

FME Desktop 培训

— Desktop 基本介绍

北京世纪安图

<http://www.antu.com.cn>



五个基本实例介绍



实例一：目标通用格式

我们运行含有通用编写器的模版时，便不需要依赖于目标格式，可以在导航窗口设置发布目标格式，来决定要输出的数据格式。

FME 基本介绍



目的：我们创建一个模版，转换INTEROPOLIS城市的数据，并且将它上传到FME Server中，这样这个城市的居民就可以在互联网上下载数据，并且我们希望用户能够自己选择数据的格式。

要求与数据路径：

启动Workbench ，将Autodesk SDF 转换成 Generic （任何格式）。

输入格式： Autodesk MapGuide Enterprise SDF

输入路径： C:\FMEData\Data\Transit\Transit.sdf

输出格式： Generic (Any Format)

FME 基本介绍



过程:

图1: 创建一个模版, 目标格式为Generic (任何格式)。

图2: 红色箭头位置显示目标格式参数。

图3: 执行设置目标格式。

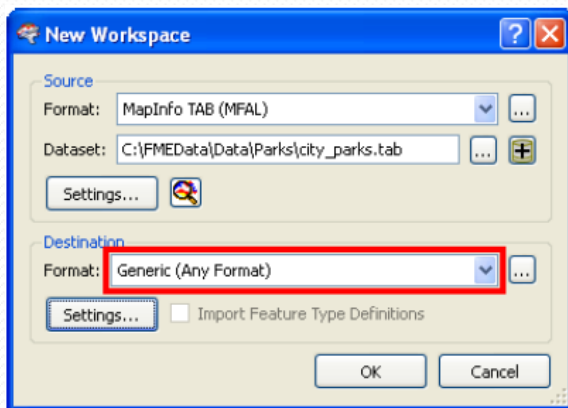


图1

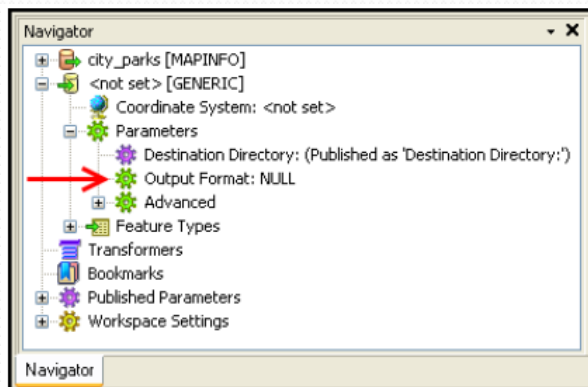


图2

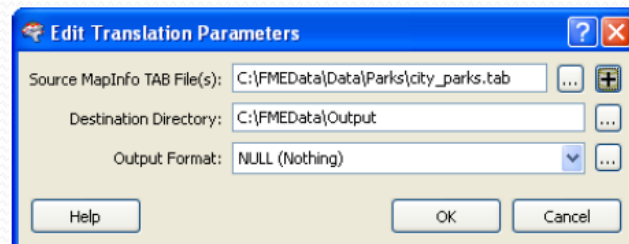


图3

FME 基 本 介 绍



实例一进行中。。。。



实例二：源图层的自定义读取

为了动态地读取一系列用户自定义的层（但是提前并不知道），我们就需要首先处理未定义的要素类，并且控制已有的要素类。

FME 基本介绍



目的：

我们希望扩展实例1中的模版，让用户能够选择要转换的源要素图层

要求：

启动实例一模版，对Feature Types to Read 进行发布，然后在进保存运行。

FME 基本介绍



过程:

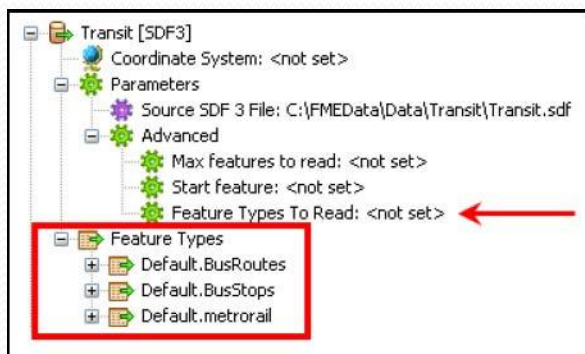


图1：读入三图层



图2：设置执行图层

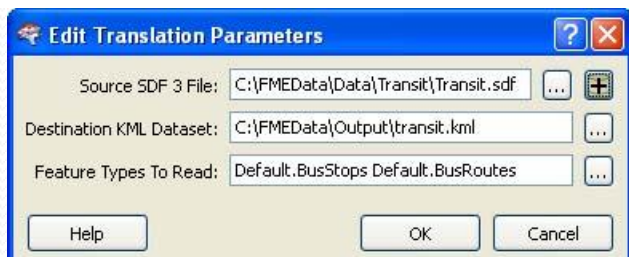


图3：执行

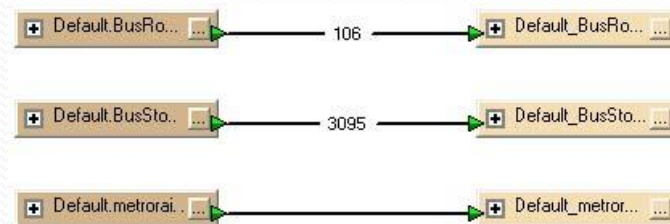


图4：结果

FME 基本介绍



实例二进行中。。。

FME 基本介绍



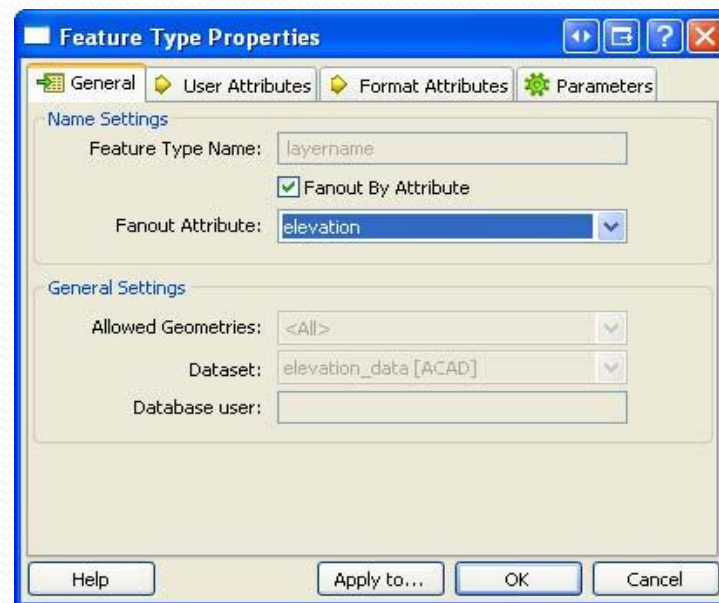
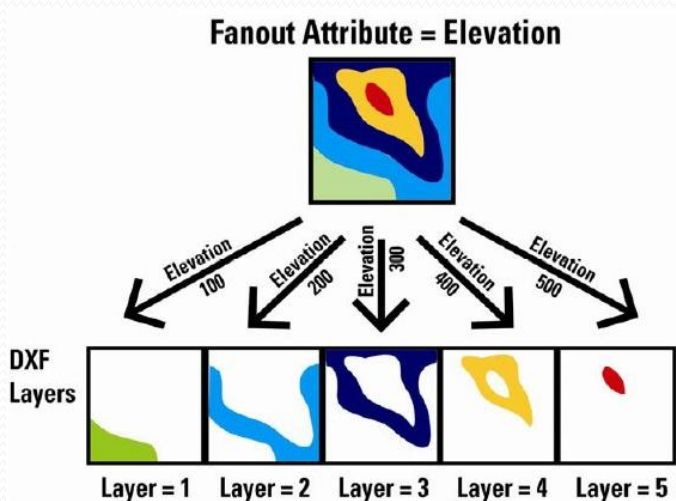
实例三：要素图层输出自动分层

不需要创建大量的输出数据层，使用它就能够在单一目标数据集中将数据分为多个层

FME 基本介绍



在立体图属性的基础上，
将数据划分成多个部分，
然后在一个DXF文件中，将
每个部分编写到不同的
层中



设置一个输出分层，就
会得到上图的分层效
果。

FME 基本介绍



目的：有一组AutoCAD文件来储存有关水管分布的信息。
你要做的就是，创建一个GML数据集，在一个单独的类别中储存每个管道的直径。

要求与数据路径：

输入格式：Autodesk AutoCAD DWG/DXF

输入路径：C:\FMEData\Data\Water\distribution L25.dwg

输出格式：GML (Geography Markup Language)

FME 基本介绍



在模版中，我们只需要一张表：water_distribution_lines

首先以DIAMETER属性进行分层

方法：打开water_distribution_lines 要素类的feature type properties对话框，在Fanout by attribute一栏打钩，然后选择DIAMETER作为扇出属性。

再以STATUS 属性进行分层

方法与上面一致

对比两次输出结果的区别

FME 基 本 介 绍



实例三进行中。。。。



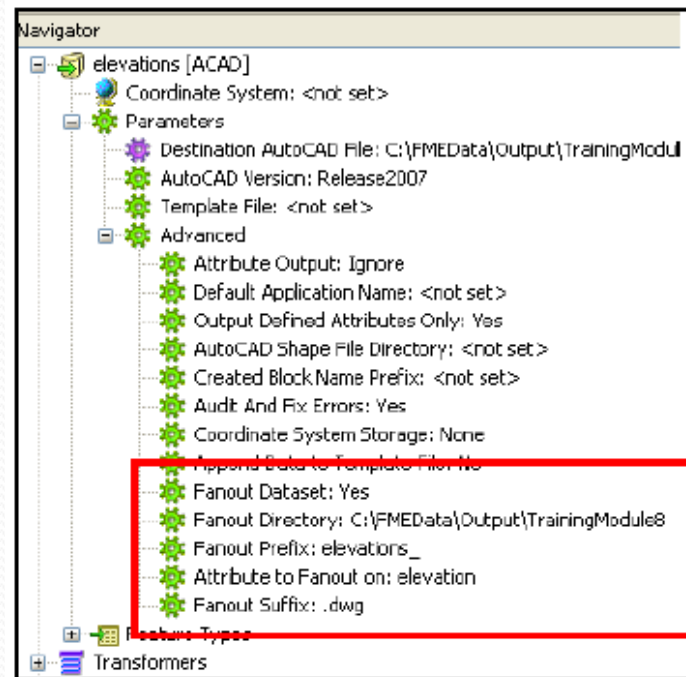
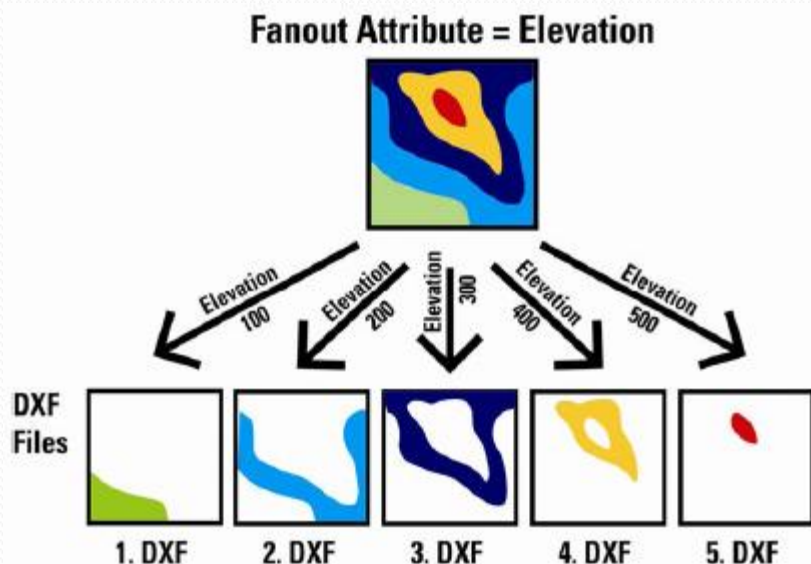
实例四：要素图层输出自动分数数据集

不需要创建大量的输出数据集，使用它就能能够在单一目标数据集将数据分为多个数据集

FME 基本介绍



在立体图属性的基础上，
将数据划分成多个部分，
然后输出多个DXF文件中，
将每个部分编写到相同的层中



右击输出数据集，选择
Fanout Dataset，将
Fanout Dataset 设置成
Yes

FME 基本介绍



目的：使用一个GML数据集来保存Interopolis城市的区域划分信息，你要做的就是为每个区域创建一组AutoCAD DWG文件。

要求与数据路径：

输入格式：GML (Geography Markup Language)

输入路径：C:\FMEData\Data\Zones\zoning.gml

输出格式：Autodesk AutoCAD DWG/DXF

FME 基本介绍



- 通用进行转换
 - 将TYPE设置为分类要素集输出
- 对比两次结果

注：我们要注意一个小问题：标有“W”的文件夹是由区域名种的“/”符号产生的。事实上，区域名是w/L0, 它导致出现越来越多的W文件夹，包含一个叫做L0的数据集。

FME 基 本 介 绍



实例四进行中。。。。



实例五：动态模式

前面已经提到了，通用编写器的最大弊端就是，它需要设置一个预先定义的模式。现在我们就用动态模式来解决这个问题。

FME 基本介绍



数据路径:

打开实例二当中的模版

我们需要将马路添加到Bus stop要素类，方便查询Bus stop数据。使用Source Data > Add Dataset，就能够添加一个新的阅读器和数据集。

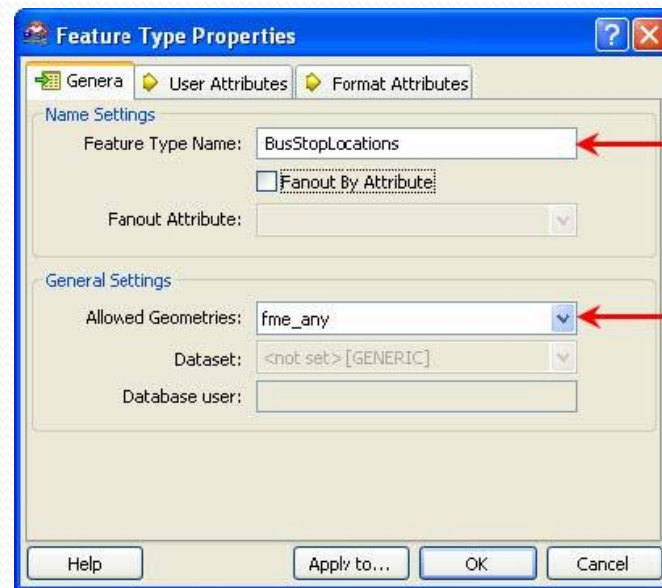
输入格式: MapInfo MIF/MID

输入数据: C:\FMEData\Data\Roads\RoadLine.mif

FME 基本介绍



将源RoadLine 连接到输出
Feature Type
Default_BusStops ，将目标要素类重新命名为
BusStopLocations 将 Allowed
Geometries设置成fme_any。



注意：输出结果Bus Stop要素含有属性，而Road则没有！

这就是Generic Writer:最大的弊端：不能够编写未定义的属性。

FME 基本介绍



现在让我们使用“动态模式”来解决这个问题

数据路径:

输入格式:

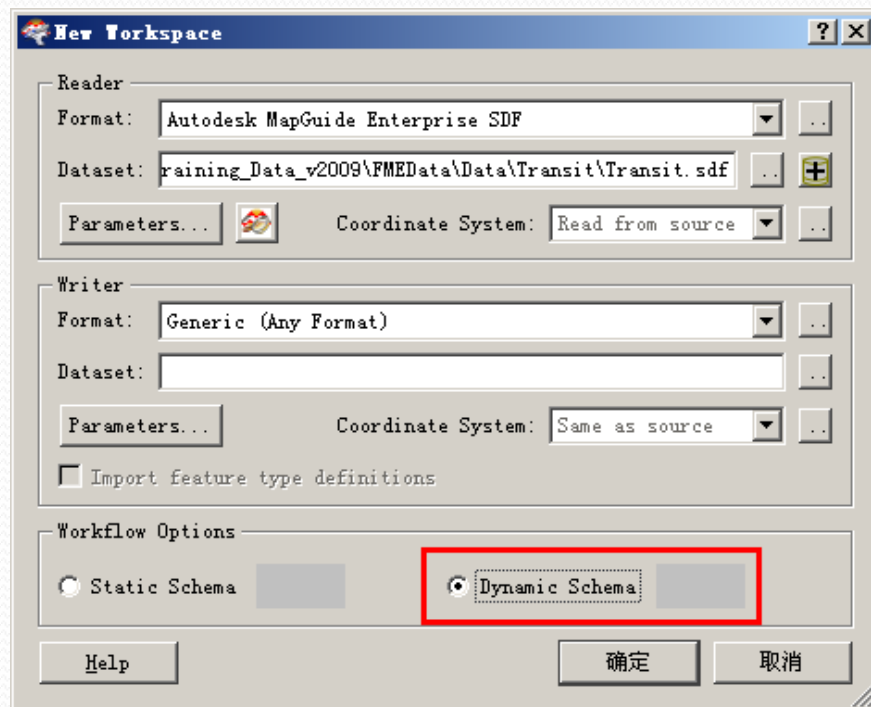
Autodesk MapGuide Enterprise SDF

输入路径:

C:\FMEData\Data\Transit\Transit.sdf

输出格式: Generic (Any Format)

注意: 这次只有一个单一的输出要素类, 而不是三个



FME 基本介绍



我们再次将马路添加到Bus stop要素类，使用Source Data > Add Dataset，就能够添加一个新的阅读器和数据集。

输入格式： MapInfo MIF/MID

输入数据： C:\FMEData\Data\Roads\RoadLine.mif

使用发布的参数Feature Types to Read，限定SDF只读取BusStop Feature Type。

在输出中有两种要素类：一个是马路，而另一个是公交车站，且两个要素类都有它们自己的属性设置。



谢谢