

FME 辅助室内地图数据拆分及整理

认识 FME 是 2012 年了，初识 FME 就被 FME 灵活快捷、搭积木式的风格所吸引。时至今日，但凡能想到用程序解决的问题，总会第一个想到 FME。FME 对于我这种有点程序思维，但又不会什么编程语言的人来说，简直是一个不可或缺的好帮手。

FME 具有强大的数据处理功能，同时又支持多样的数据格式，工作中 FME 帮我解决的问题涉及图形数据转换、入库、质检、分幅裁切、精度统计、影像处理、坐标转换、地图投影等方方面面，若是遇到解决不了的问题，也总是可以在博客、交流群里得到友爱细致的帮助。总之，FME 给我的感觉真是只有你想不到，没有它做不到。

这不，前不久又遇到一个令人头疼的整理室内地图数据的问题，繁琐大量的手工作业又让我想到了这个助手。之所以选择这个问题来分享，是因为这个方案里用到了一个新学的小技能—循环功能，有点小小的成就感。

案例情况是这样的：工作要求将 5000 幅室内地图（CAD 格式）转为要求的 SHP 格式，CAD 转 SHP 对 FME 来说岂不是小菜一碟。然而拿到手的数据是这样，室内地图数据分层排列如图 1 所示，最终提交的 SHP 要求将所有楼层数据像叠叠高一样在任意位置叠起来。

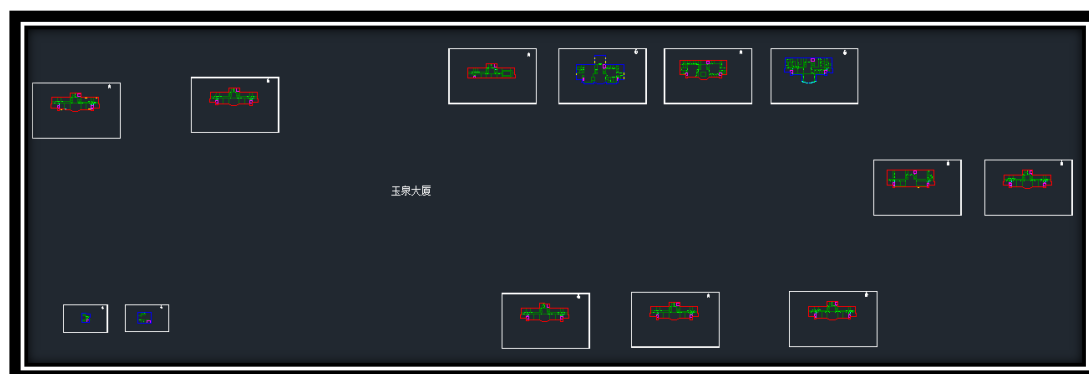


图 1 某某大厦室内地图数据（地上 12 层+地下 1 层）

图形数据的分层及属性见下表。

图层类型	图层名称	字段定义
注记点	建筑物	建筑名称；建筑地上楼层数目；建筑的地下楼层数目；
线	墙体	厚度；材质；楼层；墙体高度
面	天花板	厚度；材质；楼层；高度

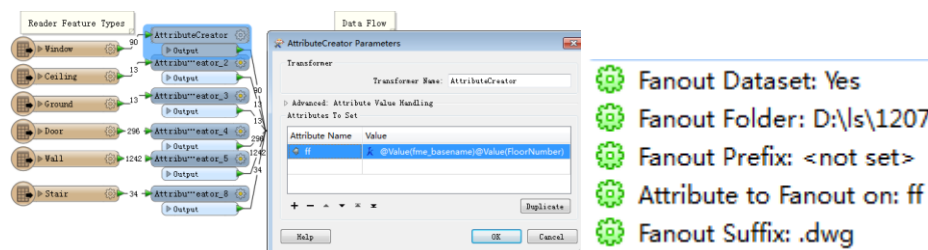
面	地板	厚度; 材质; 楼层; 高度
线	门	厚度; 材质; 楼层; 高度
面	窗	厚度; 材质; 楼层; 高度
面	楼梯	厚度; 材质; 楼层; 单阶高度
面	立柱	厚度; 材质; 楼层; 墙体高度

通过分析,可以将注记点作为数据移动的目标点,任务即为将注记点周围的 13 张图形按图形中心自动移动到注记定位点的位置。考虑到如果在原图形上进行移动叠加,数据压盖严重,后期不方便修改。因此必须从一张图中分出 13 张不同楼层的数据,并且完成移动叠加。

如果没有遇见 FME,我大概能想到的办法就是从这一幅图按楼层另存出 13 张图来,然后依次按图形中心手工移动图形到同一个位置。天呀,如果一栋楼有 31 层呢,简直不敢想象是怎样的一个工作量。然而事实当然有更简便快捷的办法,下面按当时的思路来介绍一下我的 FME 方案。

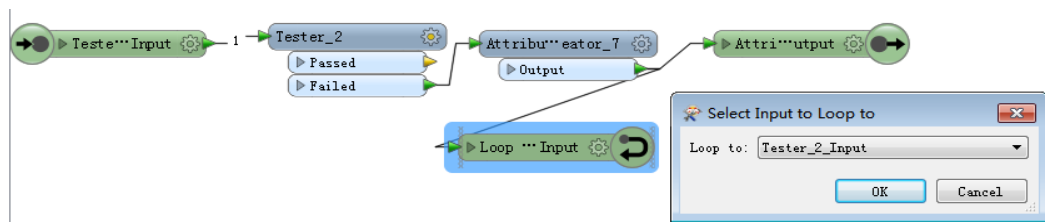
1、按文件名(即某某大厦)结合图内的楼层信息输出 13 张图;具体实现:

逐层用 AttributeCreator 创建新属性 ff@Value(fme_basename)@value(楼层),然后按 ff 属性扇出。事情到此,并没有结束。



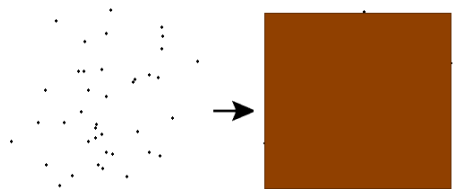
为了检查每一张分层图是否移动到了同一个注记点的位置,故计划把注记点分别输出到 13 张图中,但建筑物点图层中是没有楼层信息的,因此不能用上面的办法直接扇出。一个点利用地下楼层信息和地上楼层信息怎么变成 13 个点呢?当然用拷贝注记并逐层赋属性的方法也可以实现,但是模板就会繁琐很多。经过分析,决定用循环判断来试试看。首先对于注记点的地下楼层来说,先默认“楼层=负的地下楼层”,然后循环判断“(楼层+1)是否等于 0”,如果不等于 0,那么“楼层=楼层+1”,如此循环。同理,对于地上楼层来说,先默认“楼层=地上楼层”,然后循环判断“(楼层-1)是否等于 0”,如果不等于 0,那么“楼层=楼层-1”,如此循环。这样 1 个注记点就变成了 13 个点,并分别添加上了“楼层”属性,此时再按照上面的扇出规则就可以依次输出命名为“某某大厦-1.dwg、某某大厦 1.dwg..... 某某大厦 12.dwg.”的图形数据。注意:循环只能在自定义转

换器中创建，循环模块中可以选择输入端，这样从输出结果出来再进入上一个输入端口就构成了一个小小的循环。当然，循环需要一个条件来终止循环，这里添加了 Tester 转换器进行条件测试，如果满足条件则终止循环，不满足条件则继续循环。

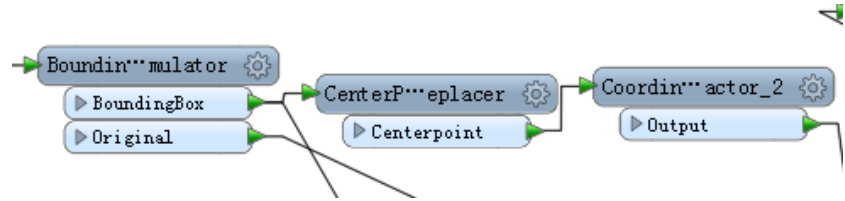


2、平移数据，使 13 张图幅中心点的位置与注记定位点的位置均保持一致。如何让整个图形通过一个点位来移动呢？用 CAD 软件操作时，我们会手工全选整个图形，然后大概选中图形中心的位置用鼠标拖动、平移到最终需要的位置。可是这个动作如何用模板实现呢？思考着操作数据的过程，终于想到了实现的办法。

①获取图形中心点坐标。首先，获取图形中心一定要把整个图形当作一个整体来看，通过寻找帮助，找到了 BoundingBoxAccumulator 这个模块，这个模块可以使图形聚合成一个外接边框，如下所示。



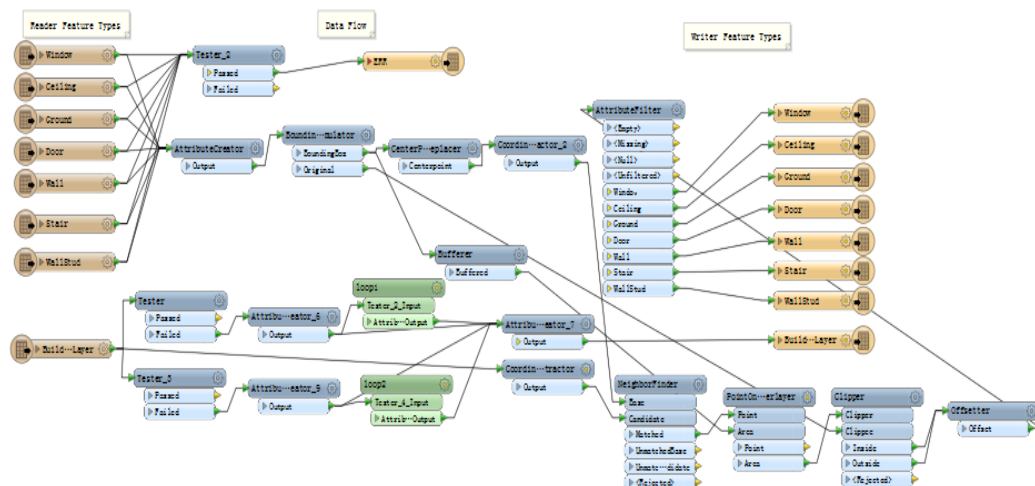
因此可以轻松利用这个模块按照楼层属性获取每个图形的边界，再利用 CenterPointReplacer 模块将边框替换为中心点，最后用 CoordinateExtractor 模块提取该点的坐标 (x1,y1)。



②关联注记定位点的坐标。这里用到 NeighborFinder 搜索一定阈值内的相关要素，阈值参考实际距离设置。搜索前同样用 CoordinateExtractor 模块获取注记定位点的坐标 (x2,y2)，这样按阈值匹配成功后的要素就同时赋上 x1,y1 和 x2,y2 两个坐标的属性。

③数据平移。通过两个坐标值计算得到图形中心点和注记定位点之间的 x 、 y 偏移量 $-(@Value(_x1)-@Value(_x2))$ ， $-(@Value(_y1)-@Value(_y2))$ ，然后使用 `offsetter` 模块进行图形偏移。实践证明，注记点和图形的位置关系并不影响图形的偏移结果。若图形在注记右上侧，即 $x_1 > x_2, y_1 > y_2$ ，那么图形向左、向下移动，即偏移量为 $-\Delta x, -\Delta y$ ($\Delta x, \Delta y$ 均大于 0)；若图形在注记左下侧，即 $x_1 < x_2, y_1 < y_2$ ，那么图形向右、向上移，即偏移量为 $-\Delta x, -\Delta y$ ($\Delta x, \Delta y$ 均小于 0)，负负得正，偏移量同样正确。

介绍到这里，关键问题都得以解决了，整个模板搭建起来很简单，如下图所示。这一次利用 FME 又成功的解决了我的数据整理问题。



后面 CAD 转 SHP 就不再赘述了，利用 FME 也是轻松搞定。

这个案例虽然可能借鉴的意义不大，但我想分享的是，FME 就是这样的可以让我用程序思维去解决一些看似繁琐复杂的问题，它省去了写代码的工作，取而代之的是一个封装的功能小模块，你只需要通过串联或者并联不同的模块就可以实现你的不同想法，这个过程奇妙而又有趣，每每实现想法，都会让不会编程的我感到小小的兴奋，获得编程带来的快乐。

没有做不到，只有想不到，很多繁琐的手工作业背后可能都有程序实现的可能性，让我们解放思想、动手搭建自己的数据处理平台吧！