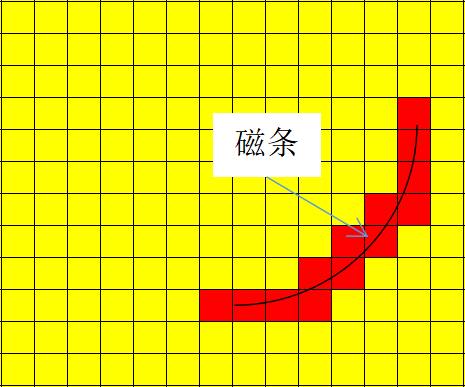
# 沙盘地图的规划

制作沙盘的时候需要实现规划沙盘模型模拟的原始地图的尺寸如下表，然后按照正确的缩放比例，缩比制作。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 项目 | 长(m) | 宽(m) | 网格块 | 备注 |
| XX21X | 10.7 | 1.4 |  | 不重要 |
| 库位 | 12 | 3 |  | 准确 |
| 库 | 120 | 9 |  |  |
| 全库 | 2400 | 1200 |  | 3中心库，3通道，5出口 |
| 通道宽度 | 不定 | 4 |  | 通道拐弯圆弧半径20米 |
| 车的宽度 | <10 | <2 |  | 车辆按照宽度为1.8米约定 |

在地图软件Tiled上，一个块建议分配8x8像素，然后对2400x1200米，真实尺寸地图进行网格化。如果按照推荐的大小800x400的网格划分，每个块（8x8像素）代表3x3米。每个网格3米代表单通道的宽度，其中AGV车辆宽度设为2米。真实场景的车辆加载物10米，占4格，规划算法将AGV简化为1格，可能导致规划路径实际运行发生偏差，但考虑到库区场景很大局部细节偏差直接依靠车端的控制软件进行动态调整。利用仿真软件对这种情况可进行仿真。

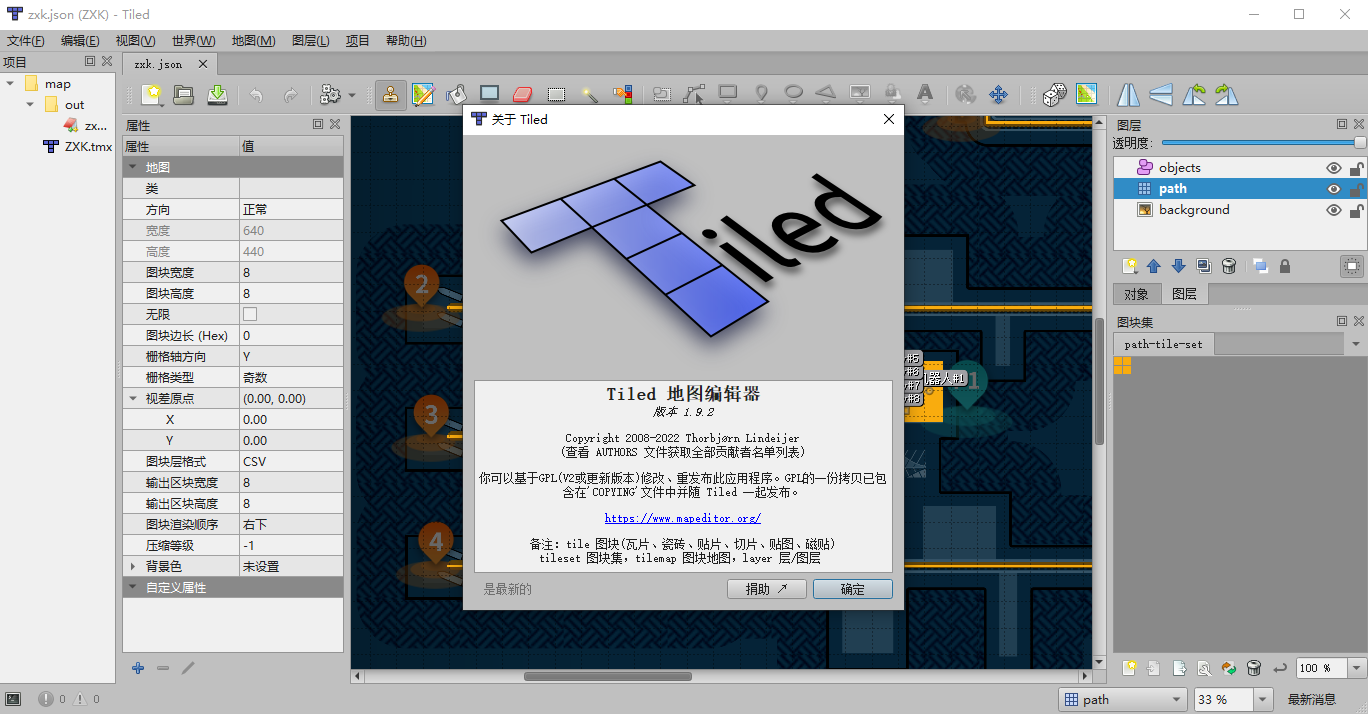


车辆行进过程中，实际是依托磁条进行导航的，如上图。车辆应该以磁条进行导航，并尽量匹配路径规划服务器下发的路径规划，即在要求的时间步长，抵达要求的方块区域，并实时上报块编号。

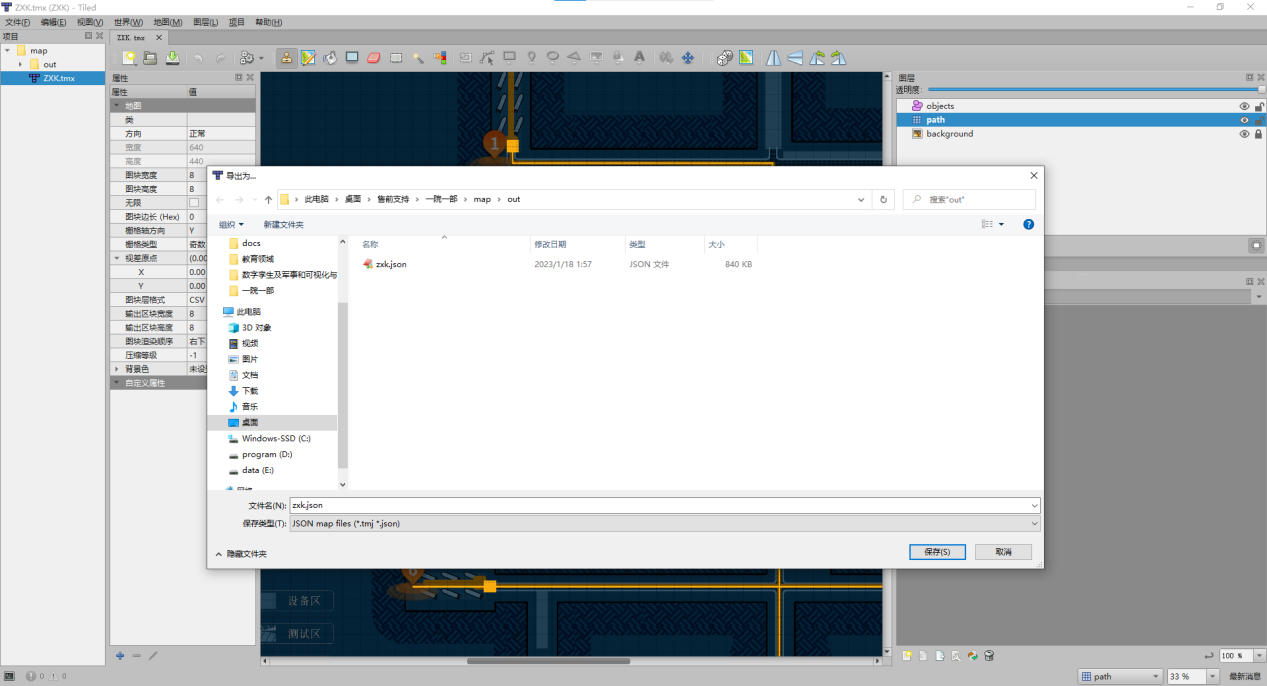
在沙盘模型中，如果小车的实际尺寸是30cm，则可以将沙盘与实际物理库区的1/10制作，30cm代表地图中的一个方格。

# 地图制作的说明

## 地图的生成使用Tile软件



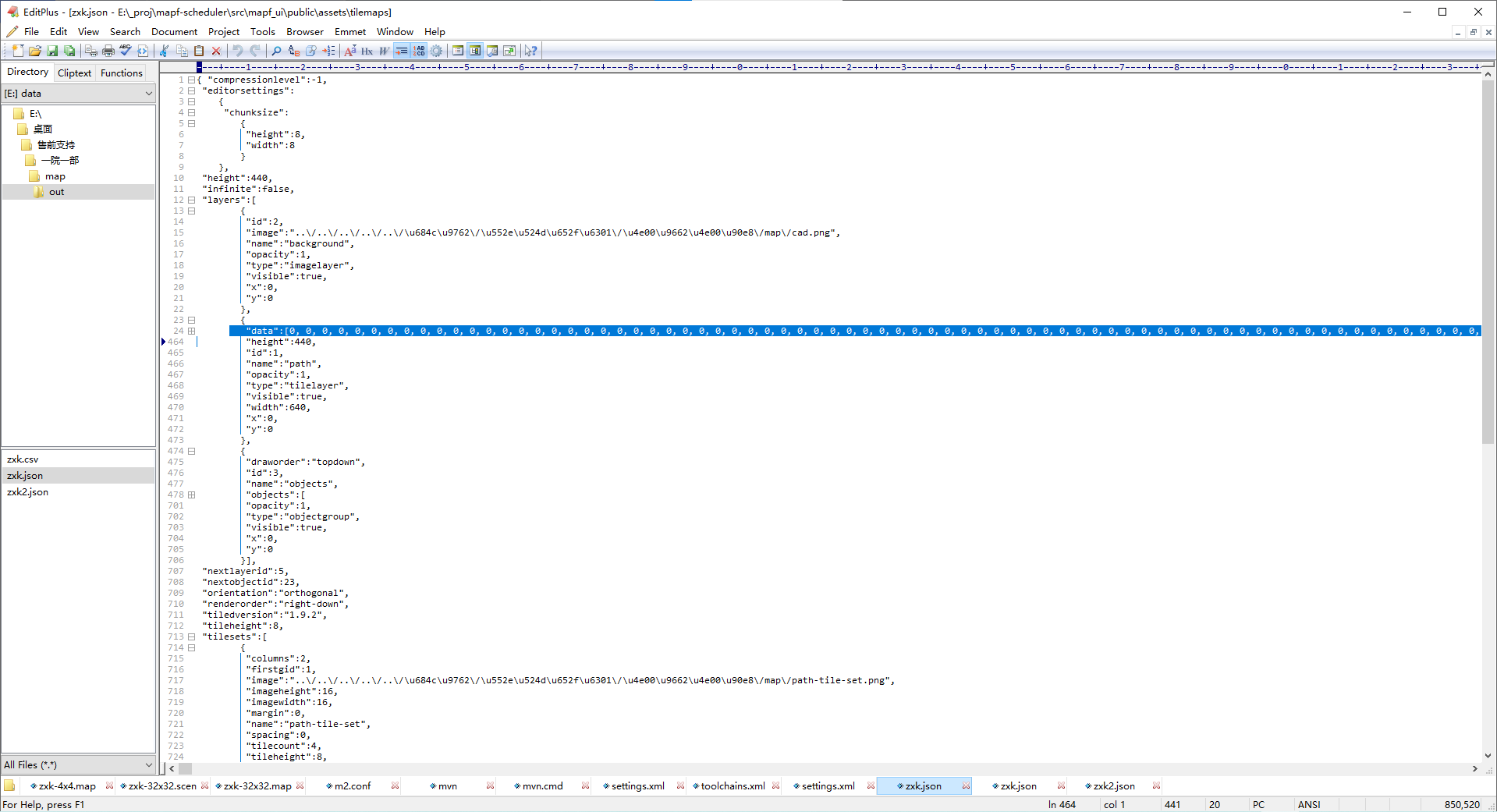
## 地图导出json文件



这个导出的地图json文件主要用于phaser3的界面绘制，它包括主要两个外部引用的内容：

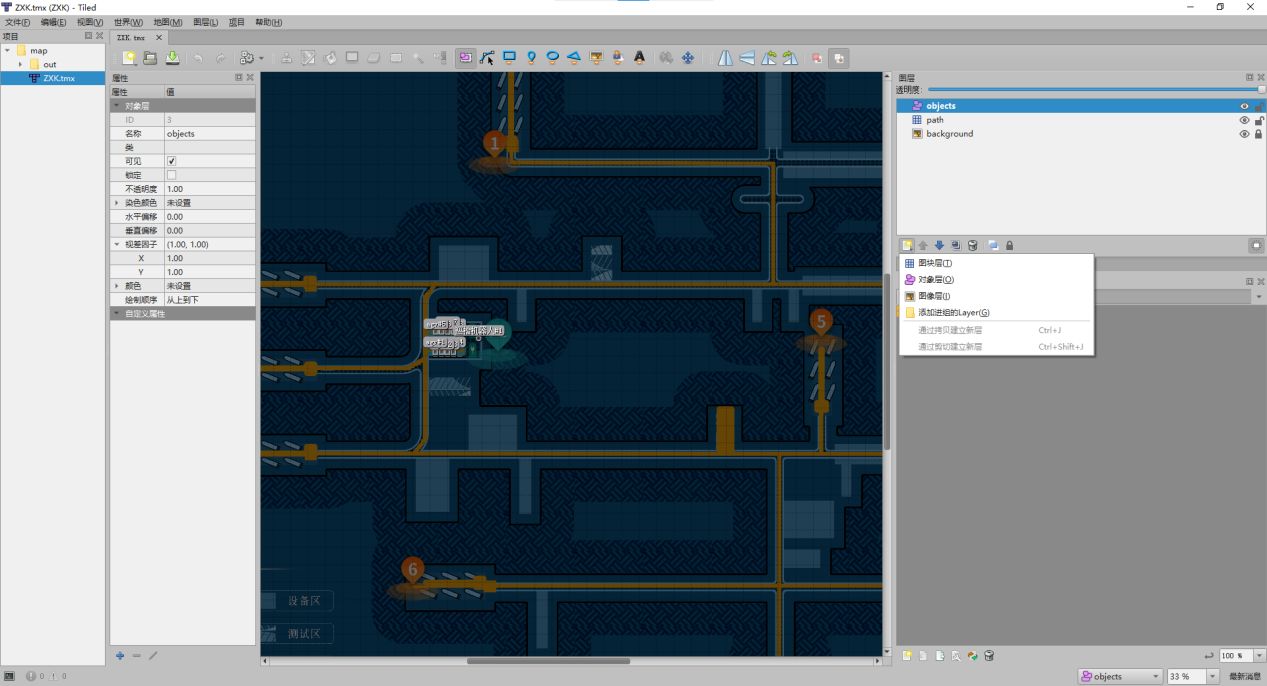
（1）tilesets

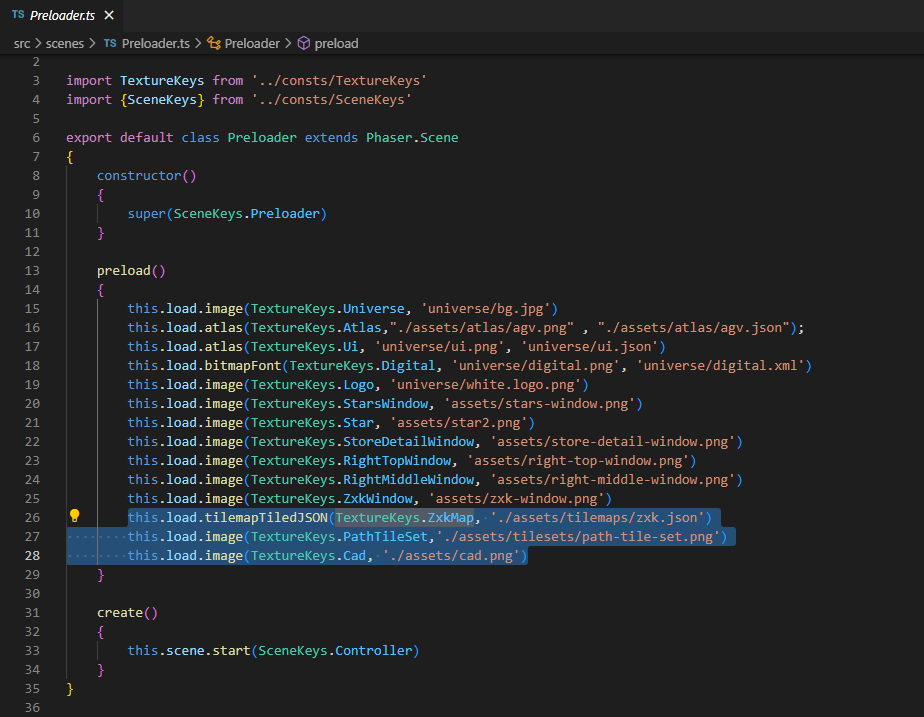
tilesets是图元信息，它可以是若干基本图块，在Tile中用于在地图底图上绘制底图的路径、障碍物等。绘制的结果就会被记录到对应的图块层的data字段中。



（2）layers

layers图层主要包含图块层（path）、对象层（objects）和图像层（background）。图层是完全按照用户自己编辑来创建的。图层创建后导出json后，被phaser3再使用。





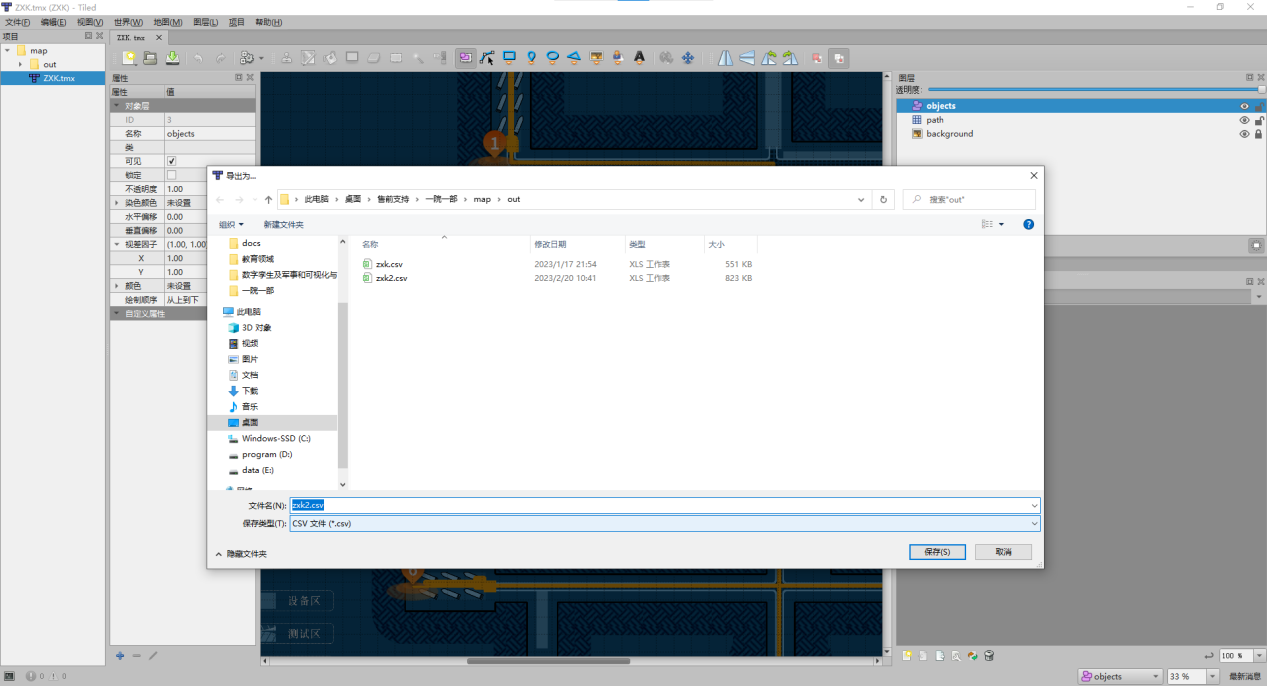
【注意】

zxk.json地图文件内部有对cad.png，path-tile-set.png等外部文件的引用，只是用Tile编辑地图的时候需要，并不会被phaser3使用。phaser3需要额外导入这些引用的外部资源如上图代码。

## 路径规划使用的地图文件

路径规划使用的地图文件格式不同于phaser3，它需要从1.2中图块层的data中提取相关的信息转化为路径规划程序需要的格式。

当前还没有这样做，而是在Tile中导出json的同时，又导出一个csv文件，然后手工编辑而成。



# 地图发布的说明

tile工具导出的csv文件（utf8编码）为输入，经过如下步骤完成发布工作。

目前产品经理会发一个xlsx地图文件，需要先将其另存为csv文件。

## 规划算法地图文件生成

（1）用editplus这类编辑工具打开csv文件，通过查找替换工具，把-1转换为@，0转换为.，逗号替换为空。-1>@，0>.，,>

（2）添加map文件的描述信息，csv文件另保存为zxk-73x39.map的文件名。

type octile

height 39

width 73

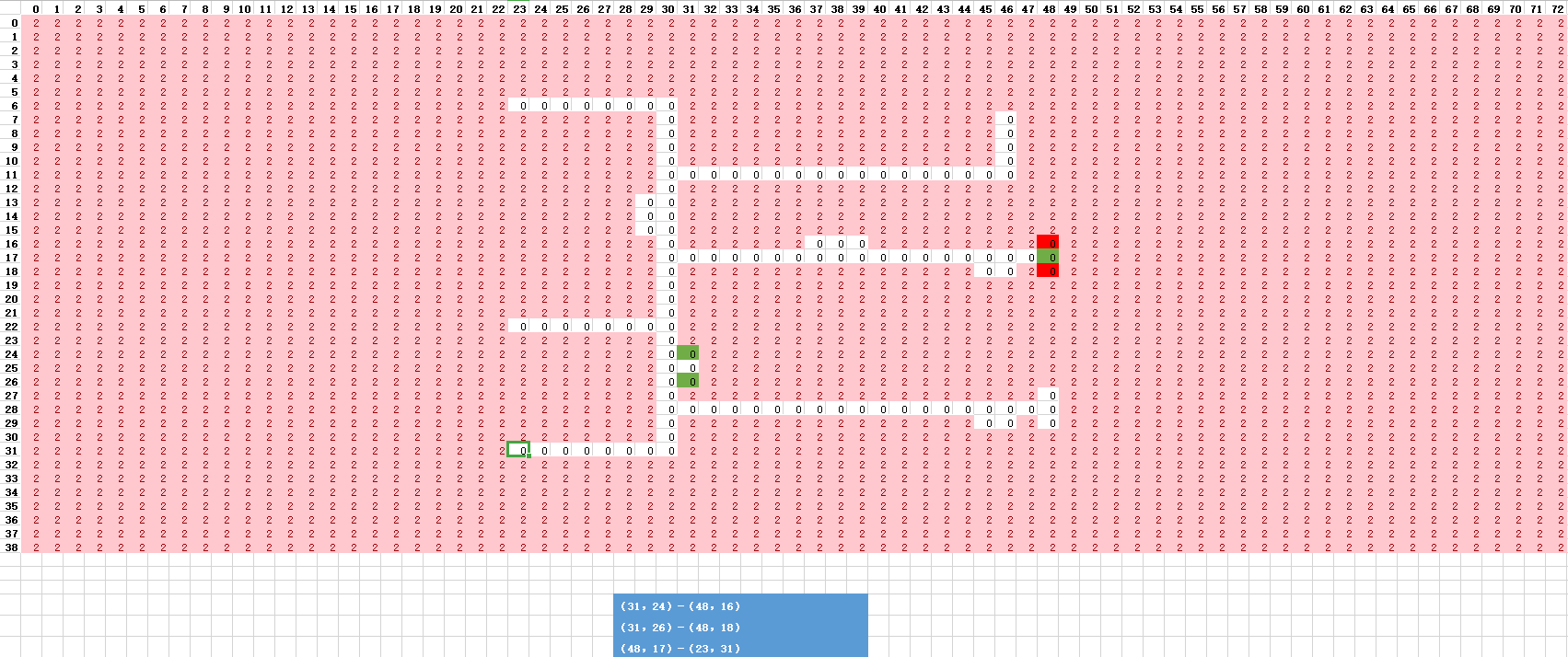
map

## 前端展示地图文件生成

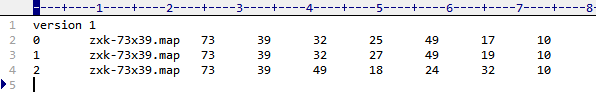
为支持前端可视化工具，csv文件还要转换一个前端版本，把原来输入的csv文件用editplus查找替换，把-1换为2，另存为zxk-73x29.csv。

## 人工生成起止点文件

为方便人工查看，可以用wps的电子表格打开zxk-73x29.csv，另存为xlsx格式，添加行列座标，并用条件格式，把单元格为2的单元格突出显示为红色，另存为zxk-73x29.xslx。注意行列座标为0开始。



在上述地图中，标记三个起点和终点任务。针对三个任务，定义一个zxk-73x39.scen文本文件（utf8编码）如下。

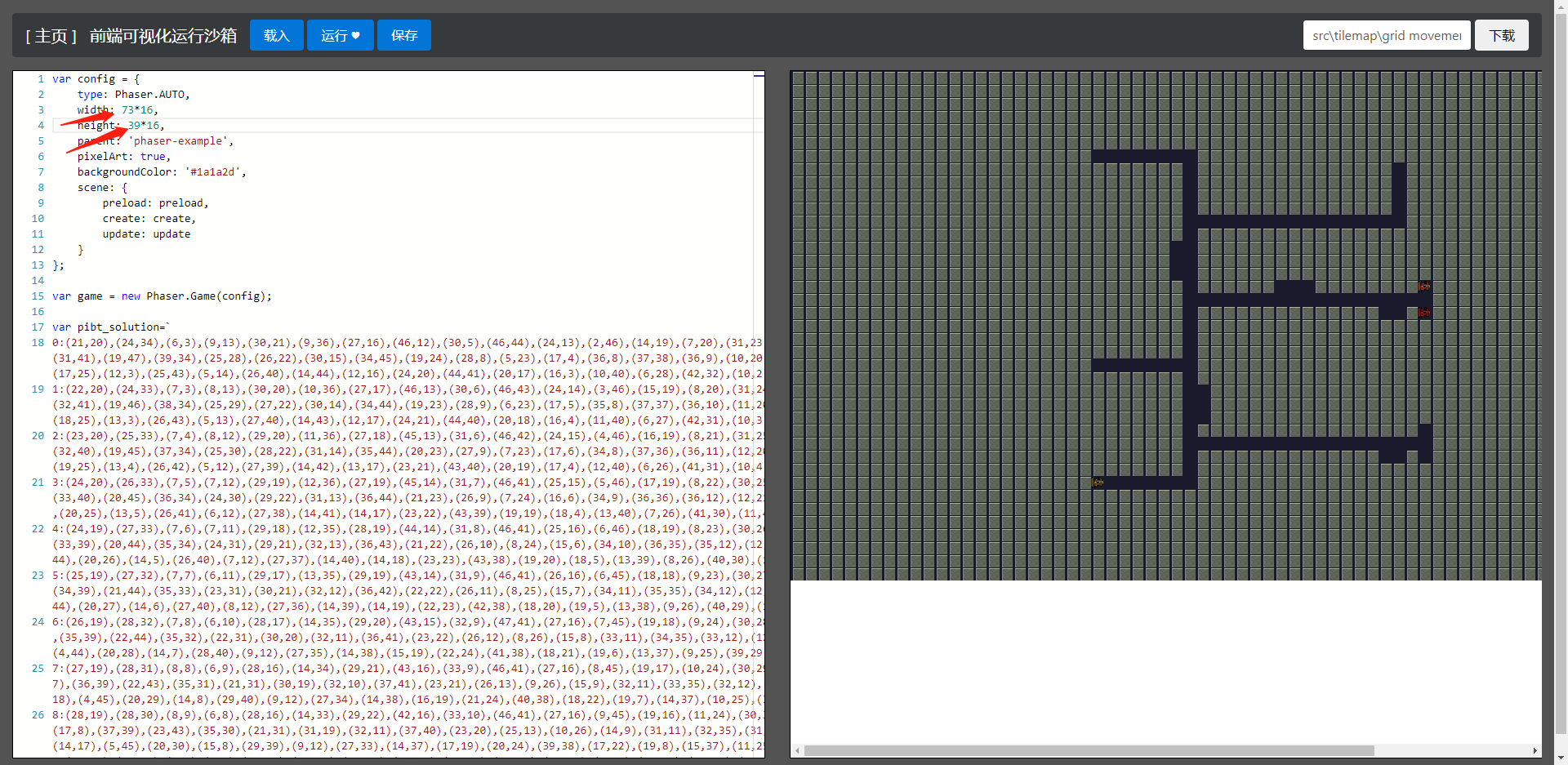


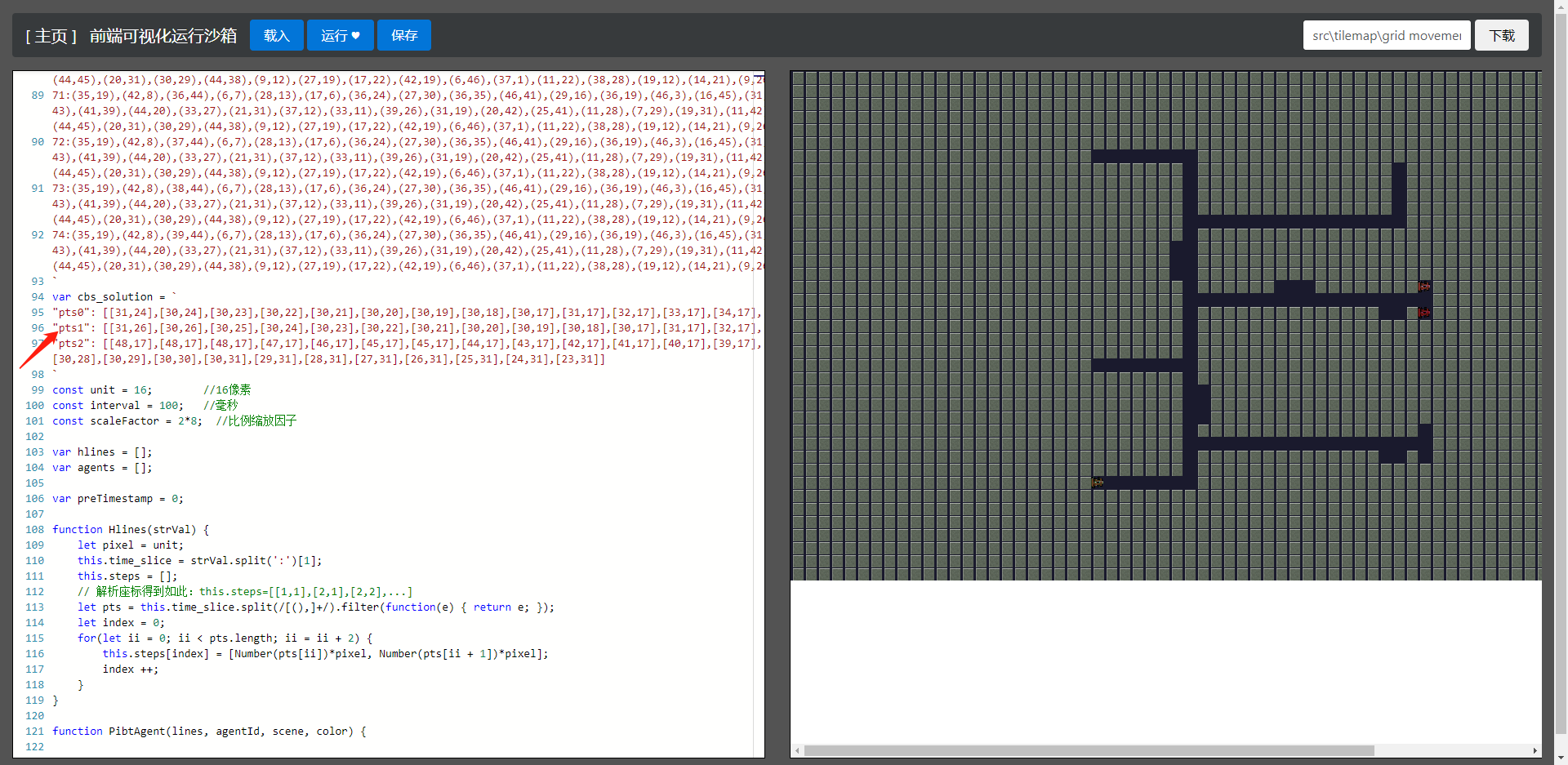
## 测试zxk-73x39.map和zxk-73x29.csv

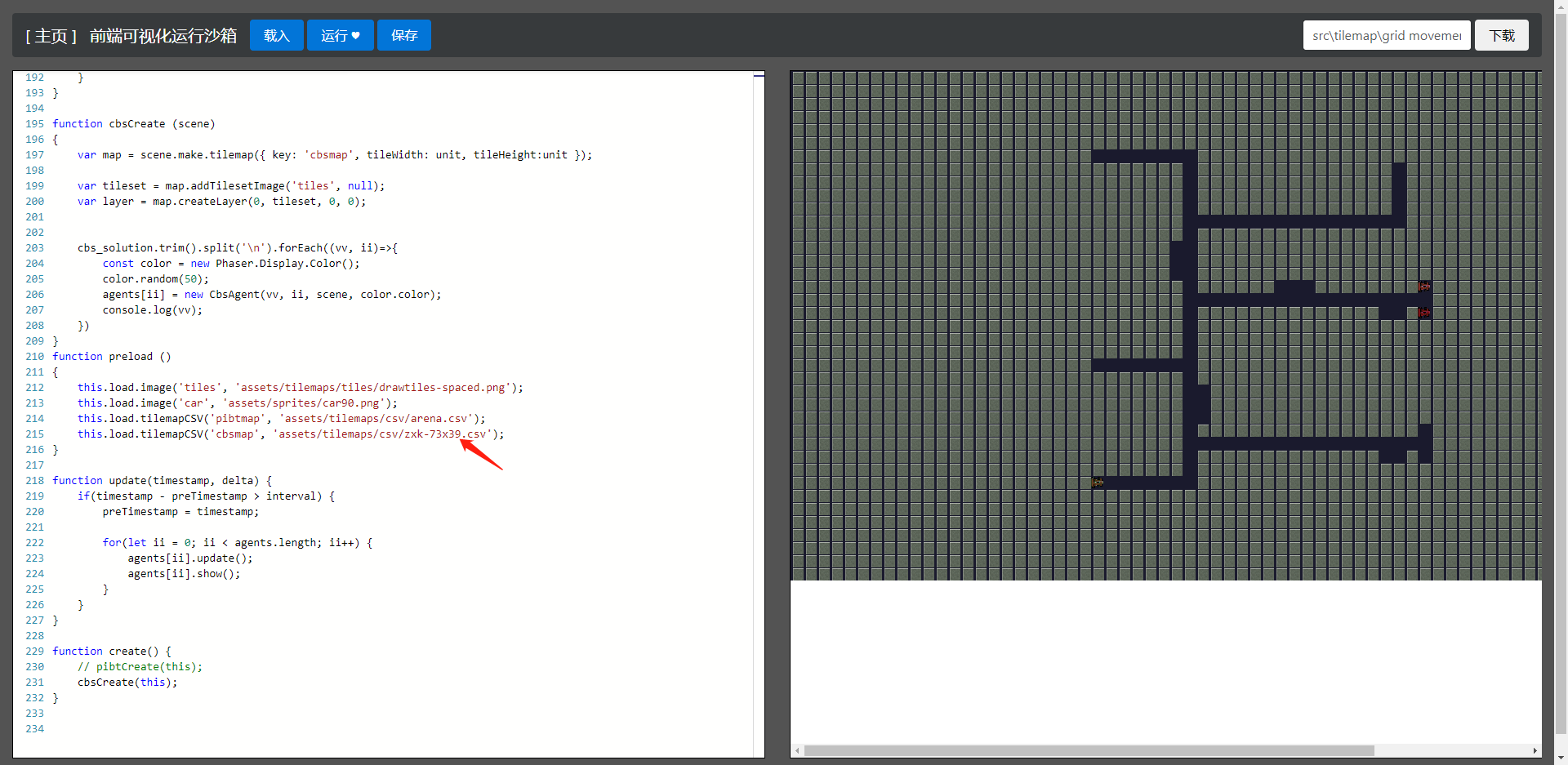
（1）在visual studio工具中，调试参数如下。

-m zxk-73x39.map -a zxk-73x39.scen -p dynamic.paths -o test.csv --outputPaths=E:\document\source\repos\CBSH2-RTC-main\paths73x39.json -k 3

（2）在visaul studio code工程中，npm start，在可视化界面上，修改三个地方，运行上面一行运行规划，生成的路径文件paths73x39.json的路径数据。







## 把zxk-73x39.map部署到测试服务器

（1）用ssh登录阿里云服务器。

（2）上传地图文件。

上传路径如下。

/home/ly/docker-build/mptools-mapf-scheduler/src/mapf/maps

（3）在ssh终端控制台运行如下命令。

cd /home/ly/docker-build/mptools-mapf-scheduler/src/mapf/maps

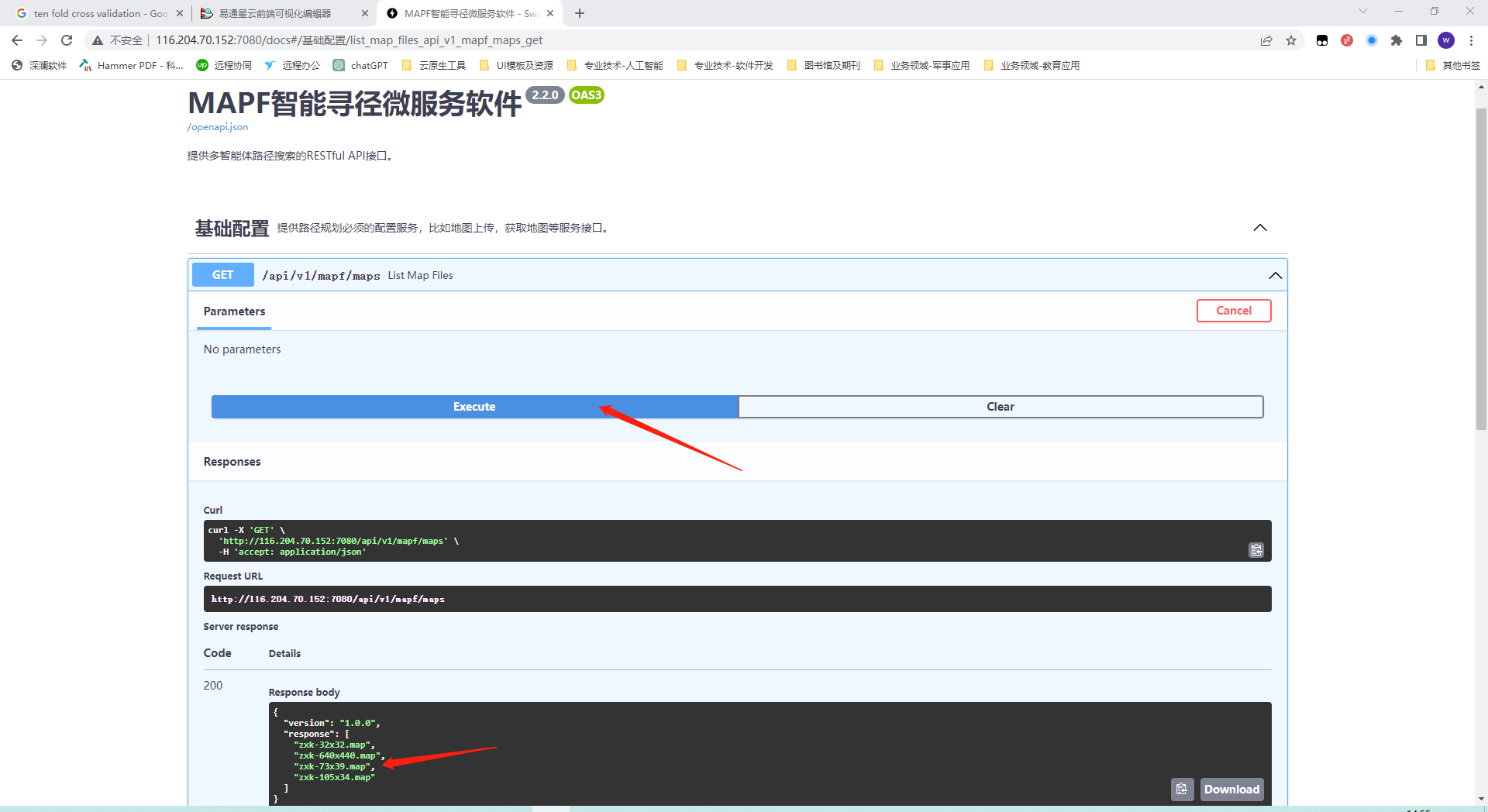
sudo docker container ls

sudo docker cp zxk-73x39.map 3cfb2826b844:/home/rootusr/src/mapf/maps

注意，在第2行命令用于查看cbs算法服务容器是否启动。

（4）如果拷贝成功，访问如下地址。

http://116.204.70.152:7080/docs



执行Execute，返回如上图，表明地图更新成功。