

[What's this all about?](#)

[Exporting the map to images](#)

[What does the macro do?](#)

[How do I use the tool for generating the macro?](#)

[How do I run the macro?](#)

[Stitching the exports](#)

[Warning - Memory](#)

[Google Maps](#)

[Generating the pre-zoomed files for google maps](#)

[Uploading](#)

## What's this all about?

I wasn't pleased with the exports of my maps, the reason for this was, I made the map too big. If the map is too big, then the exported file won't have the details, I would have to add additional files. An example: if you are creating a map of a village, you will see every room, every tree. But you might have added details to the tables, added tools to the desks and so on, this work will not be visible. This bugged me, so I looked for solutions and tried to use the google maps engine with a huge export of my map.

What is the workflow

You need to do the following steps:

- export your map as image
- split the image into usable pre-zoomed images
- create a matching html-file
- watch it

This document will help you doing this.

## Exporting the map to images

You can use the normal export and save it as jpg or png as you like, if you need more details, please read on, otherwise just skip this section.

The archive provides a tool for generating a campaign cartographer macro to export your map into many high resolution images. The main problem is, campaign cartographer has problems when exporting an image with above 9.500 x 9.500 pixels, sometimes it works, sometimes it doesn't.

## What does the macro do?

The macro calls the export rectangular selection save method with the coordinates by the java program. After you ran the macro, you will have several files you have to stitch to one huge image.

## How do I use the tool for generating the macro?

You need: lower left and upper right corner coordinates (T, not @T or ?T).  
Just edit "generate\_macro.cmd" in the scriptfolder to you desires.

## How do I run the macro?

You have to add it to your fcw32.mac (see helpfile of cc3 for details) and then call it from within campaign cartographer. Before you can do this, reload the macros by tools -> macros -> load or restart campaign cartographer after editing the macro file.  
After this, make some coffee, because the export will take a long time.

*Note: for me, the export crashed after about the 10th image. If you have the same problem, just remove the already generated image-lines from the macro and then restart campaign cartographer, just restarting the macro didn't work for me. I guess there is something wrong inside campaign cartographer when it has to do too much of exporting/rendering/saving.*

## Stitching the exports

After you did the export, you are left with multiple smaller files, but this isn't our goal. We need to have one huge image. This process is called stitching.  
In the download you find "stitch\_images.cmd" which will do this task.

## Warning - Memory

Image-processing uses immense amounts of memory, the bigger you are exporting, the more memory the scripts will need. You find a parameter "-Xmx10g" in the scripts, this tells java to use 10GB of ram. You may need to adjust those values.

## Google Maps

### Generating the pre-zoomed files for google maps

Some background: The process behind google maps is to display one image, then when you are zooming in, it replaces this image with 2x2 images containing more details, every one of those images may be replaced by another 2x2 image and so on. Each of those steps I'll call "a layer" you may also call it one level one zoom one zoom level or anything else.

This results in many pre-zoomed images, which are clustered:

```
1
=>
1      3
2      4
=>
1      4      7
2      5      8
3      6      9
```

For best results use an image with the matching sides, so  $3 \times 256 \times 3 \times 256$  if you want to use 3 layers and so on ( $2^{\text{layers}} \times 256$  pixels per side). This isn't needed, but the lower layer will not be 1:1 pixelwise if you aren't doing this. (5 layers =  $2^5 \times 256 = 8192$  pixels)

Use the script "slice\_for\_gmap.cmd" to generate those files.

The script also generates an "img"-folder and a index.html. You only need those two items.

The img-folder may contain thousands of images, don't be shocked, layer 7 would be  $128 \times 128 = 16.384$  images).

Just open the index.html and you should be able to look around in your map.

## Uploading

When you are uploading the files to a webserver, just use the index.html and the img-folder.

The google libraries will be loaded on the fly.