

tablePlug English Guide

Esteve Boix

COLLABORATORS

	<i>TITLE :</i> tablePlug 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

1	tablePlug English Guide	1
1.1	Welcome	1
1.2	Introduction to tablePlug	1
1.3	Features	2
1.4	Install	2
1.5	Future	2
1.6	What do I need ?	3
1.7	Using tablePlug	3
1.8	Using tablePlug: General	3
1.9	Using tablePlug: HTML indexes	4
1.10	Using tablePlug: Thumbnails	7
1.11	Using tablePlug: Generating	8
1.12	Using tablePlug: Some tricks	8
1.13	Registering tablePlug	9
1.14	Getting tablePlug	9
1.15	Greetings	9
1.16	History	9
1.17	New	10
1.18	Disclaimer	10
1.19	MUI	14

Chapter 1

tablePlug English Guide

1.1 Welcome

tablePlug v1.2

An HTML catalog creator

What's **new** in this version

©1999 Esteve Boix

[Introduction](#)

[Requirements](#)

[Features](#)

[Installing tablePlug](#)

[Using tablePlug](#)

[Disclaimer](#)

[Registering << Freeware](#)

[Getting tablePlug](#)

[About MUI](#)

[Greetings](#)

[History](#)

[The future](#)

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

1.2 Introduction to tablePlug

Introduction to tablePlug

tablePlug is a powerful HTML catalog generator.

It's main purpose is to create catalog files out of a directory full of images (and other directories). It will scan the directory, and then it will create the necessary HTML files and thumbnails so that you can store them, and navigate through the catalog using your favourite WWW browser.

tablePlug generates the HTML files, and uses ARexx to control image processing programs to create the thumbnails.

The program has tons of options. The user has a lot of control on how the pages and the thumbnails are created. It has some many options that may seem confusing at first... Once you've created a couple of catalogs you will see it's easier and more intuitive than it seems at first sight.

It comes with Arexx programs (or modules) for the following image processing/paint packages:

ArtEffect (v3 required for some modules)

TVPaint v3.59 (the freeware version you can obtain from NewTek's site)

GfxCon

ImageStudio (thanks to Dave Naylor).

Creating new modules is very easy. Expect support for more programs soon.

If you create a new TPX module, please let me know it, since it may be included in newer releases of the program.

1.3 Features

tablePlug features

Here goes an small list of features:

- Generates HTML v3.2+ compliant HTML files.
- The images directories can mix images and more dirs at the same level.
- The thumbnails can be placed in any directory.
- Very configurable.
- The thumbnails are generated by external programs. tP comes with some modules. More to come.
- Ability to use one rexx program for each type of file (one for jpgs, one for gifs, one for MPEG files...).
- Generates a "Navigation" file, with links to all the directories of the catalog.

1.4 Install

Installing tablePlug

tablePlug does not require installation. Just drag the icon where you want.

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. Copy the files in your MUI:Libs/ directory.

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

1.5 Future

The future of tablePlug

Ok... The todo list of this program is (as usual) very large. Here go some ideas. Waiting for suggestions:

- Ability to load and save all the settings from/to a file. Currently this is not implemented, and has now become the highest priority for tP.
 - More control over generated files
 - CSS (any Amiga browser, please ?)
 - Support for more image processing programs.
-

- Add a "rename" feature. I've found some problems burning ISO9660 (yes, there's RR, but anyway...) Level 2 CD-R's, since the burning soft renames files if they don't match the standard ISO9660 naming conventions. The problem comes because the HTML files point to the old filename, and the catalog is not useable.

If I make tP recognize bad filenames while generating the catalog, everything will be fine.

- More control (now there's no control at all) over the generated "Navigation page".
- Some more JavaScript support.
- You name it.

1.6 What do I need ?

What do I need ?

In order to run tablePlug, 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>).

- An image processing program. You can use GfxCon for simple thumbnail creation, but using more powerfull paint packages such as TVPaint or ArtEffect gives better results.

- A WWW browser (HTML v3.2+ capable) to view the catalog files.

tablePlug runs directly from the dir. No need to install it.

1.7 Using tablePlug

Using tablePlug

tablePlug's GUI is divided in 3 sections.

This program may seem a little complicated, but I think it's fairly easy to use. I hope you'll get it fast.

General

HTML Indexes

Thumbnails

Generating the catalog

Some tricks

1.8 Using tablePlug: General

General

A few (yet very important) options are available here:

Images dir:

Here you have to specify the source directory of the images you want to catalog.

IMPORTANT: Avoid spaces in filenames, ALWAYS.

Generation dir:

Here you have to specify where will tP generate the thumbnails and HTML files.

Once tP has done it's job, this directory will contain a duplicate of the original directory structure, the HTML files and the thumbnails.

While it's possible the "Images dir" and "Generation dir" could be the same, better keep them separated.

Use lowercase in HTML tags code:

If active, this option will force tP to use lowercase (instead of UPPERCASE) when generates the HTML code.

Use existing thumbnails:

Will make tP not to generate an already existing thumbnail. This is useful if the process has been stopped before, and you want to continue it.

Use existing HTML files:

Will make tP not to generate an already existing HTML file. This is useful if the process has been stopped before, and you want to continue it. Note that the behaviour of the program has changed for v1.1, and it doesn't write incomplete pages anymore.

Handle this option with care, since some strange things could happen. Imagine an already generated HTML file. If you add a new image to the directory, tP could generate the thumbnail, but would not update the HTML file, which is not what you'd like.

This option is mainly oriented to finish an unfinished catalog.

Do not generate HTML files, just thumbnails:

Will make tP not to create the HTML files, just the thumbnails for the images.

This could be useful if you want to generate the thumbs first, and then try some HTML designs over them.

Not creating the HTML files will speed up somehow (nothing spectacular, but tP will not have to check the size of each thumbnail, and this can save some seconds for every thumbnail) the thumbnail creation.

1.9 Using tablePlug: HTML indexes

HTML indexes

This page gives the user a total control on how will tP generate the HTML files... Colors, sizes, positions, etc...

It's somehow complex. Give a read before using it.

The page is divided in two sections (Items and formats), each one divided in 3 subsections.

Let's start by the 'Items' tab.

Pages

tP generates HTML files, which contains links to more pages (more directories) and thumbnails.

This subsection control how these general elements appear on the page.

You will see listviews: Available fields and Selected fields. You can drag and drop elements each list to the other (except for the 'white' elements, always selected).

The elements on the 'Selected fields' listview are the ones that tP generates.

So imagine you want an HTML file with a title text, a list of other directories, the thumbnails and a link to the previous (parent) dir.

Ok, drag 'Title text' and 'Link up' to the selected fields. You can drag-sort the elements in the 'Selected fields'. This is important, since the order of the elements in the listview is the order of the elements in the HTML file that tP will generate.

Some items have additional options. This options become available when you select the item in the 'Selected fields' listview.

Here goes a listing of the options in this section:

Current dir:

tP will insert a string with the current dir in the page.

tablePlug replaces the '%s' placeholder by the actual directory name when it generates the HTML pages. You can write whatever you want around the '%s' placeholder.

Link up:

tP will insert a link to the parent directory (i.e. parent HTML file) if available. The link may be a text or an image file. You have to specify it using the cycle gadget.

Link to root:

tP will insert a link to the root directory (i.e. root HTML file). The link may be a text or an image file. You have to specify it using the cycle gadget.

Link to navigation:

tP will insert a link to the navigation page. The link may be a text or an image file. You have to specify it using the cycle gadget.

The navigation page is an special page that contains a list of all the HTML files generated. This page will only be generated if you add a link to it with this option.

Currently there is not much control over this page. Future version will remedy this.

Text (1,2,3,4,5):

You can insert any text here. The text may contain plain text and HTML code.

Image (1,2,3,4,5):

You can specify an imagefile here.

Horiz line (1,2,3,4,5):

This item inserts a <hr> line. You can set some options for each one.

Empty Line (1,2,3,4,5):

Inserts a
.

As of v1.1, the 'Thumbnails' element has an option page !

You can add JavaScript code to the generated HTML page, so that when the user moves the mouse over a thumbnail, the browser will display some info of the image in the status bar.

This option work (as usual) with placeholders:

%f - The image filename (just the filename, not the full path)

%s - The filesize in bytes

%c - The filecomment

%x - The image width, in pixels

%y - The image height, in pixels

So, enabling JavaScript onMouseOver events and setting the string to "%f - %s bytes - %xx%y pixels" would force the browser to display the filename, the filesize and the image size whenever the user moves the mouse over any thumbnail.

Directories

This section controls what should appear for each directory of the current level:

Dir name:

This will always appear

Dir comment:

The comment attached to the directory

Dir entries:

The number of entries in this directory

Thumbnails

This section controls what should appear for each file of the current level.

You should not leave the 'Selected items' list empty. It's nonsense, and tP will complain.

Thumbnail:

Ehem...

Filename:

Ehem again

File size:

The size of the file, in bytes, followed by the word "bytes".

File comment:

The comment of the current file.

Image size:

The size of the image in the format "Width x Height".

The second tab is 'Formats'.

This one contains options to control the look of the HTML pages.

It's again divided in 2 more sections:

HTML files

You have here options to control the colors and backgrounds of the HTML files.

Page title:

You can set here the title of the generated files. tablePlug will replace the placeholder '%s' by the current directory.

BG Image:

Sets the background image file.

BG Color:

The background color of the file.

Text:

The color of the text in the page.

Link:

The color of the links in the page.

Link:

The color of the visited links in the page.

ALink:

The color of the links in the page while you press them with the mouse.

Tables

Here you have control over the look of the tables that enclose the thumbnails and the links to other directories.

Colors:

Background:

Background color for the whole table.

TR Background:

Background color for each row of cells.

TD Background:

Background color for each cell.

Alignment:

Table alignment:

Horiztional position of the table in the page (left, centered, right or browser default).

Horiz. cell align:

Horiztional alignment of the contents of each cell inside the cell.

Vert. cell align:

Vertical alignment of the contents of each cell inside the cell.

Sizes:

Cells per row for directories:

Number of cells per row for the tables that contain the directories.

Cells per row for thumbnails:

Number of cells per row for the tables that contain the thumbnails.

Width:

Width of the tables, in percent.

Border:

Size of the border of the tables, in pixels.

Cellpadding:

Space between the margin of the cell and its contents.

Cellspacing:

Space between cells.

1.10 Using tablePlug: Thumbnails

Thumbnails

This section controls how tablePlug will generate the thumbnails for your images, anims, etc...

The idea is that you assign a '.TPX' module to a given file extension. When tablePlug finds a file of this extension, it will run the '.TPX' program for it. This way, you can use a program to create thumbnails for JPEG files, another for MPEG movies, and another for GIF files. It may even be possible (mm...) to create a TPX to create thumbnails for 3D objects (!!!).

Let's see how it works:

The main listview shows the options. Each entry is a set of options that you've set in the controls:

Extensions:

You have to write here which extensions will be affected by the current TPX. You will find the most common extension in the pop-up list, but you can of course write your own.

You have to use the normal AmigaOS patterns.

Thumbnail generation:

You may want tablePlug to assign a fixed thumbnail to this filetype, or let it generate a thumbnail. You control this behaviour with this cycle gadget.

Icon image:

If you've chosen to use a fixed thumbnail, you have to specify it here.

This is not the only function of this image. If you've chosen to use a TPX to generate a thumbnail, and if for some reason it fails to create a given thumbnail, this image will be used as a replacement (and of course you'll be informed).

Generator:

You can choose here between the available TPX programs.

Pressing 'Info' will show you the source code of the TPX program. The header of the program should contain some instructions on how to use it.

Arguments:

You have to specify here the arguments for the TPX program. No checking is done. Read the source code carefully.

Default icon image:

This icon image will be used as the thumbnail of any file that does not match any entry.

Once you've created an entry, you have to press 'Add to list', so tP will store it in the list.

Delete current:

Will delete the highlighted entry.

Update current:

Will overwrite the selected entry with the current settings.

Clear list:

...

Clear fields:

Will clear all the fields, so that you can write a new entry.

1.11 Using tablePlug: Generating

By pressing the 'Generate' button, you make tablePlug start the process...

Here goes an small description of what goes on:

- 1.tP checks that the user settings are valid.
- 2.Checks if all the specified thumbnails and images exist.
- 3.Reads the images directory and stores the tree in the memory.
- 4.Starts generating thumbnails and HTML files.

You may cancel the process at any moment by pressing 'Cancel' (while the image processing program is performing an action, tP will not react to your mouse clicks... Be patient).

tP will ask if you're sure. If yes, tP will close all the opened files and free the allocated resources.

1.12 Using tablePlug: Some tricks

Some tricks:

Generating a catalog for a large number of images can be a timeconsuming task...

I've found several tricks that, while nothing spectacular, may speed up things a little.

- Generate the thumbnails first, if possible using a program that loads JPGs scaled.

- tp_dtc supports the animation datatype, so you can use it to create thumbnails for your animations... If you have the MPEG video datatype, you should upgrade to the Animation.datatype v41.3 or better, since older versions don't deliver 24bit images.

Also, you should set the NOLOADALL option to the prefs file, so tp_dtc will not have to read all the animation to save the first frame.

- tablePlug creates a 'tablePlug:' assign to it's directory. Use it in your '.TPX' modules.

1.13 Registering tablePlug

Registering tablePlug

tP 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 :)

1.14 Getting tablePlug

Getting tablePlug

I will always upload the latest version of tablePlug to Aminet.

You can also get it from my homepage (along with some other programs I've written). Registered users will be notified by email about new updates.

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

If you have any questions/suggestions/bug reports/etc..., do not hesitate to drop me a mail:

eboix@teleline.es

1.15 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 patience...
- Dave Naylor, for being the first to register tP :), and for the ImageStudio TPX module.

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).

1.16 History

Little tablePlug's history:

v1.2

- <head> is correctly set
- The status bar text of the browser is cleared when the mouse leaves the thumbnail
- A couple of other minor issues solved.
- CSS "support" :)

v1.1

- Now explicetely calls 'sys:rexxc/rx' to run the TPXs. Future versions will use smarter ways to startup modules.
 - Added onMouseOver JavaScript event support so that the browser will display any info you want in the status line when the user moves the mouse over a thumbnail.
-

- The generators list is now sorted alphabetically.
- Added some more info on each TPX module.
- Removed the
 after <div>s.
- Added options for <hr>
- Added 5 optional
.
- Added bubblehelp support.
- Ask for confirmation to quit. Added after I quitted accidentally the program twice.
- Moved a lot of things to internal custom classes, resulting in smaller executable and more source code organization.
- Added support for placeholders in most program's options.
- Increased the usage of mempools.

v1.0

- First Aminet public release

v1.0prerelease

- First release at WOA

v0.x

- Internal beta releases

1.17 New

What's new in tablePlug :

- <head> is correctly set
- The status bar text of the browser is cleared when the mouse leaves the thumbnail
- Possibility to link an external CSS file.
- A couple of other minor issues solved.

1.18 Disclaimer

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

__ __ *=====*

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

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

/// *=====*

__ __ /// * * __ __ ///

\\ \\ \\ * Version 1.2 - English * \\ \\ \\

\\ \\ \\ * 27-Jul-1996 * \\ \\ \\

\\ \\ \\ * * \\ \\ \\

"" "" *=====*

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

Iltilisweg 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.

1.19 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, tablePlug would probably not exist...
