

# **mapPlug English Guide**

Esteve Boix

**COLLABORATORS**

|               |   |                   |                  |
|---------------|---|-------------------|------------------|
|               | <i>TITLE :</i><br>mapPlug English Guide |                   |                  |
| <i>ACTION</i> | <i>NAME</i>                             | <i>DATE</i>       | <i>SIGNATURE</i> |
| WRITTEN BY    | Esteve Boix                             | February 14, 2023 |                  |

**REVISION HISTORY**

| NUMBER | DATE | DESCRIPTION | NAME |
|--------|------|-------------|------|
|        |      |             |      |

# Contents

|          |                              |          |
|----------|------------------------------|----------|
| <b>1</b> | <b>mapPlug English Guide</b> | <b>1</b> |
| 1.1      | Welcome                      | 1        |
| 1.2      | Introduction to mapPlug      | 1        |
| 1.3      | Features                     | 2        |
| 1.4      | Install                      | 2        |
| 1.5      | Relative directories         | 2        |
| 1.6      | What do I need ?             | 3        |
| 1.7      | Using mapPlug                | 3        |
| 1.8      | Using mapPlug                | 4        |
| 1.9      | Using mapPlug                | 5        |
| 1.10     | Using mapPlug                | 6        |
| 1.11     | Registering mapPlug          | 7        |
| 1.12     | Greetings                    | 7        |
| 1.13     | Future                       | 7        |
| 1.14     | New                          | 8        |
| 1.15     | History                      | 8        |
| 1.16     | Disclaimer                   | 10       |
| 1.17     | MUI                          | 13       |

---

# Chapter 1

## mapPlug English Guide

### 1.1 Welcome

mapPlug v1.41

An HTML client side image map editor

What is **new** in this version

©1998/99 Esteve Boix

[Introduction](#)

[Requirements](#)

[Features](#)

[Disclaimer](#)

[Installing mapPlug](#)

[Using mapPlug](#)

[Registering << Freeware](#)

[Getting mapPlug](#)

[Greetings](#)

[About MUI](#)

[History](#)

[The future](#)

Please note that this version has not been extensively tested... Use it at your own risk.

### 1.2 Introduction to mapPlug

Introduction to mapPlug

mapPlug is a MUI-based program that will help you to create client side HTML image maps.

An image map in HTML is a way to define "hot spots" in an image. Clicking them will bring you to a new page in the browser. For example:

```

```

```
<map name="map">
```

---

```
<area shape="poly" href="0.htm" coords="41, 139, 37, 161, 50, 185, 90, 190, 88, 163, 64, 148">
<area shape="rect" href="1.htm" coords="10,10,390,60">
<area shape="rect" href="2.htm" coords="10,70,390,120">
</map>
```

This code tells the browser to display an image that has two hotspots, two rectangular shapes defined by the coordinates given by each `<area>` tag.

As you can see, it's a kind of mess. It's even worse when you're not dealing with rectangular shapes, but circles or even multi point polygons.

mapPlug offers a simple way to do this. You load the image (using the AmigaOS v3+ datatype system, so any format for which a datatype exists can be loaded) and draw the shapes with the mouse directly over it. Once defined, the shapes are fully editable (you can move them or resize them for example), so you can adjust them to your needs without having to deal with coordinates.

From v1.1+, mapPlug offers a zoom function, so you can view the image with more detail to adjust the shapes.

You have two string gadgets where you can set the href parameter (i.e., where does the hot-spot point) and the alt parameter (the alternate text for non graphical browsers).

## 1.3 Features

mapPlug's features

Here goes an small list of features:

- Map creation completely GUI driven, very easy to use
- Shapes (rectangles, circles and polygons) are *fully* editable once created
- Zoom upto 16x.
- mapPlug's window is an AppWindow. Simply drag an image icon on it and it will be loaded and displayed, ready to be mapped.
- Support for CyberGraphX/P96's v43 datatype (thus working on high/truecolor bitmaps in high/truecolor screens)
- Correct handling of **relative dirs**
- The HTML code for the maps can be saved into a text file or copied into the clipboard
- Maps can be saved and loaded in a custom format, so you can modify it wherever you want

## 1.4 Install

Installing mapPlug

Although mapPlug will run directly from the dir, you may want to install it using the provided installer program.

Remember that it requires BetterString.mcc (by Allan Odgaard). If you don't have it installed, you can find it in the MUI/ directory inside the tP dir.

You can always download the latest version of BetterString.mcc in <http://www.diku.dk/studentd/duff/>.

## 1.5 Relative directories

Relative dirs

Relative dirs are an important concept in HTML docs creation. Here goes an small (and I hope helpfull) explanation.

When creating a new web site, usually all the file references (in images, links or whatever) are done relative to an absolute path. For example, imagine the following directory tree:

Work:

MySite (dir)

index.html

concept.html

images (dir)

page.html

myimage1.png

myimage2.png

myimage3.png

Ok.

You create the file 'index.html', and you want to make a link to 'concept.html'. Since they are in the same dir, there is not need to specify the full path for 'concept.html'. The link would look like:

```
<a href="concept.html">Link</a>
```

For example, a link from 'index.html' to 'page.html' (which resides in 'images') would look like:

```
<a href="images/page.html">Link</a>
```

A different case is when you want to link from 'page.html' to 'index.html', which resides one level up. This link would be:

```
<a href="../index.html">Link to index</a>
```

Note the '../', that tells the browser to search the linked document one level up from the current directory.

Uf... I hope I've explained myself :)

Well... mapPlug handles all this cases correctly (I hope). The only thing you have to do is take care to specify the path from where mapPlug should create the relative links. It may or it may not be the same directory of the image you're mapping. Take this into consideration when creating your map.

## 1.6 What do I need ?

What do I need ?

In order to run mapPlug, you need:

- An Amiga (or compatible :) computer with at least AmigaOS v3.0.
- MUI 3.8 correctly installed.
- BetterString.mcc (included)

BetterString.mcc is © 1997 Allan Odgaard (duff@DIKU.DK, <http://www.diku.dk/students/duff>).

- The datatypes for the images you plan to load (usually JPEG or PNG... You do use PNG, don't you ?).
- mapPlug works better in an 1:1 screen resolution like Multiscan or DblPal. Anyway, it will work on a PAL 640x256 screen (you'll see that the toolbar may no be visible due to space limitations).
- Some maps to create :)

## 1.7 Using mapPlug

Using mapPlug

mapPlug has only one window which contains nearly all the options of the program.

The program is (I hope) intuitive and easy to use, but anyway here goes a little description of each option:

**The main window**

**The menus**

**The prefs window**

## 1.8 Using mapPlug

Export to clip:

Saves the current map into the clipboard,so you'll be able to insert the map into any text editor by simply pasting it.

Export to file:

Pops up an ASL requester, so you can specify an ASCII file where mapPlug will save the current map.

Reference dir:

All the relative directories are created with this base path. It's automatically set to the current image dir, but of course you can change it whenever you want.

It should be the dir where the HTML file that will contain the map resides.

Take a look at [relative dirs](#) section.

Image file:

This is the current image. Whenever you specify a new image, the current map will be cleared (note that clearing the image currently deletes all the shapes) and the image will be loaded into mapPlug.

The reference dir will be also automatically set to the image dir.

Alt:

Alternative text for the image for non-graphical browsers. It's a good idea to set it.

Map name:

This is the name that will identify the map in the HTML code. You should set it before exporting the map (mapPlug will remind this to you :)

Toolbar:

The toolbar contains the icons that gives access to mapPlug's visual map editor. From top to bottom:

New rectangle:

Creates a new rectangular shape. Click over the image where you want it to start, and then click where you want it to end :)

Pressing "Return" finishes the shape.

Pressing "Space" cancels the creation process and deletes the shape.

New circular shape:

Creates a new circular shape. Click over the image where you want it the center, and then specify the radius.

Pressing "Return" finishes the shape.

Pressing "Space" cancels the creation process and deletes the shape.

New poly shape:

Creates a new polygonal shape. No limit in the number of edges other than the available RAM and your patience.

Just click where you want new point to be created, and press "Return" (or doubleclick) when finished.

Pressing "Space" cancels the creation process and deletes the shape.

Clone shape (RAmiga+"c"):

Creates a clone of the current shape. The new shape is moved an offset of 20 pixels. Not only the control points are cloned, but the name, target, href...

Clone shape (RAmiga+"y"):

Creates a clone of the current shape. The new shape is moved an offset of 20 pixels. Not only the control points are cloned, but the name, target, href...

---

Move Mode (Key "M"):

Clicking any point of a shape and dragging the mouse will make the shape move following the mouse.

Edit Mode (Key "S"):

Clicking any point of a shape and dragging the mouse will make this point move following the mouse.

Delete shape (Key "Backspace"):

Obvious (isn't it ???).

Please note that there is no "undo" function yet.

Zoom in (RAMiga+,"):

Zooms into the image. You can use the menus, too.

Zoom out (RAMiga+ "."):

Zooms out the image.

HRef:

This is where should the current shape point. It's automatically set to the image's directory.

Alt:

Alternative text for this shape for non-graphical browsers, such as Lynx. It's a good idea to set it :)

Target:

Target parameter for frames

## 1.9 Using mapPlug

Follows a brief description of the current menus of the program:

Project -> Clear -> All shapes :

Clears all the shapes of the imagemap

Project -> Clear -> Everything :

Clears all the shapes of the imagemap and the image itself

Project -> Load shapes:

Loads a shapes file (#?.mpf format). It contains the filename of the image and the shapes.

Project -> Save shapes:

Saves the current imagemap info a shapes file (#?.mpf format).

Project -> Export map -> To clipboard:

Copies the imagemap HTML code into the clipboard, so you can easily include in into your page using the paste command in your editor.

Project -> Export map -> To file:

Saves the imagemap HTML code into a user-defined ASCII file.

Edit -> New rectangle:

Create a new rectangular shape into the map (take a look at the 'New rectangle' button of the toolbar).

Edit -> New circular:

Create a new circular shape into the map (take a look at the 'New circle' button of the toolbar).

Edit -> New polygon:



Create a new polygonal shape into the map (take a look at the 'New polygon' button of the toolbar).

Edit -> Add point:

Adds a new point to the current polygon. The point is added "after" the currently selected point (drawn by mP as an outlined rectangle).

Edit -> Remove point:

Removed the current point from the polygon.

Edit -> Edit points:

Clicking a control point of any shape and dragging the mouse will move this point, not the whole shape

Edit -> Edit points:

Clicking a control point of any shape and dragging the mouse will move the whole shape, not the point

Edit -> Delete:

Deletes the current shape of the map

Edit -> Clone:

Creates a duplicate of the currently selected shape, and moves the copy and offset of 20 pixels (if possible). This can be very useful to create menus.

Specials -> Convert to polygon:

This function will convert the current shape into a polygon. If the shape is a circle, you may want to set the number of points that mapPlug will use to create the polygon. The more, the better, of course.

Specials -> Rectangle array:

This function will subdivide the current shape -a rectangle, of course- into an array of rectangles. You can set the number of columns and rows by dragging the sliders to the desired values.

Specials -> Subdivide polygon:

This function will double the number of points of the current shape -a polygon, of course-. The points are inserted by using linear interpolation.

View -> Zoom in:

Zooms into the image (up to 16x)

View -> Zoom out:

Zooms out the image

View -> Reset zoom:

Set the zoom of the image to 1x.

## 1.10 Using mapPlug

GUI page

Shapes color:

Color of the shapes when they are unselected.

Grid color:

Color of the grid

Grid from zoom level:

mapPlug will draw a grid when you set a zoom equal or greater than the value you set here

---

Toolbar position:

You can set the position of the toolbar on the main window with this cycle

Centered toolbar icons:

The icons of the toolbar will be centered if this option is active

HTML page

Default map name:

Default map name :)

Default img dir:

When starting mapPlug or clearing the map, the program will set this directory as default for the images

Export maps in lowercase:

If set, this option will make mapPlug generate HTML code for the maps in lowercase

## 1.11 Registering mapPlug

Registering mapPlug

mP v1.2 is freeware.

Thanks a lot to all the people that has registered mapPlug and my other programs.

Special thanks go to Paul Compton for his continued support :)

## 1.12 Greetings

Greetings

I would like to thank:

- All the people that have registered this and my other programs, for their continued support and suggestions.
- My brave betatesters team: Fredrik Zetterlund, Christian Kemp, Robert Nyberg, Phillip Goddard, Paul Compton.
- My girlfriend (:D), for all her infinite patience with me...
- All the people that registered this and my other programs. Without your support, I could not keep up with this work :)

All you made me feel that programming the Amiga is more than fun

This program has been completely developed using the AmigaE compiler v3.3a from Wouter van Oortmerssen, and requires MUI (© Stefan Stuntz) and the BetterString.mcc (from Allan Oodgard, thank you again).

## 1.13 Future

mapPlug's todos

mapPlug is not a finished product... I'll continue working on it (if there is interest, of course). Here goes a list of ideas:

- Integration with webPlug. This is the second most requested feature (well... now that zoom is here, this becomes the first one). I'm working on it, but you'll have to wait for new versions of both programs.
  - REXX port
  - More editing tools (rotate, scale, etc...)
  - Allow shapes to "exist" outside the image (?)
-

- Undo function
- Locale support

The list is longer (actually quite longer). Just register and wait :)

I'm waiting for your suggestions.

## 1.14 New

New in mapPlug

- Nothing new...

## 1.15 History

Little mapPlug's history:

1.41

- Solves a couple of minor issues in v1.4.
- Exporting a map requires less memory.
- Solved a "frontwall stomp" problem.
- Does not put "localhost-etc-etc" in front of "mailto:", "ftp:", etc...
- Selecting the menus 'Add point' and 'Remove point' did nothing.
- The special 'Rectangle to array' could leave the 'Do not redraw' bit set in the mapclass, resulting in that it did not redrawed :)
- Trying to clone a shape when there were no shapes (???) crashed mapPlug.

1.4

- Added an 'Add Point' function.
- Added a 'Remove Point' function.
- Implemented 3 "special" functions: Subdivide polygon (by doubling the number of points), rectangle array and convert to polygon.
- When unregistered, creating a new shape by cloning an exiting one lead to refresh problems (the copy was drawn, but it didn't exist).
- The initial dir for each shape was not shown when it was created. Solved.
- Changed the "Clone" function shortcut, since 'C' collides with BetterString's copy function, of course.
- Changed the layout of the toolbar.
- Removed the "top" and "bottom" toolbars. Only vertical are now possible.
- Implemented a method to inhibit the redraws of mapClass. Used in load, save and exporting function, reducing the redraws produced in these operations.
- Cannot export to the clipboard if unregistered (guess why :)
- Cannot export more than 2 shapes if unregistered (guess why :)
- This version was sold at the WOA'99

v1.3

- Improved redrawing speed on low color images on ECS/AGA systems by removing some unnecessary blits.

- Removed the 100 point limit for polygonal shapes. Actually, the method that mP used to handle polygons has been completely rewritten.
- Changed the way that mP's mapClass handles input.
- Added the Clone function.
- Increased to 4 the control points of the rectangular shape.
- Increased to 4 the control points of the circular shape.
- Improved the circle creation procedure. Now you can move the mouse in any direction to define the radius.
- Solved some small redrawing problems which caused some visual artifacts.
- More extensive usage of v39's mempools.
- Windows are correctly refreshed if the user moves a sync-filerequester and while loading a new image.
- mP redrew the shapes more times than strictly necessary. No more.
- Fixed a long standing (and not reported) bug in rectangle's "limit handling" :)
- Changed the behaviour of the class: You can select a shape and immediately start dragging it w/o releasing the mouse button.
- Sped up polygons redraw.
- You can now select a shape and immediately move it.

#### v1.21

- Closes <map> tag correctly.

#### v1.2

- Reduced required mem for zooming by 90% (or more, depending on the image)
- Sped up shapes redrawing for rects and circs (not polys yet, sorry).
- Load/Save maps in IFF files

#### v1.15

- The main class of the program made a shared library (ok... it can be done).
- Never released.

#### v1.1

- Added Zoom option to the mapclass.
- Added prefs window with user selectable colors for the shapes.

#### v1.0

- First Aminet release
- Support for all the shapes

#### v0.x

- Internal beta releases
-

## 1.16 Disclaimer

\$VER: "AFD-COPYRIGHT"(TM) English Version 1.2 (27.07.96)

\_\_\_ \*=====\*

/// \* Standard Amiga FD-Software \* ///

/// \* Copyright Note (TM) \* ///

/// \*=====\*

\_\_\_ /// \* \* \_\_\_ ///

\\V/// \* Version 1.2 - English \* \\V///

\\V/// \* 27-Jul-1996 \* \\V///

\\/// \* \* \\///

"" "" \*=====\*

### 0. INTRODUCTION

=====

If you are a user I hope you like this Standard Copyright Note for Freely Distributable Amiga® Software you only have to read once instead of individual notes. There are translated versions of this document available on Aminet® named:

/pub/aminet/docs/misc/AFD-Files1-xx.lha (xx = number of languages).

If you are a software developer and want to include this document in your own FD package then please read it carefully first, especially chapter 5 "DEVELOPERS INFORMATION". Thank you for your support.

### 1. COPYRIGHT

=====

- a. This software is copyrighted by its developer(s). That means that you are NOT ALLOWED to modify the program(s) and documentation in any way. Especially you MUST NOT REMOVE the documentation or this text file.
- b. You are NOT allowed to use this software or any part of it for any other purpose than that one mentioned in its documentation, this also includes any fonts, images or samples. If the developer(s) did NOT include the source code of the program(s) in this package you are NOT allowed to de-compile any part of it.

### 2. DISTRIBUTION

=====

This package is freely distributable. That means you are allowed to re-distribute this package as long as you follow these points:

- a. Any re-distribution has to include all files in this archive, including this "AFD-COPYRIGHT" file, without any modifications. You are NOT allowed to add any files to the archive.

b. This package may be freely distributed via BBSs, InterNet/UseNet, software libraries such as Fred Fish's and Aminet® CD-ROM, and other similar electronic channels.

c. Disk magazines and services that charge extra for file transfers may NOT distribute it without written permission by the developer(s)!

### 3. DISCLAIMER

=====

By using this product, you accept the FULL responsibility for any damage or loss that might occur through its use or the inability to use it. The developer(s) of the software and the author and the translators of this "Copyright Note" can NOT be held responsible.

IMPORTANT: The author and the translators of this "Copyright Note" do NOT give any guarantee for the quality and usefulness of ANY products that are subject to this note!!!

Some names used in this text are trademarks or registered trademarks.

The use of these names does not imply that they are free.

### 4. RETURN SERVICE

=====

"Freely distributable" only says that you do not have to pay for copying or redistributing the software. You are allowed to test this product for 30 days. If you like it and decide to use the product regularly, most of the developers want something in return for their efforts.

The documentation of this software contains one classification out of these:

- a. Freeware - You are allowed to use this software FREE.
- b. Mailware - You have to send the developer(s) a MESSAGE (by email or snailmail).
- c. Cardware - You have to send the developer(s) a POSTCARD.
- d. Donationware - You have to make a DONATION to an organization as stated in the documentation to this product.
- g. Giftware - You are requested to send the developer(s) a GIFT, for example:
  - some candy or
  - a package of disks or
  - a copy of a self-made program or
  - some money or
  - an Amiga 5000 PowerTower ;-)
- s. Shareware - You have to send the developer(s) the amount of MONEY demanded in the documentation.

### 5. DEVELOPERS INFORMATION

---

=====

The concept of a "Standard-Licence" like this one has several advantages:

- You as a developer don't have to think about copyright stuff yourself.
- The users of your product have to read it only if they haven't done so before.
- This document is available in several languages via Aminet@.

So you can be sure that the users have read it at least once and did understand it.

If you want to include "AFD-COPYRIGHT"(TM) in your own Amiga® FD-Software package you have to follow these rules:

- a. Read this whole document carefully and make sure that you agree with the whole text WITHOUT ANY EXCEPTIONS.

(If you do not agree with it, do not use it! You are allowed to copy parts of it, but IF you do so, you MUST NOT use the names "AFD-COPYRIGHT"(TM) and "Standard Amiga FD-Software Copyright Note"(TM).)

- b. Make sure that you use the following Default-Tools in the .info-files you are going to release:

- for ASCII texts: More
- for AmigaGuide® documents: AmigaGuide
- for AmigaOS scripts: IconX
- for ARexx scripts: RX
- for configuration files: Ed
- for Installer scripts: Installer
- for IFF-ILBM pictures: Display

(The idea of this rule was inspired by Osma "Tau" Ahvenlampi, Finland.

It was included to set a standard. If someone does not use these tools (s)he has to make not more than one link per file type!)

- c. Include a paragraph like the following in the manual to your product:

-----8<-----

This software is subject to the "Standard Amiga FD-Software Copyright Note".

It is <type\_of\_software> as defined in paragraph 4<x>. [...]

For more information please read "AFD-COPYRIGHT" (Version 1 or higher).

-----8<-----

Example:

This software is subject to the "Standard Amiga FD-Software Copyright Note".

It is SHAREWARE as defined in paragraph 4s.

If you like it and use it regularly please send \$15 to ...

For more information please read "AFD-COPYRIGHT" (Version 1 or higher).

- d. You have to include this file called "AFD-COPYRIGHT"(TM) (English).

IF you include an AmigaGuide® hypertext documentation you can link it to this file or include the whole text in the AmigaGuide® file just as you like. Please do not add more than two translations of this file to your package. So your package doesn't get too big.

## 6. AUTHORS

=====

"AFD-COPYRIGHT"(TM) was written by Dietmar Knoll. It is itself subject to the above copyright note (© 10.12.1994, 27.02.1996).

If you have any comments, hints or questions or some suggestions for the next version, please write to the following addresses:

Dietmar Knoll InterNet-Email: To: dknoll@gwdg.de

Iltisweg 6 Subject: AFD

D-37081 Göttingen My WWW-Homepage:

GERMANY/EUROPE ftp://ftp.gwdg.de/pub/physik1/WWW/GAUHPIL/Welcome.html

The translators of this document and their addresses are stated in the corresponding files. The translations are © by the translators.

## 1.17 MUI

This application uses

MUI - MagicUserInterface

(c) Copyright 1992-97 by Stefan Stuntz

MUI is a system to generate and maintain graphical user interfaces. With the aid of a preferences program, the user of an application has the ability to customize the outfit according to his personal taste.

MUI is distributed as shareware. To obtain a complete package containing lots of examples and more information about registration please look for a file called "muiXXusr.lha" (XX means the latest version number) on your local bulletin boards or on public domain disks.

If you want to register directly, feel free to send

DM 30.- or US\$ 20.-

to

Stefan Stuntz

Eduard-Spranger-Straße 7

80935 München

GERMANY

Support and online registration is available at

<http://www.sasg.com/>

-----  
Without MUI, mapPlug would probably not exist...