地图生成器是在勍哥原来的版本的地图生成器的基础上改出来的，有些特性是保留自原来的版本：

1. 生成的地图的预览；原来是 XML，现在会生成多幅描述各种不同对象的分布的图片
2. 多遍式的程序结构；比如生成基础地块是一遍、生成区域是一遍、生成大矿是一遍等等，整个生成过程中，地图会被“刷”上很多遍

工程的模块结构目前是这样：

1. tools 模块里面包含主要生成逻辑
2. cim 是核心接口模块，被频繁使用的对象结构被定义成接口放在这个模块中
3. ccm 是核心类模块，支持生成逻辑的一些相对独立的类定义放在这个模块中

目前地图生成的大概遍数（大步骤）：

1. 生成基础地块
2. 地表着色
3. 区域划分
4. 出生点生成
5. 地形生成
6. 据点生成

地图生成遍数的详情：

第 1 遍由 MapInfo 的构造方法完成，当 MapInfo 对象被创建后，所有的地块对象都被创建出来了。

第 2 遍由 ColorLumpCreator.instance.create 方法完成，它会从工程里的 res/terrains 目录中读取地块颜色的描述图片，这一组图片都是由美术的同事提供，文件的命名规则是：

1. 名称里所有元素由下划线连接
2. 小写字母 t 开头，t 是 terrain 的缩略，原本还有 c 开头的（表示 cover），由于覆盖物的概念已经变化不再需要 c 了
3. 第二个元素一定是颜色描述，core 意味着核心区的黑色地表，就是沙黄色，其余名称与美术约定好即可，通常就用颜色的英文名就好
4. 剩余的描述全部是 x、y 坐标值，因为美术在 Unity3D 中做的地图是 10x10 的大格子，因此 x、y 的取值范围都是 0-9，表示当前文件描述了哪几个大格子的地表颜色

目前地块颜色的描述文件只支持单色（其实是白色表示的背景色沙黄色，外加黑色表示的另外一种颜色），如果未来需要支持更复杂的地块颜色描述，需要修改 ColorLumpCreator 的逻辑。

第 3、4、5、6 遍全部在 TerrainCreator 里完成，入口是 create 方法，AreaDescription 对象是区域描述，用于将地图分出区域，区域边界的宽度目前是 3。generate\_player\_start\_on 方法用于生成玩家出生点。fill\_terrain 方法用于生成地形。据点信息有两个来源，一个是数据表 stronghold\_meta\_prototype，这个表是依据潘凡提供的据点的 excel 表格手工填的，另一个来源是 AreaDescription 的 EdgeDescription，这一来源的据点全部是边界上的关隘。

区域划分算法主要思路是，区域描述成一个多边形，然后逐点判断一个地块（点）是不是在多边形之内。由于区域的边界在咱们的实际需求中是有宽度的，所以这个判断算法中有很多逻辑是用来处理边界的。

生成出生点的算法很直接，没有特别可以强调的，基本的思路是准备一片地块（就是区域），在其中随机生成，生成一个出生点，就从这一片地块里减掉所有涉及的地块，然后继续，直到地块被消耗完。

地形生成涉及两个类，FixedDistribution 以及 ProbabilityDistribution，前者是固定数量的某种地形的描述，后者是概率的分布。这是根据策划前期给的需求直接设计出的两个类，如果策划的这两种规则不变，这两个类应当都可以继续使用。

没算在遍里的是数据导出到数据库的过程，这一步没有什么可说的，就是将 MapInfo 里所有的地块信息输出到 map\_cell\_prototype 里。

我给一些关键的行为加了单元测试，如果将来需要在我写的这些代码的基础上修改，单元测试可能可以提供一点帮助，至少能测试出新的行为和以前的行为是不是有差异。