What it is:

- Source code in RealBasic 5.2 project file (v4.12) / RealBasic 5.5 project file (v5.02) and text, (the RealBasic compiler can be obtained as free demo from: www.realbasic.com for Mac and PC!)
- PC exe file: CC2Tools.exe, requires W2K / WinXP or newer
- This readme and an in-depth manual.

Release v5.02, July 29th, 2009
- for beta testing only, not for production -
please report to Mafi:
closecombat2@claranet.de
for suggestions/criticism/errors

credits and thanks: this tool was made by me, but was impossible without the help of ...
- www.realbasic.com for the great RB5 compiler,
- Chris Ellens for his excellent CCedit (CC2-Map-Editor for MacOS)
- Cappy-R for his Gadget-Catalog,
- Gerry Shaw (aka TinTin) for the file format descriptions and his texture converting PC tools,
- Escobar for his PC tools (GadgetX/I, IntrfaceX/I) so I was able to identify what I was looking for,
- Cpl_Filth for his excellent PC tools, for showing us how to patch the soldiers' colors and CC3-roof file formats,
- Mick (xe5) Conmy for his advice (BTD format etc.) and lot of file format descriptions,
- IChrist for his CC5-Elements table,
- Andrew Glenn (aka Naked Foot) for his CC2-MapMaking Guide,
- Kyle Scott 'Fish' for a lot of help,
- Han Bos (aka NL_Attila) for his Tank-Editing online guide at CSO,
- Vince Viaud for his LOStool and file format description,
- Ken Scott for his encouragement to do the job and for testing (he is the guy who called for the first version of gadget-expanding),
- JimmyD for his CC-toolshed at http://www.wargamer.com/Hosted/CCJimmyD/index.htm,
- Sulla for keeping www.cso.org online,
- and Adam 'The Man' who made the first steps in modding CC2.
- and Nembo for beta-testing

CC3 to CC2 map conversion

This is a short description on how to use CC2Tools on the Mac. The goal is to convert existing CC5, CC4 and CC3 (custom) maps to CC2 format to use them on the Mac. There are three problems:
- CC3-CC5 maps are encoded in a different graphics file format (LITTLE ENDIAN), CC2 graphics are encoded in Big Endian,
- CC3-CC5 map data files depend on a different terrain-element definition,
- CC5 maps are too large for CC3 (CC3 limit is 30x30 mega-tiles), CC5-CC3 maps are too large for CC2 (CC2 limit is 19x19 mega-tiles).

CC2Tools can do for you:
- convert all map graphic files of all CC-versions (from CC2 to CC5) into TARGA,
- convert from TARGA into CC2-BGMap, CC2-MMMap, CC2OVMap, CC2-Vehicle-Textures,
- shrink all CC2-CC5 map files in their size,
- and the main part is: shrinking and converting from CC5/CC3 into CC2 format in ONE step,
- and making the terrain-elements conversion from CC5 to CC2 or CC3 to CC2 or in any other way if you define an external translation table (TAB-seperated text file, MS-Excel-like).
- translation tables included for CC3 to CC2, CC5 to CC2.
- and it can do some manipulations to increase the number of teams,
- and can split/join the CC2 files Gadget0 & Gadget1.

To setup a new (different) LOS file (corresponding to the CC2-terrain elements) or to have better MMMap- & OVMap-graphics please use CCedit by Chris Ellens or the MacOS version of CCLos-Tool by Vince Viaud.

An example: trying to convert an original CC5 map to CC2. I will take Cherbourg Arsenal map. Launch CC2Tools. Open the dialog "Map converting/shrinking", make first tab panel visual. There are 5 steps to do (they are numbered):
1: set translation table to "<select external table>" an search for the file "CC5toCC2.txt" which came with this archive. It will be loaded into memory. Now set ouput format to "Close Combat 2" (THIS is ESSENTIAL).
2: select CC5 map data file for input: "CHRBRGAS.TXT" from original CC5 CDROM, the file format and size will be shown.
3: set the new dimension to Left = 4, Top = 34 with a width of 14 mega-tiles and a height of 13 mega-tiles.
4a: press the button "Shrink..." and enter new name for output file "Map400".
4b: press the button "Shrink LOS..." to search for corresponding LOS-file "CHRBRGAS.LOS" and enter name of new LOS-file "Map400.los" (quicker than making a new LOS-file).

5: press button "BGMap..." and search for CC5 background graphics "CHRBRGAS.BGM" from same source as above. Set output file name to "BGMap400".

6: press button "MMMap..." and search again for same CC5 background graphics "CHRBRGAS.BGM" from same source as above. Set output file name to "MMMap400".

7: press button "OVMap..." and search again for same CC5 background graphics "CHRBRGAS.BGM" from same source as above. Set output file name to "OVMap400".

8: press button "Roof..." and search for corresponding CC5 roof graphics "CHRBRGAS.RFM" from same source as above. Set output file name to "Roof400".

9: press button "CC4/CC5-btm -> Scenario..." and search for corresponding CC5 BTM scenario file "CHRBRGAS.BTM" from same source as above. Set output file name to "Scenario".

10. open new "Scenario" with Ms-Excel or SimpleText and change the map namefrom "Map102" to "Map400" (this must be done, if not, the map will not start),

11. create a folder named "1400", place the new "Scenario" and some suitable existing "AlOOB" and "AxOOB" files in it , too.

12. change your base file "Batnames" and add an entry map 400 as single battle and increase the number of battles (Ugur's method).

13. Place the folder "1400" in the folder " A Bridge Too Far/data/battles" of your full HD-disk installation of CC2, place the file "Map400" in the folder "A Bridge Too Far/data/maps", place the graphics files "BGMap400", "OVMap400", "MMMap400" and "Roof400" in the folder " A Bridge Too Far/graphics/maps".

14. Ready, start ABTF and you will have a new map with the lower portion of "Cherbourg Arsenal".

Before I had CC2Tools for MacOS ready,
this was the way to do it:

- as an example will serves the CC3-map "Quinn's Post" (a map by Kyle Scott "Fish")

Download this map from http://www.greatwargames.com/. I have asked the original author for permission to do the conversion and to post the result.

You will now have a file "Quinns.zip" on your HD. Open the archive and extract all containing files to a new folder using common tools like WinZip (or Stuffit Expander on a Mac), which can be downloaded from the internet.

In the new folder you will have 8 files:
- Quinn's Post
- Quinns.bgm
- Quinns.los
- Quinns.mmm
- Quinns.ovm
- Quinns.rfm
- Quinns.txt
- Turk Quinn's Post

The largest one ist the file containing the background image in CC3-format: Quinns.bgm. This will be our first target to be converted to CC2. You need two PC-tools from Gerry Shaw aka TinTin (http://www.organicbit.com/closecombat/): TextureMaker3 (TM3) and the earlier TextureMaker for CC2 v2. Both versions consists of two seperate programs, one for extracting the background graphic in TARGA-format from the *.bgm file and one for creating a *.bgm file from a TARGA-16bit graphic. This programs must be run in DOS mode.

First start to extract the background graphic from Quinns.bgm. The files Quinns.bgm and cc2tga.exe must reside in the same folder. Use the program cc2tga.exe, typing at DOS prompt: C:>cc2tga Quinns.bgm Quinns.tga

The resulting file will be a 17.3 MB TARGA graphic which can be edited with any modern graphic program. Remember the file format: uncompressed 16-bit TARGA!

Next to do is to determine the original map size in pixels. So open the file "Quinns.tga" with your favorite graphics editor. Retrieve the image size: 2400x2520 pixels. This image size is to large for Close Combat 2, but a usually value for CC3 - CC5 maps. CC2 can handle (with some exceptions as too many trees) maps with a size up to 2280x2280 pixels. This is valid for both CC2 versions, MacOS and Win95. As you can see, all the mentioned values multiplies of 120. Close Combat (CC2-CC5) maps must fit to a grid of 120x120 pixels, the so called "deployment tiles". Each "deployment tile" is divided into 3x3 "elevation tiles". Each "elevation tile is divided into 4x4 "terrain tiles". A "terrain tile" represents 10x10 pixels of the background image and has (in CC3-CC5) a seperate elevation value. (The scale of the background image is intended by the inventors of CC-series, Atomic Inc., as 5 pixels = 1 meter.) In CC2 all "terrain tiles" of a single "elevation tile" have the same elevation. As Naked Foot & Mick (xe5) stated in their CC2-MapFaq.doc, maps must have complete deployment tiles.

As shown above it is necessary to shrink the size of the map. And we must do it in steps of 120 pixels. For this example I suggest that it will be best to preserve the

upper part of the map. So cut off 720 pixels from the button. To reduce the width of the map (it will be a WW1-trench battle) cut off 240 pixels from both sides. The resulting image of 1920x1800 pixels must be saved in a new file with the format 16bit color, uncompressed TARGA. I suggest for this example to use the filename "NEWimg.tga". For Mac-users: we must save it as a 16bit color PICT-file, all other graphic formats can not be imported into CCedit!

And now we can convert it back to CC2 format, creating a new background image file. In CC2 the map files are numbered and have no individual names like in CC3-CC5. It is necessary to decide which slot number we should use. I suggest for this example slot 103. The nameof the file to becreated must be therefore "BGMap103". To create a valid "BGMap###" file we can use two tools: CCedit on the Mac or TinTin' TM v2 for CC2. To use it type in at the DOS prompt: C:>tgacc NEWimg.tga BGMap103 -bgmap
The resulting file is 6.5 MB large and ready for gameplay.

Because we have changed the background image, it is useless to transform the CC3-thumbnails "Quinns.mmm" and "Quinns.ovm". We must create our own new Map-Monitor-map- and Overview-map-graphics using our graphics editor.

Let's start with the Map-Monitor-map. Naked Foot assumed in his CC2-MapFaq.doc that the maximum size of a CC2-MMmap### file should be 144x106 pixels. For our example we must reduce the image size from 1920x1800 pixels down to a size with a vertical dimension less than 106 pixels. Dividing 1800 by 17 will result in 106. We must shrink the horizontal dimension in the same way: dividing 1920 by 17 will result in 113. The same calculation does CCedit (by Chris Ellens) on MacOS. Okay, open our file "NEWimg.tga" with your graphics tool, shrink the size of the image to 113x106 pixels and save it as an uncompressed 16bit TARGA graphic, I suggest the name "NEWmmm.tga". To create a valid "MMMap###" file we can use again both tools: CCedit on the Mac or TinTin's TM v2 for CC2. Type at DOS prompt: C:>tgacc NEWmmm.tga MMMap103 -mmmap

The next graphic is the overview map which is used in the battlemaker edit screen. The graphic is larger than an MMMap###. Naked Foot suggested that maximum size of a CC2 OVMap### file is 1142x646 pixels. Examining some of the original OVMap###s show that they often (but not all, for example OVMap100 is of 527x377 pixels at 14x10 deployment tiles) have size 904x646 pixels (for 14x10 deployment tiles) up to 1076x646 pixels (for 15x9 deployment tiles). For our example we must reduce the image size from 1920x1800 pixels down to a size with a vertical dimension less than or equal to 646 pixels. Dividing 1800 by 3 will result in 600. Same reduction for horizontal width: 1920 / 3 = 640. Or if we take the vertical limit 646 pixels we must use a width of 689 pixels. CCedit generates OVMap### files with smaller images: the maximum visible area in the battlemaker

screen is 594x369 pixels. It will be sufficient to provide the map with an OVMap###-image with a height around 369 pixels. CC2 will calculate its own visual impression, if the image is a little bit larger. CCedit would create an image of 402x377 pixels. For testing purposes only we will use the size suggested first: 640x600 pixels. Open our file "NEWimg.tga" with your graphics tool, shrink the size of the image to 640x600 pixels and save it as an uncompressed 16bit TARGA graphic, I suggest the name "NEWovm.tga". To create a valid "OVMap###" file we can use again both tools: CCedit on the Mac or TinTin's TM v2 for CC2. Type at DOS prompt: C:>tgacc NEWovm.tga OVMap103 -ovmap

No we will go to the terrain- and elevation datas, which are necessary to tell CC2 what kind of battlefield the image should represent. This datas are stored in text based files, editable with all common basic text editors and MS-Excel. The file format of this files in CC2 is simple: CR-delimited lines (0Dhex), datas in each line are seperated by TAB-chars (09hex). In CC3 up to CC5 these files slightly differs in having CRLF-delimited lines (0D0Ahex). But as the older versions of CCedit shows, MS-Excel and CC2 understand even LF-delimited lines in the files. Error messages only occurs with CRLF-delimited lines in the MacOS-version of CC2!

The datas are stored in the CC3-version of our example map in the file "Quinns.txt". Feel free to view it with Excel. The header in this file differs sligthly from those Map###-files in CC2. And the terrain-values (defined in the CC3-base-file "Elements.txt" / CC2-base-file "Elements") are similiar. The most important difference is that CC3-txt-files contain 16 elevation defeinition per line (each terrain tile has its own elevation). In CC2 there is only 1 elevation defined per line. I have made a simple tool for DOS "CC3-CC2.exe" which uses a translation table to convert the CC3-element values to CC2-element values. Just type in at the DOS prompt:
C:>cc3-cc2
The program will ask you for the name of the translation table and for the file to translate to CC2 (in this example type in: "Quinns.txt"). All files must reside in the same folder to work. The result is saved in the file "MAP-CC2". Rename this file to "NEW103". The statistic will show that only the values "1 = CC2-Hih Grass" and "31 = CC2-Wood Rubble" were changed.

The size of the resulting file "NEW103" is still fitting to the original CC3-size of the map. In our first step we had shrinked the graphics. So we must shrink the size of the datas, too. For this task I have made the little DOS utility "CCshrink.exe". Just type in at the DOS prompt: C:>ccshrink
The program will ask you for the file format, you must enter "2" for CC2. Then you must enter the name of the file to shrink: enter "NEW103". The program will than check the old map size and will ask you for the new dimensions. The old range is 1-60 horizontal and 1-63 vertical (in elevation tiles). We had cut off 240

pixels = 6 elevation tiles) at each side, so the new range horizontally will be 7-54. And we had cut off 720 pixels at the buttom = 18 elevation tiles. The new range vertically must be 1-45. After entering these values the program will do the rest, saving the result in a file named "MAP-SHR". Rename this file to "Map103". You can control this map with Excel if you like. And of course you can make refinements using CCedit (on a Mac) or TinTin' "MapMaker.exe"! For both programs it is necessary to determine first where the CC2-data folder resides, because the programs want to load the file "Elements". Than you can select our file "Map103" from the folder data\maps for editing. Before that you must place the new created file "BGMap103" in the folder graphics\maps.

In CC2 the elevation of the maps ranges usually between 10 and 45 meters. On the Mac CCedit can not handle elevations below 10. I do not know if it is suitable to create higher or lower elevations in CC2, so I recommend to change the elevation values. For this purpose I have made the little tool "CCelevat.exe". Type in at the DOS prompt: C:>ccelevat The program will ask you for the file you want to modify. For our example type in "Map103". The program will determine the file format and the range of the used elevations. In our example the elevation range will be 0-54. You must enter the new elevation range. For our example I recommend 10-45. The resulting file will be saved as "MapE-CC2". Rename it to "Map103".

Now we must create a suitable "Roof###"-file. Let us use Cpl_Filth's tool "RoofExt.exe"
(http://www.student.oulu.fi/%7Epsaastam/) to get the maps original interior graphics. Place this file and the CC3 file "Quinns.rfm" of this example in the same folder and type at DOS prompt:
C:>roofext Quinns.rfm
The little utility will extract all exterior and interior graphics from the *.rfm-file. The format will be BMP. In our example we will have 30 resulting files "ext000.bmp" - "ext014.bmp" and "int000.bmp" - "int014.bmp". Now I must describe two different ways on how to create the "Roof###" file:
1. for Mac: using CCedit the creating of the Roof-file is really simple. Open the desired map (our example: Map103) after having the files "Map###" and "BGMap###" ready. Load the generated files of the exterior and interior into your favorite graphics editor. Determine which exteriors are really present on your map after the size correction. Those who are not needed can be purged. Turn to CCedit and switch terrain editing off and roof editing on (Cmd-R). Now you are able to select the part of the background you want as a roof. Use Cmd-D to define it as a roof. The graphical content of this roof is transferred automatically to the clipboard by CCedit. Turn to your graphics editor and create a new picture from the clipboard. Paste in the suitable interior into this new picture from the original interior graphics. Put the whole new content of this new picture back to the

clipboard. Turn back to CCedit, switch to interior editing (Cmd-I), the as a roof defined area of the background will be shwon as a black hole. Paste the content of the clipboard in and you will have the new interior integrated. The exteriors for your roof file will be generated automatically by CCedit. Don't forget to save your work, the needed "Roof###"-file will then be created by CCedit. In our example: "Roof103".

2. for PC: there exists a good tool by Cpl_Filth: "Groof2.exe". For this tool (which is suitable for CC2-CC5) you need two images of the background: the original one and one image with the interiors pasted in. So make a duplication of your background image and paste in the by "RoofExt.exe" created interior graphics (in your graphics editor). Open both files with "Groof2.exe". Define the roof rectangles in "Groof2.exe". Remember: CC2-roofs are always rectangles! Save your work and the needed "Roof###"-file will be created. In our example: "Roof103".

Perhaps we must do further refinements using "MapMaker" or "CCedit" to get the correct datas for the interior graphics.

Now it is time to generate a new CC2-los-file. This file contains the Line-Of-Sight values of the map. It must be recalculated, the old CC3 file can not be used. Take Vince Viaud's tool "ccLos.exe" (downloadable from TinTin' site). Because this calculation will take a while it is recommended to do this as the last step, after the Map###-file is completely done. On the Mac you can use CCedit, too, and it is significant faster. On a PC you must put the program "ccLos.exe" in the root dir of CC2 (same folder as "CC2.EXE") and you must put the new Map###-file in the data\maps folder. In this example: take the file "Map103" into the correct folder on your HD and than type in at the DOS prompt:
C:>cclos
The program will ask you for the map number, type in "103". The rest will be done by the program and the result will be saved in the file "Map103.los" in the folder data\maps.

The last we must create is a "Scenario" file for the battles folders. In our example we must create this file for the folders data\battles\1103 for single battle play and data\battles\103 for campaign play. We can extract the names of the intended victory locations from the file "Quinn's Post" by using a common HexEditor (Mac or PC). Use an existing "Scenario" file to see where the names must be pasted in. Don't forget to set the correct x- and y-values for the victory locations. In the second part of the "Scenario" file the starting setup Axis/Allies/Neutral must fit to the size of the map, otherwise the loading of the map in ABTF will fail. If we use the original "Map100"-scenario file to transform it into our new "Map103"-scenario file we must add 2 columns at the rigth side of

the table and 5 lines at the bottom. If we want the first victory location to be "blowable" by the Axis we can set a bridge blow time greater than 0. That's all.

Again I made a little utility helping to create CC2 Scenario files from CC3 input: "CCscenar.exe". To use it you must convert the name of the file "Quinn's Post" to something shorter, for example "Quinn.scn". Place this file together with the little program in the same folder. Just type in the program's name at the DOS prompt. In our example enter "103" for the map number and answer with "yes" when the program asks if you want to convert an existing CC3 file. Then you have to enter the horizontal and vertical dimension of our new CC2 map: type in "16" and "15". And the resulting file is "SCEN-103". Rename it to "Scenario" and place it in the folders data/battles/103 and data/battles/1103. I made further corrections with the datas in this file: because we had cut off 2 deployment tiles from the left side of the original map I reduced the X-values of the victory locations by 2! Recommended bridge blow time is 1500 (= 2 minutes). I used side 0 for initial startup and moved the starting lines (= neutral zone) to the middle of the map.

Appendix 1 - LOS file problems

on 01.12.2002 01:17 Uhr written by Victor Barone (victor.barone@globo.com):

> Hey Mafi...
> Thanks for yur explnaion about trees/elevation etc...
> Now.. (tell me if I perturb yu) I have other question... working on map...
> seems the los calculate is more big... all is allright... but when I try
> use the map I receive that information
>
> Close Combat Error
> out of comand data space
>
> Its cause the map is so big on los calculate? Or many trees maybe? ita a
> original 1800x1800.. I change for 1320x1560 to cut some buindilngs....
>
> PS: that hedgerous thing.. I try manipulate cc3-cc2 txt but dont understand...
> if yu have time some day can yu change that? hedgerous for bush? If not
> no problem..
>
> Thanks anyway man..
> Yu help me a lot !!!!
>
> Cheers
>

Hello Victor,

LOS calculation takes a lot of time, more time if you have more obstacles on the map (buildings, wrecks, trees, hedges ...).

its new to me that memory problems can occur on a PC too. On the Mac a known problem. It cannot resolved for all times. There are memory limitations inside CC2-ABTF. And as we see now this is common for PC- and Mac-version. I believe (but it is only my theory) that ABTf has a heap/stack combination during runtime which is limited. It can be expanded by allocating more memory to the program, but even if you allocate more than 100MB to it, some very large maps with many trees/buildings will not work. It seems to me that the limitations is always around the number 50: not more than 50 roofs if you have a large map or many trees, not more than 50 battles possible, not more than 50 maps perhaps. It is not proven, but some memory elements during runtime are not allocated in the heap/stack portion of the program's RAM but reside probably in RAm of fixed size (like arrays of given size in PASCAL).

For the Ramelle-map I solved this problem by removing some roofs. For the Firebase-map I shrinked the map size and removed a lot of trees. You must make some tests to find out which will be more usefull in the individual case.

In the attachement you will find a new translation table which will translate CC3-hedge in CC2-Bush. One line is changed (element No 14). Keep inmind that the input must be a CC3 file. If you want to convert existing CC2 files with my tool you must set up a new translation table. But I think it will be easy to do in MS-Excel as well.

Mafi

Appendix 2 - Trees

written 30.11.2002 12:36 Uhr by Victor Barone (victor.barone@globo.com):

> Hello Mafi...
> Im working in sveral maps have castre done.. and work on snow2... I note
> a commn problem... when a reduce the maps for playple cc2 consitiond (using
> the photoshop)... loos like the trees desapear. I see only the tree
> shadows....
> I see this problem in my yelnia coverted map too... first IO think is some
> distorcion on map reducion.. but now seems is some tree files no compatible

> with cc2... is that?
> Thanks
> Braz
>
>
> Hey Mafi...
other question..
about elevation...

You said in the manual yu send me that quinns example have elevation range
0-54. You sugest 10-45.
I have to use 10-45 for all maps?
Example. I have a map here where elebvation range is 0-8... what hapen is
I use 10-45?
Cheers

>
>

Hey.. other thing...
When I put 10-45 elevation range.. los program work...

A question. Some normandy maps have that hedgeropus bushs... inf ans tanks
CANT cross it right? so, if we dont make some exit units can stay closed
there.. its right? In that way some maps are really impossible to convert.

Other question.. the tree files are really weird for me.. ther are read
separated from the map right? but some maps, after converted, domt show
tree.. (pnly shadows).... whatr I need do?
>
>

Hi Victor,

good to see that you are getting forward in high speed making new maps.

here are the answers to your actual questions:

- trees in Close Combat series
Trees in CC2, CC3, CC4 and CC5 are obstacles on the maps which are pasted over
the background graphics of a map during gameplay. Therefore the background
graphic of a map must be free of trees. The data file (in CC2: Map###; in CC3-
CC5: *.txt) contains the terrain's values. If there is a tree encoded in the data file,

it can be of several types. Some (but not all in CC2) of the different types of trees correspond to a seperate tree graphic in the TERRAIN file, which is part of the graphical environment of the CC2-CC5 program.

The map designers of the original CC2 maps placed a graphical element on their maps to indicate where a tree should be defined. This element contains of a TRUNK, the center-shadow of a trunk (which is often dark grey/black) and (due to the time where ABTF took place: fall) and leave graphics around the trunk. Most of the custom map designers did not place a trunk on their graphics (as I did, too), because you must respect the terrain element grid to place the trunk in the middle of an terrain element. Cpl_Filh made a tool which can do this all automatically for CC3: placng trunk and shadow on the graphic AND adding the terrain value to the data file.

In CC3-CC5 it is much more difficult: Atomic decided to place the tree graphics into the TERRAIN file WITHOUT shadows! So CC3-CC5 map makers are forced to place tree shadows on their maps, otherwise they will have no shadows on it. This is the reason why conversion of CC2 maps to CC3 needed more graphical artwork than the way converting CC3 maps to CC2 maps. But it is another problem: European maps must have European style shadows and Pacific maps must have palm tree shadows. It is not usefull to play a European map with a Pacific TERRAIN file.

In CC2 it doesn't matter: you can play every map with every TERRAIN file, you will always have the correct trees and shadows on it. But it is the same method CC2 and CC3-CC5 are using. Theoretically the CC3 TERRAIN file can contain the tree shadows, too. But up to now no one had tested it yet. It is the same thing with the VictoryFlag animations: same principle in CC2 and CC3-CC5 (with shadows).

So we have to look at the tree graphics themselve: CC2 trees are fall trees in green and orange/red/yellow. There are dead tree graphics, too. And two conifere tree graphics not used by CC2. In CC3 the tree graphics are summer trees in ligth green, several conifere trees and trees for winter with snow on them. The summer trees are a little bit smaller than those of CC2. So if you convert a CC3 summer map to CC2, you will have "double" shadows (those of the map and those generated by the TERRAIN file). But they fit good enough together. Even the CC3-Pacific/Africa maps have their palm tree shadows (invented by the unforgotten George Thanos) in a way they will fit together with my TERRAIN files for CC2-Pacific/Africa/Nam. The reason is that I have first took a look at them before I made my own graphics, so my palm trees are a little bit larger then those made by George.

All in all you should have trees on your map when you have used my conversion tools. The trees of CC3 not available in CC2 (winter trees, conifere trees) are translated into "Big tree" or "Medium tree" terrain values.

If you see now trees after conversion you can add them to the Map### file by hand using MapMaker.exe by TinTin. It is easy enough because the CC3-shadows lead the way where to place the trees.

- the elevation question:
CC2 took place in the Netherlands, a really flat country (the name says it exactly, in French: "les pays bas").

If you look at original CC2 maps, you will see that the lowest elevation is 10 m. But there are some errors in the map's datas (for example in Map100: one single elevation 2 m, an error, it must be 20 m). Chris Ellens (maker of CCedit) reported that his tool works only with values up to 45 m correctly. So I recommend to use elevation between 10 and 45 meters. You can use my tool CCelevat.exe as an "Elevator": it can lift the entire maps values up or down and can shrink the elevation range (of course you can use for the same effect MS-Excel). I do not know if vehicles or soldiers in CC2 can climb steep slopes (elevation differences of more than 2 meters). And I do not know if CC2 can really handle elevations correctly which are lower than 10 or higher than 45 meters.

To the example: if the elevation range is 0-8 m in CC3, I recommend to use in CC2 an elevation range of 10-18 m (simply adding 10 to every elevation value). If the elevation range is 60-80 m, I recommend to lower the whole map down to a range of 20-40 m.

- the hedgerow question
when I have understood the problem correct, you want to have those hedges crossable by tanks. I do not know if CC2-terrain element "Hedgerow" prohibit tank crossing. This must be tested on a map. It exists a map with hedgerpws on them, I believe its name was "HEDGEROWS". Look into its datas.

Feel free to make your own version of the file CC3-CC2.txt to get better results while making map conversions. If "Headgerow" is too heavy for tanks, replace this value by "Bush". Thick enough to protect from rifle fire, but passable by tanks.

In reality the battle in Normandy's headges was terrific for the tanks because they were caugh in those traps and shut up easily by Panzerfausts. I think that the CC2/CC3 element files just rebuild this situation good and those maps are really intended not for tank battles.

- stopping tanks from crossing
the only CC2-terrain element which really prevent vehicles from crossing parts of a map is the value "No Vehicles". Tested on my Minqar Qaim maps succesfully. There the tanks cannot cross the stone rubble area or if they are placed in it they are prisoner of this area. But I believe this was not your question.

So far so good, I hope this will help you a little bit. If you want to know more about trees look into my guide concerning the TERRAIN file. I can be edited with Cpl_Filth's SprTool.exe.

Cheers

Yours truly Mafi

Hope this guide helps and encourages you in CC2 map making.

Mafi
closecombat2@claranet.de
November 23rd, 2002

--

Our Goal was: expanding the number of teams in CloseCombat2 beyond the 133 limit.
The result is the tool in this archive, feel free to distribute it and/or expand it unless you don't remove the copyrigth notice and the credits to the other modders!

My personal investigations – not error free

The TARGA Header of extracted Gadget#s

0000 // first integer 0
0200 // second integer 2 = color depth? = 16bit
0000
0000
0000
0000 // 4 integers with value 0
xxxx // integer representing width in pixels
yyyy // integer representing height in lines

```
1001 // integer of unknown purpose 1000 or 1001 or other value
data


FileFormat of the Gadget# files (BigEndian = reverse byte format /
Motorola style)

Header of 8 bytes:
47414447  // string "GADG"
0000xxxx  // long int: number of graphical gadgets in this file
Offset table, each entry 10 bytes long:
xxxx // integer: visible width = number of pixels
xxxx // integer: height = number of lines
xxxx // integer: length in bytes of each line = (graphics width) /
2
xxxxxxxx  // long int = offset of data entry from start of file

Data entries: in reverse line format: bottom line first!!!!, and
in reverse byte format!!!!

You can calculate the size of each entry: height * length (in
bytes)

The "visible width" (in hex) is used in the filename of the
extracted TARGAs by Escobar's tools GadgetX/Gadget (and my new
tool, too)


Mafi
closecombat2@claranet.de
first provisional release: Sept. 28th, 2003



my eMail with KenScott:
```

Von: Mafi <closecombat2@claranet.de>
An: Ken Scott <KenScott@hestanroad.fsnet.co.uk>
Datum: Montag, 29. September 2003 22:54 Uhr
Betreff: Re: Trying to crack the "133 barrier" - next problem

```
Hi Ken,

next problem:

null problemo with extending Gadget0. Good. But:

when inserting in Gadget1 at entry 806 a larger gadget for Team133, this larger gadget is
available for Team133, but from then on Team000 has no small gadget but the large gadget
of Team133. And at the end: all small gadgets for the team are shift one entry lower
(that means: all are wrong).
```

So it doesnt make any sense to insert an additional gadget for team133 at entry 806 unless we manage to tell CC2 to start from entry 807 with the smaller gadgets. We have to look inside the executable if we could find there something.

In the meantime it might be usefull to accept that the additional team133 will have the smaller gadget of Team000.

We will see ...

and a few minutes later I found the position:

in Mac version (not for sure for windows) there are in the object code the names of the original high level language sub-routines they made at Atomic:

The first occurence of the value 806 = 326hex is in a sub-routine called CheckTeamMonitorForHotMouseActor
in Mac version v2.0b: Offset in the obj. code is: 03FD02h
(of course the position in windows version will be somewhere else)

and this is the point: changing here 326h to 327h and we will have the correct sequence of small gadgets for all teams. YES. YES. It can be done by a simple patch program to modify the executables (like did for the SoldierColorPatch), so we must not violate Microsofts copyrigth during publication.

From my experience with the soldier color patch, I think you will be able to find the right location in the Win-EXE: because the byte sequence is different between Mac and PC, you have to look for the sequence 2603h
(I made a look: approx. 10 occurences, simply test them step by step). The sequence 0326h is not in the file CC2.exe at all.

No its real hardcore hacking. Hope we made it for both versions.

Cheers

Mafi

And I did it, here are the results:

in Mac-Demo version: Offset in the obj.code is: 03F8F6hex
in Mac ABTF v1.0 version: Offset in the Obj.code is: 03FD16hex

in PC ABTF v2.0 version from CD: Offset 2603hex in the Obj code is: 0609BAhex
(that is the 4th occurence) (program version 97.0.9.2202)

in PC ABTF v2.0b (last update): Offset 2603h in the obj. code is: 060967hex
(that is the 2nd occurence) (program version 1998-01-19)

My personal suggestion to make 3x3 km maps as of Sept. 2006:

It is possible to join maps up to "unlimited" size using the existing CC2Tools.exe v4.12.
Example: you want to have your 3x3 CCM map = 120x120 deployment tile map
to be created by joining 9 standard RtB-maps (40x40 deployment tiles).
Let me call these maps "A", "B", "C" for the three maps of the first upper "line",
maps "D", "E" , "F" for the middle "line" (= map "E" is the center of your new 3x3 map),
and maps "G", "H", "I" for the third line below. Map "I" is the one in the lower right
corner.

- Ensure to have all these 9 maps ready with *.txt, *.bgm and *.rfm.
- Launch CC2Tools.exe, go to section "map-convert/shrink/rotate/join", subsection
"EXPAND".
- press "Select file..."-button to select map "I",
- add "at the left side" 240 elevation tiles by selecting this value from the popupmenu
on the left,
- press button "Expand..." to expand the map "I"'s *.txt file by 80 deployment tiles (=
240 elevation tiles),
- do the same for map "I"'s *.bgm and *.rfm by pressing the according buttons,
- and save the new expanded map under a new name "I-expanded.txt", "I-expanded.bgm", "I-
expanded.rfm".

Expanding is a fast process. Now you have the room reserved in "I-expanded" to place
maps "G" and "H" over there at their intended position by switching to subsection "JOIN":

- go to subsection "JOIN",
- keep "at top of map = 0", keep "at the left side = 0",
- press button "Join two maps *.txt",
- select as first map "I-expanded.txt" (the map which one is larger must be selected
first!),
- select as second map "G"'s *.txt
- and do the same with *.bgm and *.rfm by pressing the according buttons below,
- and save the joined map under the new name "IplusG.txt", "IplusG.bgm", "IplusG.rfm".

Now paste over the map "H":

- stay in subsection "JOIN"
- keep "at top of map = 0", change "at the left side = 120" to place "H" between "G" and
"I",
- press button "Join two maps *.txt",
- select as first map "IplusG.txt" (again the map which one is larger must be selected
first!),
- select as second map "H"'s *.txt
- and do the same with *.bgm and *.rfm by pressing the according buttons below,
- and save the joined map under the new name "IplusGplusH.txt", "IplusGplusH.bgm",
"IplusGplusH.rfm".

Now you have the lower line ready stichted together. Keep the results as backups until
you are satisfied with all.
Next task is to expand "IplusGplusH" to get room for the middle line: adding 40
deployment tiles (= 120 elevation tiles)
at the top:

- go back to subsection "EXPAND".
- press "Select file..."-button to select map "IplusGplusH",
- add "at top of map" 120 elevation tiles by selecting this value from the popupmenu on
top,

- press button "Expand..." to expand the map "IplusGplusH"'s *.txt file by 40 deployment tiles (= 120 elevation tiles),
- do the same for map "IplusGplusH"'s *.bgm and *.rfm by pressing the according buttons,
- and save the new expanded map under a new name "IGH-expanded.txt", "IGH-expanded.bgm", "IGH-expanded.rfm".

Now you have the room reserved in "IGH-expanded" to place
maps "D", "E" and "F" over there at their intended position by switching to subsection "JOIN":

- go to subsection "JOIN",
- keep "at top of map = 0", keep "at the left side = 0",
- press button "Join two maps *.txt",
- select as first map "IGH-expanded.txt" (the map which one is larger must be selected first!),
- select as second map "D"'s *.txt
- and do the same with *.bgm and *.rfm by pressing the according buttons below,
- and save the joined map under the new name "IGHplusD.txt", "IGHplusD.bgm", "IGHplusD.rfm".

Now paste over the map "E":

- stay in subsection "JOIN"
- keep "at top of map = 0", change "at the left side = 120" to place "E" right to "D",
- press button "Join two maps *.txt",
- select as first map "IGHplusD.txt" (again the map which one is larger must be selected first!),
- select as second map "E"'s *.txt
- and do the same with *.bgm and *.rfm by pressing the according buttons below,
- and save the joined map under the new name "IGHplusDplusE.txt", "IGHplusDplusE.bgm", "IGHplusDplusE.rfm".

Now paste over the map "F":

- stay in subsection "JOIN"
- keep "at top of map = 0", change "at the left side = 240" to place "F" right to "E",
- press button "Join two maps *.txt",
- select as first map "IGHplusDplusE.txt" (again the map which one is larger must be selected first!),
- select as second map "F"'s *.txt
- and do the same with *.bgm and *.rfm by pressing the according buttons below,
- and save the joined map under the new name "IGHplusDplusEplusF.txt", "IGHplusDplusEplusF.bgm", "IGHplusDplusEplusF.rfm".

Now you have 2/3 of the map ready stichted together. Keep the results as backups until you are satisfied with all.
Next task is to expand "IGHplusDplusEplusF" to get room for the top line: adding again 40 deployment tiles at the top:

- go back to subsection "EXPAND".
- press "Select file..."-button to select map "IGHplusDplusEplusF",
- add "at top of map" 120 elevation tiles by selecting this value from the popupmenu on top,
- press button "Expand..." to expand the map "IGHplusDplusEplusF"'s *.txt file by 40 deployment tiles,
- do the same for map "IGHplusDplusEplusF"'s *.bgm and *.rfm by pressing the according buttons,

- and save the new expanded map under a new name "IGHDEF-expanded.txt", "IGHDEF-expanded.bgm", "IGHDEF-expanded.rfm".

Now you have the room reserved in "IGHDEF-expanded" to place
maps "A", "B" and "C" over there at their intended position by switching to subsection "JOIN":

- go to subsection "JOIN",
- keep "at top of map = 0", keep "at the left side = 0",
- press button "Join two maps *.txt",
- select as first map "IGHDEF-expanded.txt" (the map which one is larger must be selected first!),
- select as second map "A"'s *.txt
- and do the same with *.bgm and *.rfm by pressing the according buttons below,
- and save the joined map under the new name "IGHDEFA.txt", "IGHDEFA.bgm", "IGHDEFA.rfm".

Now paste over the map "B":

- stay in subsection "JOIN"
- keep "at top of map = 0", change "at the left side = 120" to place "B" right to "A",
- press button "Join two maps *.txt",
- select as first map "IGHDEFA.txt" (again the map which one is larger must be selected first!),
- select as second map "B"'s *.txt
- and do the same with *.bgm and *.rfm by pressing the according buttons below,
- and save the joined map under the new name "IGHDEFAB.txt", "IGHDEFAB.bgm", "IGHDEFAB.rfm".

Now paste over the map "C":

- stay in subsection "JOIN"
- keep "at top of map = 0", change "at the left side = 240" to place "C" right to "B",
- press button "Join two maps *.txt",
- select as first map "IGHDEFAB.txt" (again the map which one is larger must be selected first!),
- select as second map "C"'s *.txt
- and do the same with *.bgm and *.rfm by pressing the according buttons below,
- and save the joined map under the new name "IGHDEFABC.txt", "IGHDEFABC.bgm", "IGHDEFABC.rfm".

Thats it. Joining is a time consuming process, but it will not consume much RAM, because only two elevation tiles
will be in RAM at the same time, no matter how large the two maps might be. For me it took 15 minutes
to join a "4x4" (=160x160 deployment tiles) with a standard RtB map on a real PC under WinXP.

Now you might ask "why not expanding map "I" in one step to 3x3 size"???
The reason is simple: CC2Tools.exe's internal limit of popupmenu-entries is 255. For 3x3 it is possible to do this.
But for 4x4 maps, so I decided to show you an "universal" way.

What you must know: while joining, the WinXP screen-refresh-cycle will terminate, and the window will go white,
even the progesbar will no longer show any activity, but dont worry, the program is still working.
Take a cup of coffee (like in the early days).