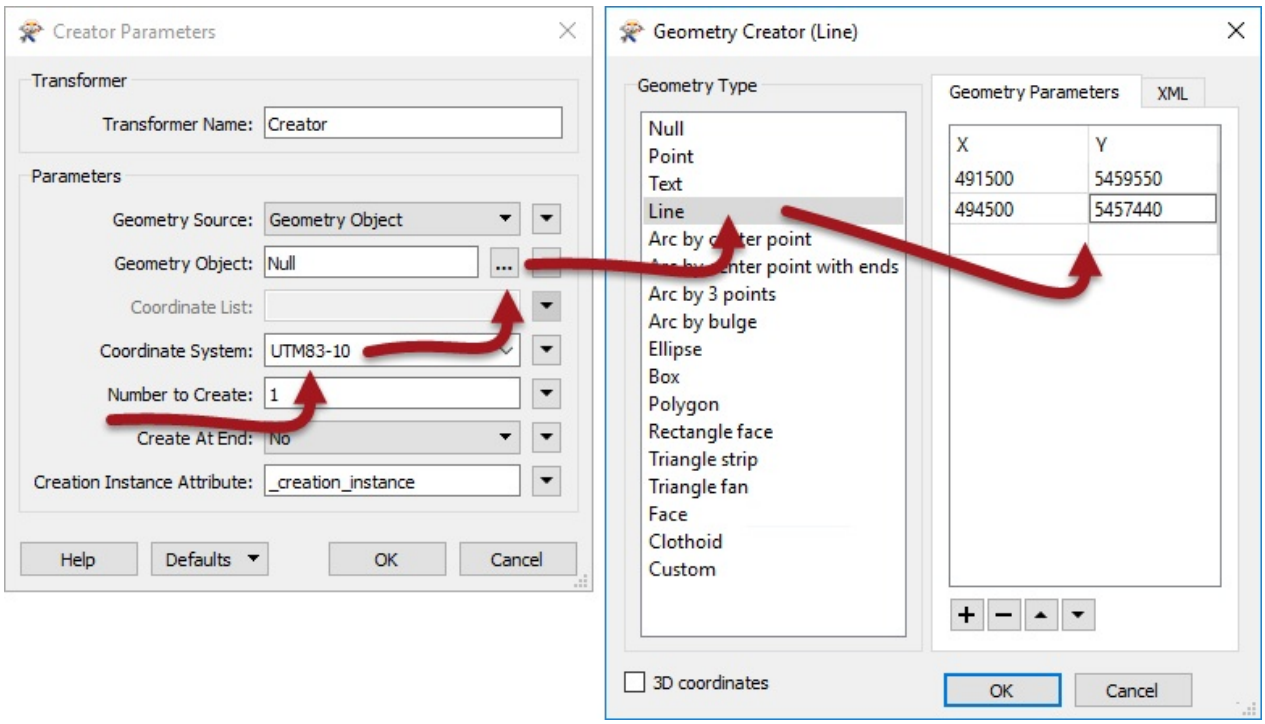


检查Creator的参数。

首先输入UTM83-10作为我们即将创建的数据的坐标系。对于Geometry Object参数，单击右侧的[...]浏览按钮以打开几何创建对话框。选择Line作为几何类型并输入以下坐标：

```
X 491500 Y 5459550
X 494500 Y 5457440
```

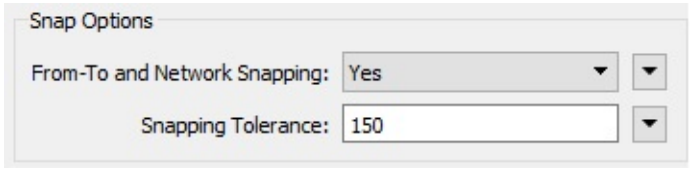
第一个坐标是黑客马拉松场地的坐标，第二个坐标是我们网络中最接近我们要去的咖啡馆的点。



单击“确定”按钮以确认更改。

6）检查**ShortestPathFinder**参数
我们添加的要素的坐标可能不完全位于道路网络上。为了解决这个问题，我们可以在ShortestPathFinder中使用参数。

因此，请检查ShortestPathFinder参数。在Snap Options下，将**From-To**和**Network Snapping**设置为*Yes*，并输入*150*作为**Snapping Tolerance**捕捉容差：



另外，请注意网络成本的参数 - 我们稍后会使用它们。

7）运行工作空间
确保打开要素缓存并运行工作空间。检查日志，然后检查ShortestPathFinder：Path缓存。如果一切顺利，输出将如下所示，并定义了一条路线：

地图瓷砖由Stamen Design，许可由CC-BY-3.0。数据由OpenStreetMap，许可由CC-BY-SA。

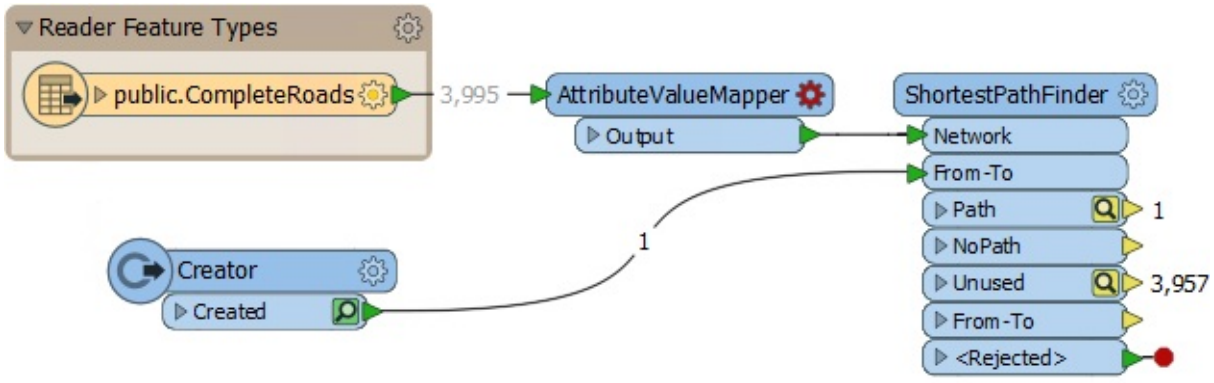
当然，如果一切还没有正确进行，您必须使用您的调试技能来尝试和查找错误!

8）添加AttributeValueMapper

结果看起来很好，但有些事情我们仍然不确定：例如，如果这条线路使用的是较慢的住宅区道路怎么办?我们可以通过对每个道路要素应用不同的成本来强制路线更喜欢干线路线。

成本取决于道路类型。本质上，我们将道路类型映射到成本，而使用AttributeValueMapper转换器的方法是这样做。

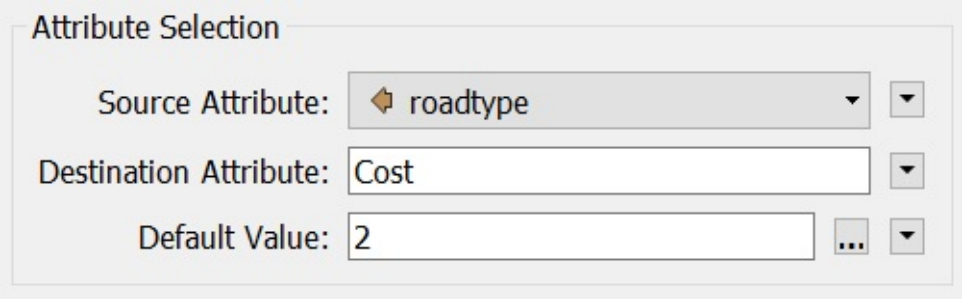
因此，在CompleteRoads要素类型和ShortestPathFinder:Network端口之间向工作空间添加AttributeValueMapper转换器：



9）编辑AttributeValueMapper

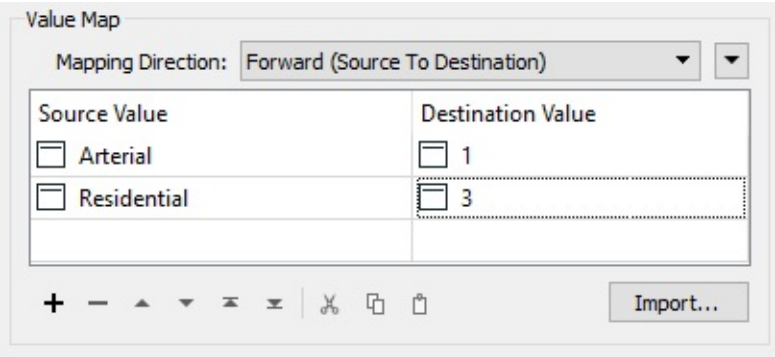
检查AttributeValueMapper的参数。输入以下值：

源属性	**roadtype**
目标属性	Cost
默认值	2



现在，在这些参数下面，我们将映射一些数据。

源值	目标值
干路	1
住宅	3



如果路线是干路（主要道路），它将获得1的成本，住宅路线将获得3的成本，而所有其他类型将获得2的成本（因为这是默认值）。单击“接受/确定”以确认参数。

10）应用成本
现在我们必须应用我们刚刚创建的成本。检查ShortestPathFinder的参数。输入以下值：

成本类型	由两个属性
正向成本属性	成本
逆向成本属性	成本

技巧
为什么“两个属性”？那是因为只有正向成本，我只能沿着与顶点排列方向相同的方向沿着一段道路行进。因为我不想根据它们的顶点方向避开道路，所以使用两个属性告诉FME两个方向的成本是相同的。

现在重新运行工作空间以查看结果是否有任何差异。它应该如下所示：

地图瓷砖由Stamen Design，许可由CC-BY-3.0。数据由OpenStreetMap，许可由CC-BY-SA。

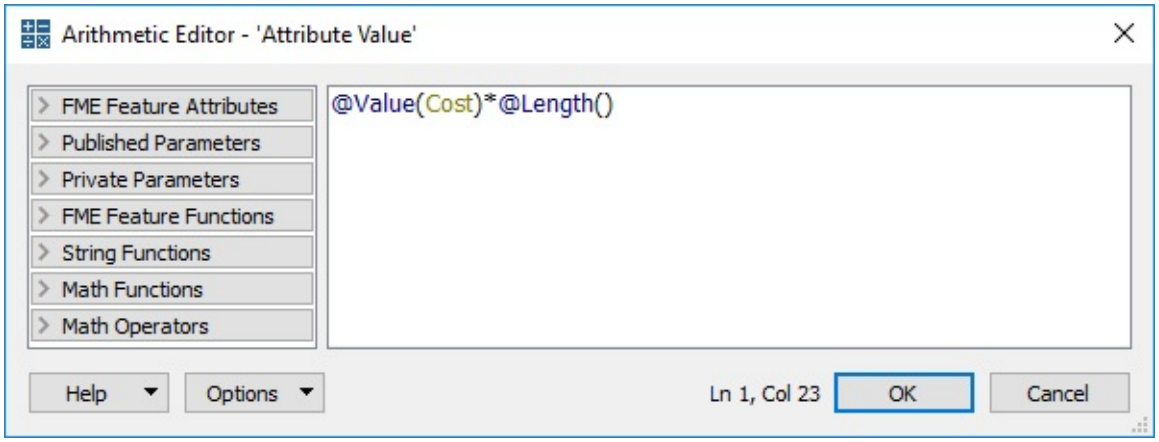
因此，成本加权已经产生了影响。但是这个结果有问题.....

> 分析师女士说**.....**
>
> 这次路线走的是更长的路径，我可以看出原因可能是：成本被用来衡量路线来取代距离，而不是两者都有。 为了解释这一点，假设我想从A行进到B。有一个住宅道路要素从A开始到B结束，路线距离为1.5公里。 还有一条主干道路的要素，从A开始，到B结束。相当极端，它环绕着月球的黑暗面，路线距离为768,000公里。 目前我们的解决方案将选择768,000公里的行程，因为与“3”的住宅路线成本相比，它的成本为“1”！ 显然，这里的练习数据远没有这么荒谬，但同样明显的是，在将距离考虑回结果之前，路线可能不是最佳的。

11）添加**AttributeManager**
在AttributeValueMapper：Output端口和ShortestPathFinder：Network端口之间添加AttributeManager转换器并查看参数。

在“成本”属性的值字段中，单击下拉箭头并选择“打开算术编辑器”。在该对话框中输入表达式：

```
@Value(Cost)*@Length()
```



简而言之，我们现在将成本乘以道路长度，以便为我们提供综合权重。

重新运行转换，看看它是否有所作为：

地图瓷砖由Stamen Design，许可由CC-BY-3.0。数据由OpenStreetMap，许可由CC-BY-SA。

是的，它确实证明了这条路线比它可能的要长。当然，我们使用的表达式也是非常主观的，并且可以变得更复杂，以提供更好的结果（我们可以尝试使用对数标度来查看产生的结果）。

12）编辑AttributeManager

无论我们的表达式如何，还有最后一个问题需要解决。.....单行道。目前，我们没有解决方案来阻止我们驾驶错误的方向。

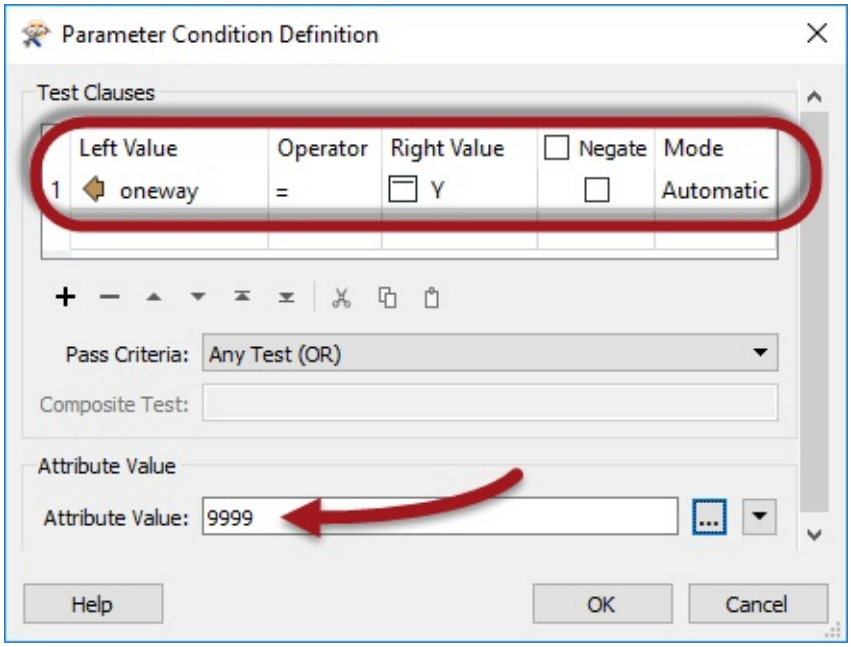
幸运的是，每条单行道都标有一个属性，其顶点按照允许的行程方向排序，因此我们知道要避免哪种方式。让我们用这些信息来解决这个问题。

我们现在需要为每个方向计算不同的成本属性，尽管只有涉及单行道时该值才会有所不同。像往常一样，有多种方法可以在FME中处理这个问题; 让我们适度轻松一点。

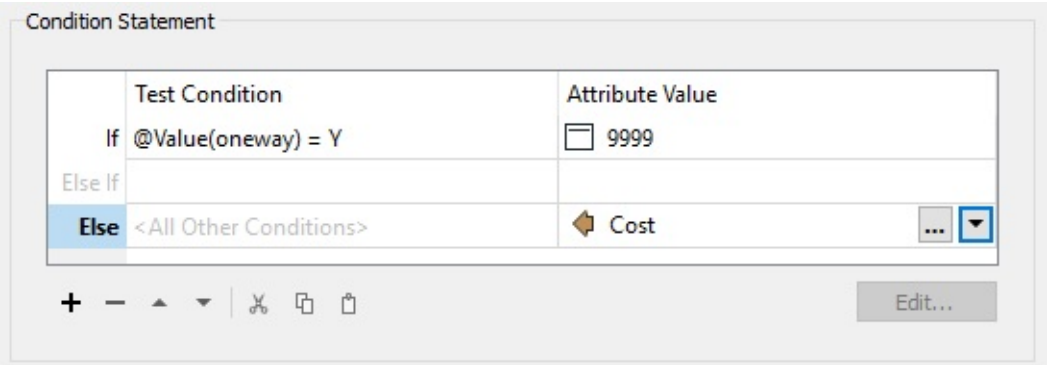
再次查看AttributeManager参数。这次创建一个名为ReverseCost的新输出属性。在值字段中，单击下拉箭头并选择“条件值”。

技巧
条件值是根据测试条件设置的值。这就像将Tester合并到AttributeManager中一样。这些内容在FME Desktop高级培训课程中有更详细的介绍。

在打开的定义对话框中，双击第一个“If”行并打开“测试条件”对话框。在这里设置一个测试oneway = Y。对于输出值（对话框的底部）输入值9999（即，错误的方式旅行的成本是非常昂贵的）！



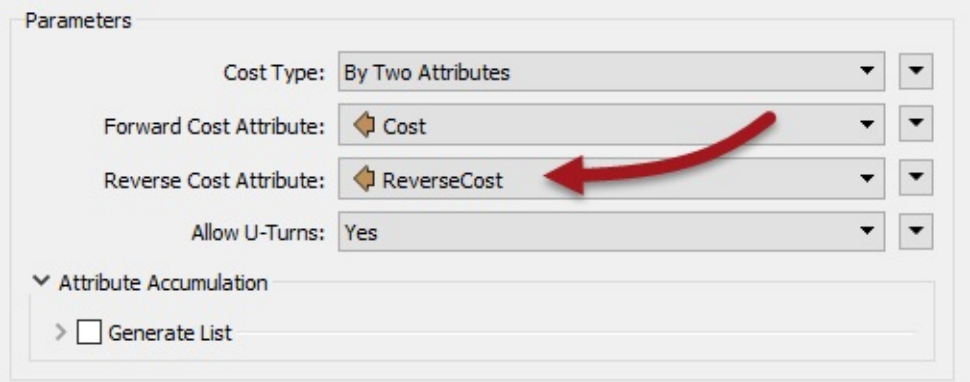
单击“确定”关闭该对话框。返回上一个对话框，双击显示<No Action>的位置，选择下拉箭头，然后选择Attribute Value> Cost：



再次单击“确定”两次以关闭这些对话框。

我们基本上做的是说如果这是一条单行道，设置一个过高的反向成本，否则只需设定通常的远期成本。

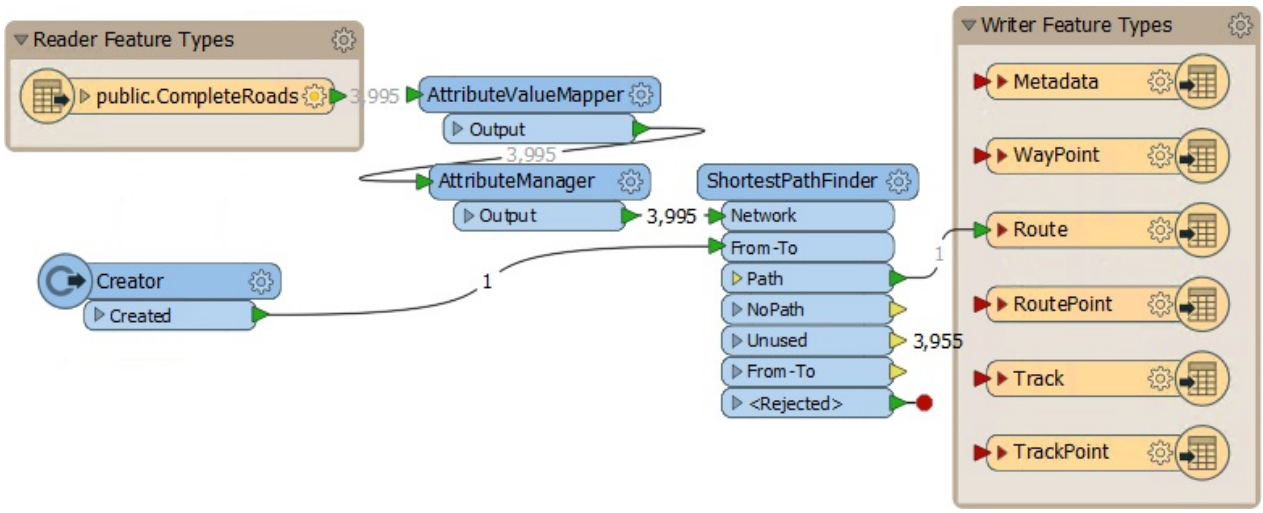
13）应用成本
最后一次更改。检查ShortestPathFinder参数并将ReverseCost属性从Cost更改为ReverseCost：



现在我们准备好了。重新运行工作空间并检查输出：

之前沿着一条单向（东西向）的街道错误行进（西 - 东）的路线至少会发生一次变化（上图）！

14）连接模式
哦！不要忘记关闭要素缓存并将路径端口连接到路径输出要素类型：



现在运行工作空间，将数据上传到您的GPS设备，您就可以开始了！

高级练习
不是真的高级，但你一直都在使用最佳实践，对吧?我的意思是，在需要的地方有书签和注释，没有重叠连接? 如果不是，那么你可能想要解决这个问题！

恭喜
通过完成本练习，您证明了您知道如何： • 创建并使用FME数据库连接 • 使用各种转换器创建原型FME工作空间 • 使用调试技术查找练习中遇到的任何问题 • 使用好的样式来开发工作空间

课程总结

虽然你的FME培训现在已经结束了，但是有大量的专家信息可供将来的帮助。

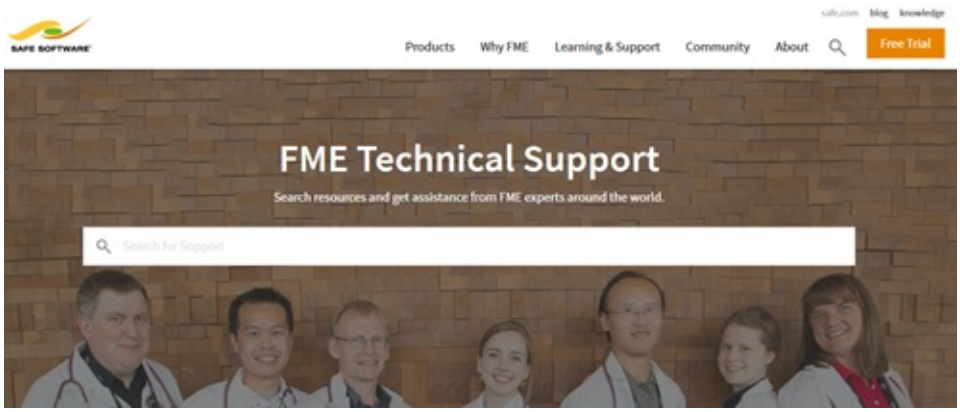
产品信息和资源

Safe Software网站

Safe Software网站 是FME所有东西的官方信息来源。它包括有关FME产品，Safe Software服务，FME解决方案，FME支持和安全软件本身的信息。

Safe支持团队

FME的背后是热情，有趣，知识渊博的专家，随时准备帮助您取得成功，并建立在知识转移原则基础上的支持团队理念。



您可以通过支持案例（网络/电子邮件）或使用在线聊天来请求产品支持。

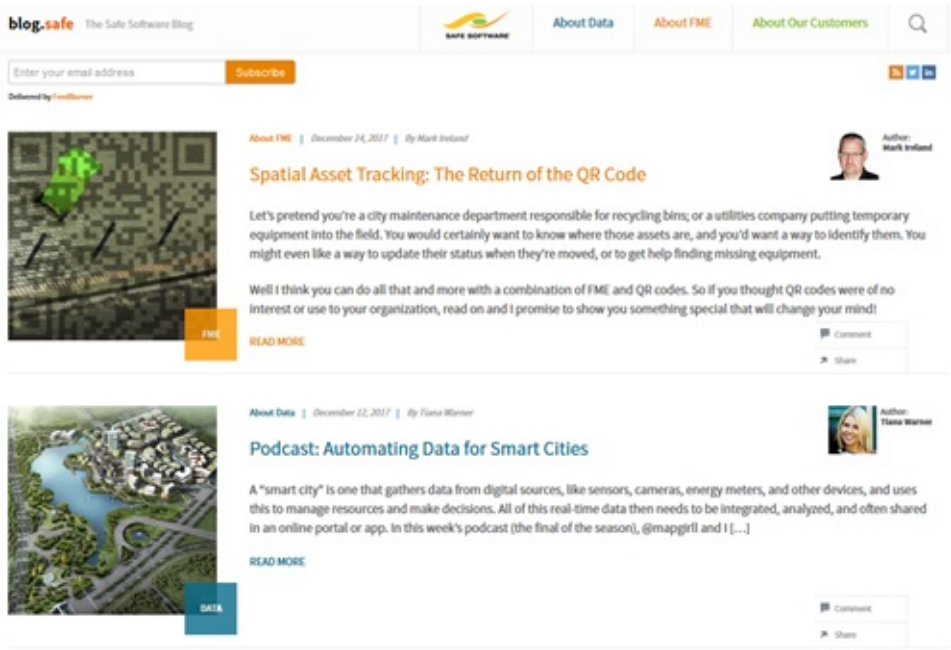
您的本地合作伙伴

Safe Software的合作伙伴和经销商遍布全球，为您所在地区和您的语言提供专业知识和服务。

您可以在Safe Software网站上找到官方合作伙伴列表。.

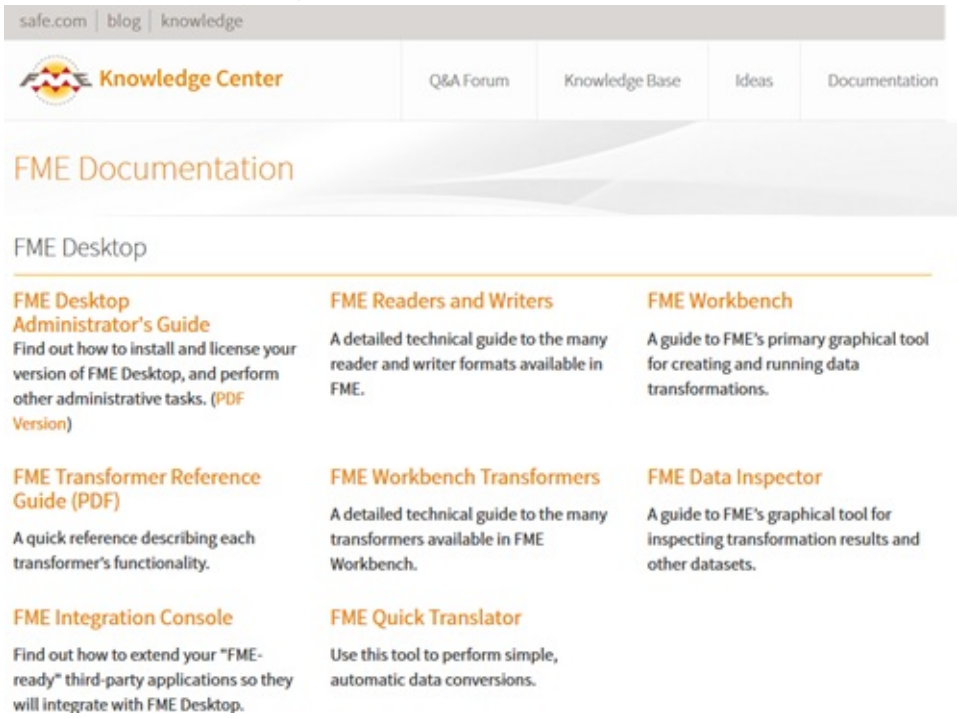
Safe Software博客

Safe Software博客提供有关FME技术信息，有关客户的使用情况的文章，以及空间数据互操作的总体思路。



FME手册和文档

使用FME Workbench中的“帮助”功能访问FME Desktop的帮助和其他文档。或者，在我们的网站上查看知识中心部分 。



社区信息和资源

Safe Software积极推动FME用户成为FME社区的一员。

FME知识中心

FME知识中心是我们的社区网站-所有社区资源的一站式商店，加上浏览文档和下载工具。

知识基地

FME知识基地包含大量信息; 包括提示，技巧，示例和常见问题解答。都有一些部分关于FME Desktop和FME Server，其中包含从安装和许可到最高级转换和变换任务等主题的文章。

问答论坛

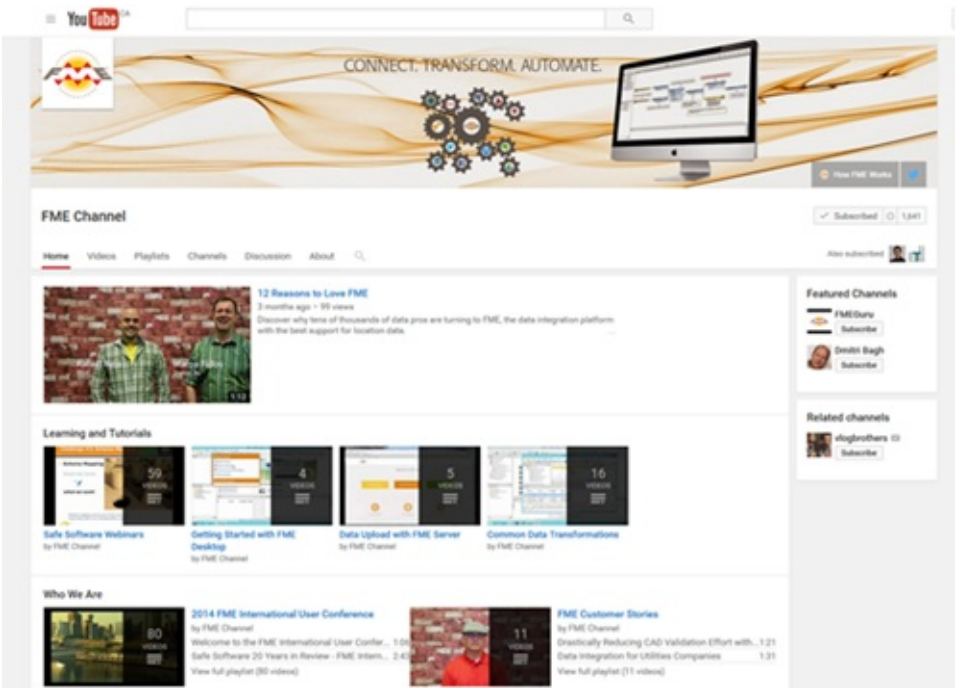
FME社区成员发布与FME相关的消息，提出问题并分享回答其他用户的问题。会员获得“声誉”和“徽章”，并且有顶级参与用户的排行榜。快来看看他们如何帮助您的FME项目！

想法交流

FME开发非常依赖于用户。Ideas 交流让用户有机会发布他们对新FME功能的想法，或对现有功能的改进，并允许每个人对提议的想法进行投票。想法获得的投票越多，实施的可能性就越大！

FME频道

此FME YouTube频道适用于只能通过截屏视频或电影正常欣赏的演示。除此之外，还有许多解释性和有用的电影，包括大多数培训和教程的录音。


CONNECT. TRANSFORM. AUTOMATE.
FME Channel
Home
Videos
Playlists
Channels
Discussion
About
12 Reasons to Love FME
Featured Channels
Related channels
Learning and Tutorials
Safe Software Webinars
Getting Started with FME Desktop
Data Upload with FME Server
Common Data Transformations
Who We Are
2014 FME International User Conference
FME Customer Stories

反馈和证书

该培训课程的格式经常受到以前课程的评论和反馈的影响。

课程反馈

Vector小姐说...
最后一组问题 - 这次你会告诉我答案是否正确！

Safe Software非常重视培训课程参与者的反馈意见，我们的反馈表可以告诉我们您对我们实现培训目标的满意程度。

您现在可以填写反馈表, 但在课程结束后不久您也会收到电子邮件的提醒。Safe Software的合作伙伴进行培训可能会要求您填写单独的表格，但如果您愿意，也可以使用官方的Safe Software表格。

证书

E. Dict先生（**FME**律师）说...
为了证明您参加了此培训课程，证书将自动通过电子邮件发送给在Safe Software举办课程期间登录的任何人。

谢谢

感谢您参加此FME培训课程。