TPaint:TPaint

COLLABORATORS			
	TITLE : TPaint:TPaint		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		July 19, 2024	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

TPaint:TPaint

1.1   TPaint:TPaint.guide

```
TPaint
******

   This is the manual for

    TTTTTTT  PPPPP
       T     P  P      i           t        V   V   000      999
       T     P  P                    ttttt    V   V  0   0    9   9
       T     P  P  aaaa  i   nnnn    t        V   V  0   0    9   9
       T     PPPPP     a  i   n   n   t        V V   0   0     9999
       T     P     aaaaa i   n   n   t        V V   0   0        9
       T     P     a   a i   n   n   t        V V   0   0 ..      9
       T     P     aaaaa ii  n   n   tt        V     000  .. 999

   Copyright (C) 1995 by Jens Bender
```

```
   Copyright and Disclaimer   Read this first
   Introduction               A short overview over the features of TPaint
   Getting started            Installing and starting the program
   Screen-Requester           The thing you see after starting TPaint
   File-Requester             Used everytime you're asked for a filename
   The Menus                  The description of the menus of TPaint
   Key Commands               Commands just reachable by keyboard
   Error Messages             What can happen you don't expect
   Bug Reports and Comments   Now its your turn
```

1.2   TPaint.guide/Copyright and Disclaimer

```
Copyright and Disclaimer
************************

   THIS MANUAL AND THE DESCRIBED PROGRAM TPAINT INCLUDING THE LIBRARIES
```

TEDDI.LIBRARY, DITHER.LIBRARY AND LOSAPR.LIBRARY ARE COPYRIGHTED (C)
1995 BY JENS BENDER.

   This is a fact.

   Permission is granted to copy the program and related files as far
as no rights of third parties are affected. TPaint is shareware. This
means that you are allowed to test it and copy it but if you want to
use it remember to register.

   No one but me is allowed to take money for distributing it. To state
it clearly: if you distribute or copy this program and manual you are
not allowed to take any money (except for your own expenses, of course).

   Copying this program is only allowed in connection with this manual
and all other files included in this distribution. You are not allowed
to make any changes to the program or the manual, especially the
copyright notice.

   TPaint and the libraries are written by me from scratch using
Matthew Dillon's wonderful DICE C-compiler (freeware-version). No code
written by someone else but me is included except code included from
c.lib written by Matt and from amigas.lib created also by Matt. The
manual is written using TEXINFO distributed by the Free Software
Foundation. The printed manual is typeset using TeX, written by Donald
Knuth and ported by Georg Hessmann.

   This distribution currently contains the following files:

     TPaint   ;the main program
     TPaint.info
     TPaint.readme   ;The short description
     TPaint.guide   ;the AmigaGuide-format manual
     TPaint.guide.info
     TPaint_default_icon.info   ;icon for saved pictures and brushes
     palette/#?   ;some examples of palettes
     filter/#?   ;some examples of filters
     libs/teddi.library   ;needed by TPaint (requester, menu ...)
     libs/dither.library   ;needed by TPaint (dithering)
     libs/losapr.library   ;needed by TPaint (load, save, print)
     locale/deutsch/tpaint.catalog   ;german strings for TPaint
     locale/deutsch/teddi.catalog   ;german strings for teddi.library
     locale/deutsch/dither.catalog   ;german strings for dither.library
     locale/deutsch/losapr.catalog   ;german strings for losapr.library
     install/deutsch.info   ;for german installation
     install/english.info   ;for english installation
     install/TPaint.install   ;the installation script
     install/Installer   ;Copyrighted (C) by AMIGA Technologies
                                     ;   (I think so)

   Amiga, AmigaDOS, Intuition, WorkBench, ... are trademarks of
someone, I think it's currently AMIGA Technologies, but who knows?

   NO WARRANTY NEITHER IMPLIED NOR EXPRESSED IS GIVEN THAT THIS PROGRAM
OR THIS MANUAL OR THE INFORMATIONS OF THE MANUAL ARE USEFUL FOR ANY
PURPOSE NOR THAT THEY ARE FREE OF ERRORS. USE OF THIS PROGRAM OR THE
MANUAL OR INFORMATIONS OF THE MANUAL FOR ANY PURPOSE IS ABSOLUTELY AT

YOUR OWN RISK.

1.3 TPaint.guide/Introduction

Introduction
****

   Caution: This manual is only an simple introduction to the functions
of TPaint, to learn how to use the functions and about the results of
them you have to try. About menus, screens ... see your Amiga-Manuals.

   TPaint is a 24bit painting program. All operations on pictures are
performed in full 24bit. This means a choice of 16777216 different
colors. This number of colors is thought of to be enough to present
every color the human eye can distinguish. The Amiga equipped with the
AA-chipset is able to display all of these colors. The problem is that
only 256 different colors (except HAM-modi) can be displayed on the
same screen. The solution is to keep the 24bit picture somewhere in
memory and to compute a picture having at maximum 256 colors to
approximate the 24bit picture. This process is called dithering. This
is the way TPaint works. The 24bit picture is maintained in memory and
all actions performed by you handle with this picture. After this the
display is recomputed to show the user what he has done. The trick is
to do this a way the user notice this procedure as little as possible,
especially speed is a problem while recomputing the display.

   To speed up things TPaint knows two different algorithms to compute
the display, one fast and one good. The fast one (ordered dithering) is
used while painting a picture. The better algorithm (Floyd-Steinberg
dithering) is only invoked by user's request since it is slower than
ordered dithering. In most cases the results are much better than
ordered dithering. (see 'Picture/Display')

   Now we have another problem. The picture can be build up out of
16777216 colors (if the picture has at least 16777216 pixels), but the
palette of the display has at maximum 256 colors (depending on your
choice 2, 4, 8, 16, 32, 64, 128 or 256 colors). The question is which
colors should be loaded in this palette to present the picture as well
as possible. TPaint provides a default palette which contains a wide
variety of colors. This palette should be ok to get a rough impression
of the actual picture and is ok while painting. To compute the finally
displayed picture TPaint can compute a new palette depending on the
colors actually used in the picture, or you can change the palette
manually if you think you can do a better (or faster) job. (see
'Picture/Display')

   After starting TPaint you are asked to select a screentype which
should be used as the working area of TPaint. (see Screen-Requester).
TPaint opens a standard Intuition screen which causes that all
screentypes supported by Intuition are usable for TPaint. The program
has been developed to be used with the AA-chipset but it works via the
Intuition interface with ECS or OCS. I haven't tested this but also
graphic cards supporting Intuition should be supported (but no modes
having more than 256 colors. Also the speed could be ridicolous,

depending on the cooperation of the card driver and TPaint.) Because
the colorselector (see 'Toolbox/Color') comes up on its own screen the
graphics card must be able to handle multiple screens.

As mentioned, TPaint works on a standard Intuition screen using the
Intuition window management (same like the WorkBench). This feature is
used to give you the possibility to work on several pictures
simultaneously. Every picture you create or load (See
'Picture/New Picture' and 'Picture/IO/Load') is displayed in its own
window and can be processed by you. You can change the size and
position of the windows to see different pictures the same time. This
allows also pictures much larger than the screen. TPaint sets the size
limit at 16384 X 16384 pixel for each picture. Using the sliders at the
right and at the bottom of the window of a picture you can select the
part of the picture you want to see on screen. They work like the
sliders of the WorkBench-windows. You can open as many pictures as you
want, you are only limited by available memory.

Furthermore TPaint is able to handle brushes. There may also be as
many brushes as you want, you are again only limited by available
memory. You can use brushes for all operations that normally would
effect a point to be set or an area to be filled. Also the brushes can
be manipulated in many ways. (See the whole 'Brush'-menu and many other
sections.)

For actual painting TPaint has a set of painting tools (see
'Toolbox/Tools') that can be used for drawing. The behaviour of the
tools can be altered in many ways. See 'Toolbox/Pencil',
'Toolbox/Filling' ... All tools and functions can work with a variable
intensity. This means that the tools may be altered that way that they
don't overwrite the old picture but the result of painting is composed
out of the old picture and the tool. The intensity may be altered
globally using the gadget [Intensity] located at 'Toolbox/Pencil' or
locally using the 'Picture/Alphachannel'.

For painting details TPaint offers you a zoom (see 'Toolbox/Zoom'
with variable magnification. All tools and functions may be used
directly in the zoom area.

Some functions of TPaint may take some time to compute. Most those
functions display their state of progression in the status window (of
course only if it is active, see 'Toolbox/Status'). Every function
displaying this progression indicator and some more functions can be
cancelled by just typing ESC.

TPaint should work with any OS version starting with 1.2, but I'm
not sure since I have no possibility to test it. It's written on a
machine equipped with OS 3.0 but where it uses features of this version
it falls back to elder OS functions if necessary.

If you have installed locale.library (standard since OS 2.1) TPaint
uses german texts if you want.

1.4 TPaint.guide/Getting started

```
Getting started
***************
```

I hope, you'll find this doc useful for your first experiences with TPaint. For I don't have the patience to write a big manual with lots of examples etc. instead of this doc, its your turn to make experiments with TPaint to learn about using it and about all the features.

Unfortunately I don't have managed to write a german doc. I hope, you can manage to read this. If you use german texts, the keys used as shortcuts for menus and gadgets are different from that described in this doc, you have to look at the menus and at the gadgets to see the right keys.

To ease reading this, I've introduced some conventions how things are said (written). If I'm saying (writing) something about menuitems, I'm speaking (writing) of a command from the Intuition-pull-down-menus at the top of the screen. You should know them from the WorkBench. Menuitems are always written like this 'menu/menuitem' or 'menu/menuitem/subitem' if there is a submenu. Things you have to type on your keyboard are written like this. Gadgets in a requester are labeled like [this].

Every requester in this program has a gadget labeled [OK] which is used to quit the requester and to start the action the requester is used for resp. to use the settings made in the requester. Most requesters have also a gadget labeled [Cancel] which is used to quit the requester and to undo all changes you performed in the requester. If a requester is a part of a function, the function will be terminated. Because these gadgets are the same in all requester I'll not mention them in the descriptions of the requesters.

Many gadgets in requesters can be called by the keyboard. In the name of a gadget having this possibility a letter is underlined. Typing this letter on the keyboard (upper and lower case are the same) is the same like clicking the gadget with the mouse. Typing RETURN is equivalent to the [OK]-gadget, ESC to the [Cancel]-gadget.

```
  Installation
  Starting TPaint
```

1.5 TPaint.guide/Installation

```
Installation
============
```

The installation is done using the Installer. Just click deutsch or english in the install-drawer to start the german or english installation procedure. The Installer tells you what to do. Refer to

your Amiga-Manual for more information about the Installer.

   If you want to install manualy, copy the files in the libs-drawer in
your Libs:-drawer (the libraries are necessary to start TPaint). Put
the files in the locale-drawer in your Locale:-drawer if you want to
have german texts. Put TPaint, TPaint.guide and the corresponding
.info-files somewhere you want. The file TPaint_default_icon.info
is used as the standard icon for saved pictures and brushes, put it
anywhere, but you have to make an assign TPaint: to the drawer where
you placed it. The pathes TPaint:palette and TPaint:filter are the
places where TPaint looks for saved palettes and filters (there are some
examples in the drawers palette and filter in this distribution). Add
this assign to your s:user-startup (preferred) or to your
s:startup-sequence.

1.6 TPaint.guide/Starting

Starting TPaint
===============

   You may start TPaint either from the Shell or from the WorkBench. In
both cases you may say TPaint to immediatly load a picture.

   To start from the Shell, just type TPaint and, if you wish a picture
to be loaded, the name of the picture: TPaint picturename.

   To start from the WorkBench, doubleclick the TPaint-icon. To say
TPaint to load a picture, hold down the SHIFT-key, click on the icon of
the desired picture and then doubleclick the TPaint-icon, still holding
the SHIFT-key. In case that the picture is created by TPaint it is ok
just to doubleclick the icon of the picture.

   The first thing you see after starting is the Screen-Requester.

1.7 TPaint.guide/Screen-Requester

The Screen-Requester
****

   After starting TPaint a requester appears that asks you for the
screen you wish TPaint to run on. You may select the type, the size and
the number of colors of the screen. If you have said TPaint to load a
picture after start the attributes of that picture are the default
settings of the requester.

   Selecting the Screentype
   Selecting the Screensize
   Selecting Overscan

```
   Selecting the Number of Colors
   Informations about Selection
```

   There are two ways to quit the requester: First you may choose the
[OK]-Gadget (or the RETURN-key). This causes TPaint to open a
screen with the selected attributes. Second you may exit TPaint
immediatly with the [Cancel]-Gadget (or the ESC-key).

1.8   TPaint.guide/Selecting the Screentype

```
Selecting the Screentype
========================
```

   At the left of the requester a list with the available screentypes is
shown. If the list has more than six entries you may scroll through the
list with the scrollbar at the right of the list. The current selected
screentype is marked at the left of the corresponding entry. At the
right edge of the requester some informations about the selected type
are displayed, see Informations about Selection.

1.9   TPaint.guide/Selecting the Screensize

```
Selecting the Screensize
========================
```

   Just below the list of screentypes there are two gadgets where you
may enter the size of the screen you desire. The upper gadget
corresponds to the width, the lower to the height of the screen. The
valid range for these numbers are shown at the right of the requester,
see Informations about Selection.

1.10   TPaint.guide/Selecting Overscan

```
Selecting Overscan
==================
```

   Intuition is able to manage different types of overscan. This means
the portion of the monitor actually used to display the screen. You may
select three different types of overscan by the gadget in the lower
middle of the requester. For the sizes of overscan see
Informations about Selection. About overscan in general refer to the
Amiga-manual.

1.11   TPaint.guide/Selecting the Number of Colors

```
Selecting the Number of Colors
==============================
```

   Just at the right of the list of screentypes is a list for selecting
the number of colors to be used for the screen. The more colors are
used, the more the picture matches the picture maintained in 24bit in
memory, but the slower display refreshing will get.

1.12   TPaint.guide/Informations about Selection

```
Informations about Selection
============================
```

   At the right of the requester there are some informations about the
selected screentype displayed:

   * In the first row the minimum screensize of the selected type is
     shown, first the width and second the height.

   * The second row contains the maximum screensize.

   * The maximum number of colors. Note that TPaint maintains pictures
     always with 24bit in memory, the number of colors is just the
     number of colors used to display the picture, see
     Selecting the Number of Colors.

   * This row shows the standard size of the screentype. This size is
     the default setting of the screensizegadgets, see
     Selecting the Screensize.

   * The next three rows shows the dimensions of the different kinds of
     overscan managed by Intuition. The overscan has no effect to the
     screensize, just to the size of the portion of the screen
     displayed. About selecting overscan see Selecting Overscan.

   * The last row shows some technical informations about the selected
     screentype. The first number is the number of screenrefreshes
     (vertical blanks) per second. If this number is less 50 or 60 Hz
     (depending on the monitor) the screen may flicker. The second
     number is the horizontal frequency.

1.13   TPaint.guide/File-Requester

```
The File-Requester
******************
```

   The filerequester is called everytime you are asked to enter a
filename. There are several gadgets to help you to do your job:

* The largest thing is the file list. You can select one of the
  entries of the list by clicking on it. The entries are also
  selectable if TPaint is searching for more entries. At the right
  of the list is a scrollbar which allows you to scroll through the
  list if there are more than 10 entries. If you click on an entry
  that is a directory TPaint jumps to that directory and starts
  reading the entries of that one. If you select a filename the
  filename is copied to the gadget [File].

* The gadget [Drawer] contains the name of the current directory.
  The current directory can be changed in different ways. First you
  can select a directoryname in the file list. Second you can choose
  a gadget in the last row of the requester. Third you can enter the
  name of a directory directly into this gadget.

* The gadget [File] contains the name of the currently select file.
  You can alter this name by selecting a filename in the file list
  or by entering the name of a file into the gadget.

* [Pattern] is used to enter a pattern as known from the shell.
  Only files those names matches this pattern are displayed in the
  file list. You can change the pattern by entering a new one into
  this gadget. After altering the pattern the current directory will
  be searched again for matching files. The pattern affects only
  names of files, directories are always displayed. Currently the
  following special symbols are recognized for pattern matching: #,
  ?, * , (,), |, %. The meaning of these symbols is explained in
  the AmigaDOS-Manual.

* [Drives] is used to display a list of the name of volumes and
  assigns. Volumenames are the name of the disks, hardisks etc. that
  are currently accesible by AmigaDOS. Assigns are shorthandnames
  defined by DOS or by you, see the AmigaDOS manual. You change to a
  volume or an assign by clicking on its name.

* [Root] takes you to the root directory of the current volume.

* [<<<<<<] takes you to the parent directory of the current one.

If you need some more informations about directories, volumes,
patterns... refer to the AmigaDOS-Manual.

1.14 TPaint.guide/The Menus

The Menus
****

   TPaint has four Intuition-menus. They are used the same way like the
WorkBench-menus: hold down the right mousebutton, move the mousecursor
above the desired menuitem and release the mousebutton. Menuitems that
have subitems are marked with '»' at the left. Menus that causes a
requester to appear are marked with ... after the menu-name.

Many menuitems are reachable by key-shortcuts. The key you have to press to call a menuitem is at the end of the menuitem, separated by |. Case is significant, that means upper case keys call a different menuitem than lower case keys. A ^ preceding the key means that you have to hold CTRL down and then press the key.

All key-shortcuts of the 'System'-menu and of the 'Toolbox'-menu are CTRL-sequences, shortcuts from the 'Picture'-menu are lower case keys and shortcuts from the 'Brush'-menu are upper case keys.

```
System                  Look at it
Picture                 Menuitems affecting the current picture
Brush                   Menuitems affecting the current brush
Toolbox                 Tool-selection, Color-selection ...
```

1.15   TPaint.guide/System

The 'System'-menu
=================

This menu contains items that have no direct effect to a picture or a brush.

```
Shell...        |^c
IO-Control...
Delete file...
Rename file...
Info            |^i
About           |^a
Screen          |^s
Q u i t         |^q
```

1.16   TPaint.guide/menu0001

'System/Shell' - Interface to AmigaDOS

Opens a Shell on the WorkBench which allows you to enter shellcommands. Exit the Shell with endcli.

1.17   TPaint.guide/menu0010

```
'System/IO-Control' - Icons, Data Compressing
---------------------------------------------
```

   Opens a requester which allows you to select some settings for saving
pictures and brushes.

   *   TPaint uses the IFF-ILBM standard format for saving pictures
           and brushes. This standard is used by nearly all programs on the
           Amiga dealing with graphics. This standards allows to pack picture
           data in order to save diskspace. This is used by most programs,
           but some (very) old programs may not understand packed data. By
           switching this option off you are able to cause TPaint to write
           unpacked data. Most probably you don't ever need to use this
           option.

   *   If this option is switched off, TPaint doesn't write a
        WorkBench-icon corresponding to the saved picture or brush. If you
        switch this option from off to on, a File-Requester will appear to
        allow you to select a standard icon. The name must end with .info.
        The preset for the name of the standard icon is
        "TPaint:TPaint_default_icon.info".

1.18   TPaint.guide/menu0002

```
'System/Delete File'
-------------------
```

   A File-Requester appears, the selected file will be deleted.

1.19   TPaint.guide/menu0003

```
'System/Rename File'
-------------------
```

   A File-Requester will appear twice. The first time you have to
select a file you want to rename. The second time you select the name
the file should be renamed to.

1.20   TPaint.guide/menu0004

```
'System/Info' - Infos about Screen and Memory
---------------------------------------------
```

   The appearing requester shows the amount of memory in system not
used and the size of the largest unused block of memory. The
informations are divided into chipmem and fastmem. For information

about these different types of memory refer the Amiga-Manual.

   Also shown is the current time, the size of the screen and the
number of colors used to display the pictures.

1.21   TPaint.guide/menu0005

'System/About' - Infos about TPaint

   The most important requester of the whole program. It shows the
versionnumber of your version of TPaint and my copyright-notice.

1.22   TPaint.guide/menu0006

'System/Screen' - Change the Screen

   This menuitem closes the TPaint-screen and allows you to select a
new screen, see Selecting the Screentype. All pictures and brushes will
survive this change. If you have changed the number of colors of the
screen it may be necessary to recompute the display, see
Recomputing the display and 'Brush/Controlcenter'.

   Warning: You should save all pictures and brushes before calling this
menuitem. If TPaint isn't able to open the new screen (in case of low
memory) it will quit without giving you a chance to save your work.

1.23   TPaint.guide/menu0007

'System/Q u i t' - Quitting TPaint

   What do you think this menuitem will do? Right, it causes TPaint to
leave you. If there are any unsaved pictures, TPaint asks you if you
really want to destroy this picture. If you say [OK], the picture data
will be lost, if you say [Cancel], you will get the chance to save some
data before quitting. TPaint doesn't care about unsaved brushes.

1.24   TPaint.guide/Picture

The 'Picture'-menu
==================

   This menu contains all functions that deal with pictures.

  New Picture... |n
  Controlcenter... |c
  » IO
  » Manipulate
  » Alphachannel
  » Processing
  » Display
  Fill Colorrange
  Make Transition...
  Undo |u

1.25 TPaint.guide/menu0101

'Picture/New Picture' – An Empty Canvas

   Creates a new picture. The dimensions are asked from you by a
requester. The default values of the requester are the dimensions of
the current picture, or, if none exists, the dimensions of the screen.

1.26 TPaint.guide/menu0102

'Picture/Controlcenter' – Maintaining the Pictures

   Opens a requester which allows you to make some modifications on the
list of pictures. The controlcenter consists of these gadgets:

  * At the top of the requester you find the list of pictures currently
    existing in memory. You can choose a picture by clicking on it.
    The current picture is marked at the left of the name. If you
    leave the requester the marked picture will get the current
    picture.

  * At the right of the list the width and the height of the selected
    picture are displayed.

  * Every picture may have an annotation and a copyright-notice. If
    they exists they are saved along with the picturedata if you save
    the picture. If a loaded picture has an annotation or a
    copyright-notice they are displayed here if you call the
    controlcenter.

* With [Duplicate] you can make a copy of the current picture. The
  new picture is exactly the same as the old one.

* If you call [Delete] the current picture is entirely deleted from
  the picturelist and form memory. There is no was to recover. This
  is the same as clicking the close-gadget at the upper left edge of
  the window of the picture.

* The gadget [Recompute Display] is the same as the menuitem
  'Picture/Display/Recompute'.

* If the option [Display] is active you can see the picture on
  screen. If the option is inactive the picture doesn't appear on
  screen. This may help you to prevent your screen from being
  overcrowded. If you create a new picture this option is active.

* With the option [Undobuffer] you can enable or disable the
  undo-function of the program. For information about the
  undo-function see 'Picture/Undo'.

1.27   TPaint.guide/menu0110

'Picture/IO' - Loading/Saving/Printing a Picture

   The submenu of this menuitem contains commands for loading, saving
and printing the current picture.


```
  Load...           |l
  24bit Save        |s
  24bit Save As... |v
  Save As...
  Print...          |p
```


1.28   TPaint.guide/menu0111

'Picture/IO/Load'
....

   Loads the picture choosen by you via a File-Requester. In the current
version TPaint is able to load pictures that are saved in the IFF-ILBM
format. Both pictures with colormaps and 24bit-pictures are supported.

   If there are screenattributes saved along with the picturefile and
these attributes don't match the attributes of the current screen,
TPaint asks you if you want to change the screen. If you say [OK], the
Screen-Requester appears. The attributes of the picture are used as

default-settings for the requester, if TPaint is able to find a
screentype that matches the attributes of the picture.

1.29 TPaint.guide/menu0112

'Picture/IO/24bit Save'
....

   Saves the current picture as a 24bit picture. This means that the
file doesn't contain the picture you see on your monitor but the
picture that TPaint maintains in memory. The picture on screen is just a
approximation of this 24bit picture that is computed using the colors
available on screen. The filename that is used is the name last used for
this picture. If the picture has been saved before the name used there
is used again. If the picture has not been saved before the name which
has been used to load the picture is used. These two possibilities
imply that the old picture will be overwritten!.

   If the picture has neither been saved nor load before, TPaint asks
you for a name. In this case this is the same as
'Picture/IO/24bit Save As'.

1.30 TPaint.guide/menu0113

'Picture/IO/24bit Save As'
....

   The same as 'Picture/IO/24bit Save' except that you are asked for a
filename to be used for saving.

1.31 TPaint.guide/menu0114

'Picture/IO/Save As'
....

   Asks you for a filename and saves the current picture as it is shown
on screen. This means that not the 24bit picture in memory is saved but
just the approximation on screen. If you load this picture again the
24bit memory picture is the same as this screen representation and does
not match exactly the 24bit picture of the picture you originally saved.

1.32 TPaint.guide/menu0116

'Picture/IO/Print'
....

   By use of this menuitem you can print the current picture. First a
requester appears asking you a few things:

   * The topmost gadget changes between [Pixel] and [1/1000 Inch] if
     you click on it. This means the unit of measurement for the next
     two gadgets.

   * These two gadgets are used to enter the size of the printing
     according to the unit of measurement choosen by the first gadget.
     The size of a [Pixel] depends on your printer and the choosen
     density.

   * The last gadget is used to choose the density that is used for
     printing. The higher the number the better the result.

   You should notice that the standard WorkBench printer drivers are
used for this command. The results are in the best case usable as
drafts of your pictures. For better quality you should use a
specialized printing program which for example may be downloaded by you
from AMINET. The advantage of using the WorkBench drivers is that
nearly all printers are supported and I don't need to write a big bunch
of drivers.

1.33 TPaint.guide/menu0120

'Picture/Manipulate' – Manipulating a picture

   Here you find some basic functions to manipulate the current
picture. The 'Picture/Alphachannel' has no effect on these functions.
The function of this menu perform an automatic
'Picture/Display/Recompute'.

```
  Clear    |e
  Invert   |i
  Mirror X |[
  Mirror Y |]
```

1.34 TPaint.guide/menu0121

'Picture/Manipulate/Clear'
....

   Clears the current picture. This means that all points of the current

picture are set to black. The 'Picture/Alphachannel' is not changed, so
you can build up an alphachannel, clear the picture you have used to
build it and than draw the final picture using the alphachannel. The
clearing itself is not effected by the alphachannel.

1.35 TPaint.guide/menu0122

'Picture/Manipulate/Invert'
....

   Inverts all points of the current picture according to the RGB color
model. Black becomes white, red becomes cyan, green becomes magenta,
blue becomes yellow and vice versa.

1.36 TPaint.guide/menu0123

'Picture/Manipulate/Mirror X'
....

   Mirrors the current picture in the x-direction, that means at the
y-axis, left becomes right.

1.37 TPaint.guide/menu0124

'Picture/Manipulate/Mirror Y'
....

   Mirrors the current picture in the y-direction, that means at the
x-axis, top becomes bottom.

1.38 TPaint.guide/menu0130

'Picture/Alphachannel' - Masking out Areas from Painting

   Via the alphachannel you have the ability to protect some regions of
the picture against changing or to make the tools work with different
intensity at different regions of the picture. Remember that all
operations refering to colors of the picture refer to the colors of the
24bit memory picture and not to the display of the picture.

```
 Alpha transparent
 Invert Alpha
 Picture -> Alpha
 Alpha -> Stencil
 Add Colorrange
 Sub Colorrange
```

1.39   TPaint.guide/menu0131

'Picture/Alphachannel/Alpha transparent' - Switch off
....

   Makes the alphachannel transparent. That means that the alphachannel
is switched off. You can write all pixels with full intensity.

1.40   TPaint.guide/menu0132

'Picture/Alphachannel/Invert Alpha'
....

   Inverts the behavior of the alphachannel. Pixels that were protected
against overwriting are now unprotected, pixels that were unprotected
are now protected and pixels that were two third protected are now one
third protected ....

1.41   TPaint.guide/menu0133

'Picture/Alphachannel/Picture -> Alpha'
....

   Converts the current picture into the alphachannel for this picture.
Pixels that are black becomes protected against overwriting, white
pixels get unprotected. The brighter pixels are, the more they get
unprotected and the darker they are, the more they get protected.

1.42   TPaint.guide/menu0134

'Picture/Alphachannel/Alpha -> Stencil'
....

   Turnes the alphachannel into a stencil. Whereever the alphachannel
wasn't fully protected it becomes unprotected.

1.43   TPaint.guide/menu0135

`Picture/Alphachannel/Add Colorrange`
....

   After selecting this command you have first to select a colorange.
To do this you select a color by clicking on a point of the picture.
Then you have to define the fuzziness of the colorrange. How to do this
see Filling Areas. After selecting the colorrange TPaint start to
process the alphachannel. Points of the alphachannel corresponding to
points of the picture being in the selected colorrange are set to the
[Intensity] selected in the at `Toolbox/Pencil`. If you have
selected an intensity of 100% the points in the alphachannel are set to
full transparency, if you have selected an intensity of 0% the points
of the picture falling into the colorrange are protected against
overwriting.

1.44   TPaint.guide/menu0136

`Picture/Alphachannel/Sub Colorrange`
....

   This is likely the same as `Picture/Alphachannel/Add Colorrange`
except that the way that the [Intensity] (see `Toolbox/Pencil`) is
interpreted is reversed.

1.45   TPaint.guide/menu0150

`Picture/Processing` – Image Postprocessing

   The functions found in this menu are used for postprocessing the
current picture. They affect the 24bit image directly, not only the
display.

  Filter...
  Color Correction...

1.46   TPaint.guide/menu0151

`Picture/Processing/Filter`
....

   This command allows you to filter your picture. A filter is an array

of signed numbers. The new value of a pixel is computed as a weigthed
average of the neighborhood of the pixel, the numbers in the array are
the weights assigned to the pixels in the neighborhood.

First a requester shows up to select the filter to use.

At the upper right of the requester the current filter is shown. The
entries are displayed as signed numbers in the range [-63,63], the
entry in the middle corresponds to the pixel that is filtered. The
filter is changed using the gadgets at the left and the bottom of this
display:

   * [Radius] defines the size of the filter. The bigger the filter,
     the slower the processing.

   * [Load Filter] loads a filter saved to disk. Some filters come
     along with this distribution in the TPaint:filter-drawer. Try them
     to see how filtering works.

   * [Save Filter] saves a filter.

   * If you want to change entries of the filter manually, you choose
     the entry to change using the gadgets [X] and [Y]. The entry (0,0)
     is the entry in the middle, that corresponds to the pixel
     currently filtered. The selected entry is marked by « at the
     right of the entry in the display.

   * Using [Val.] you choose the value of the selected entry.

For further informations about filtering see a book about image
processing.

1.47 TPaint.guide/menu0152

'Picture/Prcoessing/Color Correction'
....

This function isn't finished yet. Currently only changing of the
contrast and the brightness of the picture is supported. For both a
value of 100% means no change of contrast resp. brightness.

1.48 TPaint.guide/menu0140

'Picture/Display' - Image Display Control

This menu is used to tell TPaint how to compute the display of the
24bit menory picture. The colors of the display are choosen as well as
the algorithm that uses TPaint for dithering. You may also force a
recomputing of the display.

```
Colors...      |f
Auto. Colors   |a
Recompute      |r
Controlcenter...|t
```

1.49 TPaint.guide/menu0141

'Picture/Display/Colors' – Change the Palette Manually
....

   This menuitem calls the colorrequester to appear. Using this
requester you are able to change the colors that are used to display
the current picture manually. Using this you don't change the 24bit
memory picture, you just change the colors that are used to approximate
the picture on screen. Each picture has it own set of colors, called
palette, so you change just the palette if the current picture, the
other pictures are not affected.

   The colorrequester consists of a couple of gadgets:

   * At the left edge of the requester is the list of the current
     displaycolors of the current picture. At the right of the list
     there is a scrollbar which you can use to scroll through the list.
     On an AA-Amiga this list may be made out of up to 256 colors. The
     current color is marked at the left of the color.

   * At the right of the color list there are six sliders to change the
     selected color. About this sliders see 'Toolbox/Color'.

   * At the very right edge of the requester is a second list of the
     displaycolors of the current picture. This list is used to define
     the range of colors that the following gadgets act on. A range is
     defined by the start and the end of a connected set of colors. The
     range is marked at the left of the colorlist and may be changed by
     you by clicking on the colors. TPaint tries to figure out if you
     think of the start or the end to be changed. If it doesn't make
     the right choice, try again.

   * [Invert] inverts the colors in the selected range, black gets
     white ...

   * [Mirror] mirrors the order of the colors in the selectes range.

   * [Rotate] rotates the order of the colors in the selected range.
     The second color becomes the first, the third the second ..., the
     first color gets the last.

   * [Exchange] exchanges the topmost and the bottommost color of the
     selected range.

   * [Trans. RGB] creates a transition through the world of RGB-colors.
```

The first color of the current range is used as start, the last
one is used as end of the transition. The other colors of the
range are replaced by the colors computed by TPaint.

* [Trans. HSV] is like [Trans. RGB] but travels through the world of
  HSV-colors instead the world of RGB-colors.

* [Load Palette...] loads a palette from disk. You select the
  palette file by the File-Requester. The preset path for
  palette-files is TPaint:palette. The selected file may be a file
  saved with [Save Palette] or an ILBM-file. In the latter case
  TPaint extracts the palette used for that picture. Of course, this
  works only for pictures not saved in 24bit-format. If the selected
  file has more colors than the current screen, TPaint uses only the
  first, if the file has less colors, only the first colors of the
  screen are changed. There are some examples of palettes in the
  TPaint:palette-drawer of this distribution.

* [Save Palette...] is the reverse to the gadget above, it saves the
  current palette in a file on disk.

## 1.50  TPaint.guide/menu0142

'Picture/Display/Auto. Colors' – Computing an Automatic Palette
..............................................................

   If you call this command TPaint tries to compute a palette for the
current picture. The colors are choosen in that way that the display of
the current picture will match the 24bit memory picture as close as
possible.  To achieve this aim TPaint has to count the colors used in
the picture. In the worst case this may be as many colors as the number
of pixels of your picture (for example: a normal PAL:Hi Res picture has
640*256 = 163.840 pixels). But an AA-Amiga has at maximum 256 entries in
the palette, so TPaint has to unify colors. The counting und unifying
can take a very long time, especially if your picture has many colors.
To avoid this you may choose some options from
'Picture/Display/Controlcenter'.

   While TPaint is counting and unifying the colors the current state
of the operation is shown in the status window, if it is active (see
'Toolbox/Status'. Both phases of computing may be stopped by ESC.
Choose other settings in 'Picture/Display/Controlcenter' and restart
this command.

   The first to colors of the palette are out of bounds for this
command. They are left at black and white to ensure that the menus and
requesters keep readable. You may change this two colors manually. This
behavior effects that this command has essentially no effect on a
two-color-screen.

## 1.51  TPaint.guide/menu0143

'Picture/Display/Recompute' – Recomputing the Display
..................................................

   Recomputes the display of the current picture. The options from the
'Picture/Display/Controlcenter' are used for this operation. Especially
Floyd–Steinberg–Dithering could talk a while computing, be patient.

## 1.52  TPaint.guide/menu0144

'Picture/Display/Controlcenter' – Controling the Display
....................................................

   This menuitem calls a requester which allows you to choose some
settings affecting the commands 'Picture/Display/Auto. Colors' and
'Picture/Display/Recompute'.

   The upper half of the requester affects 'Picture/Display/Recompute'.
By toggling the gadget beyond [Dithermethod] you are able to choose
different methodes to approximate the 24bit memory picture by the
displayed picture. Currently supported are the following two methodes.

   * [Nearest Color]: The color in the palette of the display of the
     current picture that matches the color of each point best is used
     to display this point. This is the simplest possible method and
     should be choosen if speed is a problem for you.

   * [Floyd–Steinberg]: This method, invented by Floyd and Steinberg,
     achieves in most cases better results than the other method. Like
     there the color matching the real color of a point best is
     choosen. But in most cases you the choosen color isn't exactly the
     real color of the point. The error made by this choice is
     distributed to the pixels in the neighborhood of the point.

   You should use [Floyd–Steinberg] if you want to get a better
spproximation of the 24bit picture and [Nearest Color] if you want to
get a fast computing of the display. While painting always the first
method is used for speedup.

   The lower half of the requester is used to affect
'Picture/Display/Auto. Colors'. The upper gadget [min. Colordist.] is
used to define a fuzziness that must be exceeded to treat two colors as
different. About fuzziness see the tool Filling Areas. This option
takes effect in the phase of counting colors. The higher the value of
this option is choosen the less colors TPaint will find in the picture.
This speeds up counting as well as unifying. A rule of thumb is to
increase the value of fuzziness if the number of colors TPaint counts
exceeds the double of the number of palette entries of the screen.

   The second gadget of the lower half effects the phase of unifying
while computing a new palette. The two possible selections of this
gadgets are the following:

* [Most Used]: After counting the colors of the current picture the
  colors most often used in the picture are stored in the palette of
  the current picture. This is much faster than the second
  possibility since no unifying is done.

* [Unifying Colors] tries to unify colors. Colors that are very
  similar and not often used are treated as one single color. The
  operation of unifying could take a long time for TPaint has to
  find the colors that may be unified introducing the smallest
  error. You can stop this operation using ESC. If you do that [Most
  Used] is used instead.

## 1.53  TPaint.guide/menu0104

```
'Picture/Fill Collorrange' – Filling a Range of Colors
------------------------------------------------------
```

   If you call this menuitem the first thing you have to do is to
select a color that should be filled. Unlike the tool Filling Areas the
points to be filled need not to build up a connected area. All points
having the selected color in the picture are filled. About the way
filling is done see 'Toolbox/Filling'.

   If you have selected a color a requester comes up to ask for the
fuzziness that TPaint should use for searching the points having the
right color. For more informations about fuzziness and how to define it
see Filling Areas.

   The 'Picture/Alphachannel' takes effect on this operation, so you
may restrict this operation on certain regions of the picture.

1.54 TPaint.guide/menu0103

```
'Picture/Make Transition'
-------------------------
```

   Creates an transition between the four colors selected in the color
selector (see 'Toolbox/Color'). The four colors are placed in the four
corners of the screen as follows: The first one in the upper left, the
second one in the upper right, the third one in the lower left and the
last one in the lower right. After calling this function a requester
appears which allows you to select some atributes to the colors.

   For each of the four colors you can change the scope of this color.
This means what distance from the color the right edge of the
distribution diagram stands for. A scope of 100% means that the right
edge of the diagram represents the width resp. height of the picture.
If you choose 144% the influence of the color extends to the diagonal
corner of the picture. A scope of 0% disables this color. The intensity

of a color within the selected scope can be choosen by the distribution
diagrams just below the sliders. (see Airbrush)

    After clicking [OK] the computing of the transition starts. The
computing may take a while. I suggest to use [Floyd-Steinberg]-dithering
(see 'Picture/Display/Controlcenter') since the results of [Nearest
Color] may be very ugly.

1.55 TPaint.guide/menu0105

'Picture/Undo' - Very important Function

    Performs an Undo on the current picture. Your last action performed
on this picture is undone. If you select 'Picture/Undo' again the
undone action is redone. This menuitem is only selectable if the
current picture has an activated Undobuffer. You can activated and
deactivate the Undobuffer in the 'Picture/Controlcenter'.

1.56 TPaint.guide/Brush

The 'Brush'-menu
================

    This menu contains many functions to work with brushes.

  Cut Brush |B
  Controlcenter |C
  » IO
  » Manipulate
  » Shape
  » Size
  » Turning

1.57 TPaint.guide/menu0202

'Brush/Cut Brush' - Create a Brush

    If you select this command you are able to cut a brush. This is done
the same way like drawing a Box but instead drawing the box actually the
specified section of the picture is taken as a brush. For more
informations about brushes see the the whole 'Brush'-menu.

    If you have selected a brush using this command the Pencil (see

'Toolbox/Pencil') is switched to brush.

1.58   TPaint.guide/menu0201

'Brush/Controlcenter' – Maintaining the Pictures

   Opens a requester which allows you to make some modifications on the
list of brushes. The controlcenter consists of these gadgets:

   * At the top of the requester you find the list of brushes currently
     existing in memory. You can choose a brush by clicking on it. The
     current brush is marked at the left of the name. If you leave the
     requester the marked brush will get the current brush. The current
     brush is that one that will be used for drawing if you select
     drawing with a brush and the one that all operations on brushes
     will be performed on.

   * At the right of the list the width and the height of the currently
     selected brush are displayed.

   * Every brush may have an annotation and a copyright-notice. If they
     exists they are saved along with the brushdata if you save the
     brush. If a loaded brush has an annotation or a copyright-notice
     they are displayed here if you call the controlcenter.

   * With [Duplicate] you can make a copy of the current brush. The new
     brush is exactly the same as the old one.

   * If you call [Delete] the current brush is entirely deleted from
     the brushlist and form memory. There is no way to recover.

   * The gadget [Recompute Display] recomputes the display of the
     currently selected brush. This operation uses the palette of the
     current picture. If you select another picture with another
     palette to be the current picture the display of the brush may get
     absolutely different. If you paint with this picture the
     colorinformations of the 24bit memory brush are used, so you get
     the right results even if the display doesn't match the actual
     brush. The selections from 'Picture/Display/Controlcenter' take
     effect on this operation.

   * If the option [Background transp.] the color black is treated to
     be transparent. If you draw using this brush the original picture
     will stay intact whereever the brush has black pixels. If this
     options is unselected black will be used as a normal color and
     overwrite the original picture.

   * The option [Antialiasing] affects the way operations of the menus
     'Brush/Shape', 'Brush/Size' and 'Brush/Turning' are done. These
     operations map the old brush to a new one. Every pixel of the new
     brush corresponds to a area of pixels and parts of pixels of the
     old brush. If this option is deselected, only the pixel at the
     center of this area is used to define the new pixel. If this

option is selected, all pixels covered by this area are used to
define the new pixel by computing a weighted average of the
totally and partially covered pixel. Try
'Brush/Turning/Any Angle'with and without this option to see the
difference.

1.59 TPaint.guide/menu0210

'Brush/IO' - Loading/Saving/Printing a Brush

   The submenu of this menuitem contains commands for loading, saving
and printing the current brush.


```
  Load...           |L
  24bit Save        |S
  24bit Save As... |V
  Print...          |P
```


1.60 TPaint.guide/menu0211

'Brush/IO/Load'
....

   Loads the brush choosen by you via a File-Requester. In the current
version TPaint is able to load brushes that are saved in the IFF-ILBM
format. Both brushes with colormaps and 24bit-brushes are supported.

1.61 TPaint.guide/menu0212

'Brush/IO/24bit Save'
....

   Saves the current brush as a 24bit brush. This means that the file
doesn't contain the brush you see on your monitor but the brush that
TPaint maintains in memory. The brush on screen is just a approximation
of this 24bit brush that is computed using the colors available on
screen. The filename that is used is the name last used for this brush.
If the brush has been saved before the name used there is used again.
If the brush has not been saved before the name which has been used to
load the brush is used. These two possibilities imply that the old
brush will be overwritten!.

   If the brush has neither been saved nor load before, TPaint asks you
for a name. In this case this is the same as 'Brush/IO/24bit Save As'.

1.62   TPaint.guide/menu0213

`Brush/IO/24bit Save As`
....

   The same as `Brush/IO/24bit Save` except that you are asked for a
filename to be used for saving.

1.63   TPaint.guide/menu0215

`Brush/IO/Print`
....

   By use of this menuitem you can print the current brush. The use of
this command is exactly like `Picture/IO/Print`.

1.64   TPaint.guide/menu0220

`Brush/Manipulate` – Manipulate a Brush

   This menu contains some commands to change the shape of the current
brush.


```
  Invert   |I
  Mirror X |
  Mirror Y |
```


1.65   TPaint.guide/menu0221

`Brush/Manipulate/Invert`
....

   Inverts all points of the current brush according to the RGB color
model. Black becomes white, red becomes cyan, green becomes magenta,
blue becomes yellow and vice versa.

1.66   TPaint.guide/menu0223

`Brush/Manipulate/Mirror X`
....

   Mirrors the current brush in the x-direction, that means at the
y-axis, left becomes right.

1.67   TPaint.guide/menu0224

`Brush/Manipulate/Mirror Y`
....

   Mirrors the current brush in the y-direction, that means at the
x-axis, top becomes bottom.

1.68   TPaint.guide/menu0230

`Brush/Shape` - Knead the Brush

   The commands from this menu take care of the [Antialiasing]-option in
the `Brush/Controlcenter`.

  Shear X
  Shear Y
  Bend X
  Bend Y

1.69   TPaint.guide/menu0231

`Brush/Shape/Shear X`
....

   Shears the currently selected brush along the x-axis. How to select
the amount for shearing see `Brush/Size/Change Size`.

1.70   TPaint.guide/menu0232

'Brush/Shape/Shear Y'
....

     Shears the currently selected brush along the y-axis. How to select
the amount for shearing see 'Brush/Size/Change Size'.

1.71   TPaint.guide/menu0233

'Brush/Shape/Bend X'
....

     Bends the currently selected brush along the x-axis. How to select
the amount for bending see 'Brush/Size/Change Size'.

1.72   TPaint.guide/menu0234

'Brush/Shape/Bend Y'
....

     Bends the currently selected brush along the y-axis. How to select
the amount for bending see 'Brush/Size/Change Size'.

1.73   TPaint.guide/menu0250

'Brush/Size'

     The commands in this menu allows you to change the size of the
current brush.

     The commands from this menu take care of the [Antialiasing]-option in
the 'Brush/Controlcenter'.

  Double X |<
  Double Y |>
  Change Size |G

1.74   TPaint.guide/menu0251

`Brush/Size/Double X'
....

   Doubles the size of the current brush in x-direction. That means
that the width of all pixels is doubled.

1.75   TPaint.guide/menu0252

`Brush/Size/Double Y'
....

   Doubles the size of the current brush in y-direction. That means
that the height of all pixels is doubled.

1.76   TPaint.guide/menu0253

`Brush/Size/Change Size'
....

   This menuitem lets change you the size of the current brush to any
size you want. If you select this command the current brush will be
glued on the mouse. You select the desired size by pressing down the
left mousebutton, moving the mouse until the drawn box has the desired
size and releasing the mousebutton.

   The algorithm to change the size of the brush is very simple, no
interpolation is done. If you enlarge the brush simply some pixels are
repeated and if you scale down the brush some pixels some pixels are
omitted. The result is often not satisfactory and may just be usable as
a draft for you and you need to process the result by hand. To do this
make a Points with your brush, make your changes and cut the result as
a new brush.

1.77   TPaint.guide/menu0260

`Brush/Turning'

   The current brush may be turned.

   The commands from this menu take care of the [Antialiasing]-option in
the `Brush/Controlcenter'.

   90 degree |#

```
  Any Angle |^
```

1.78   TPaint.guide/menu0261

'Brush/Turning/90 degree'
....

   This turns the current brush 90 degrees clockwise. Repeating this
operation four times will result the original brush.

1.79   TPaint.guide/menu0262

'Brush/Turning/Any Angle'
....

   You may also turn the current brush any angle choosen by you. The
selection of the angle and the limitations of the algorithm are the
same like 'Brush/Size/Change Size'.

1.80   TPaint.guide/Toolbox

The 'Toolbox'-menu
==================

   The menuitems in this menu activate parts of the toolbox. The
toolbox of TPaint is split up in different parts. The parts are
activated and deactivated by selecting the corresponding menuitem.
Activated parts are marked by a checkmark in front of the menuitem. All
selections done in one of the parts of the toolbox take effect
immediatly, so you can draw in your picture while parts of the toolbox
are on-screen.

```
  Color...    |^o
  Tool...     |^t
  Pencil...   |^p
  Filling...  |^f
  Zoom...     |^z
  Gemotrie... |^g
  Status...   |^u
```

1.81 TPaint.guide/menu0301

```
'Toolbox/Color'
---------------
```

   Of course you are able to select different colors to paint with. The
part of the toolbox used to select the color comes up on an own screen
at the bottom of the monitor.

   At the left of the colorselector are the sliders to choose the color
to be used. The topmost three are used to select a color according to
the RedGreenBlue color model, which is used internally by the Amiga and
by your monitor. The color is made up out of three components, red,
green and blue. These three components range from 0% to 100%. The
current settings of the compenents are displayed at the right of the
sliders.

   The lower three sliders are used to select a color according to the
HueSaturationValue color model. All possible colors are thought of to be
placed in a cone that balances on its tip. The pure colors (red, green,
blue, ...) are placed at the outer border of the cone. The axis of
symmetrie represents the value or brightness, the distance from this
axis is called saturation and the angle between the position of red and
the choosen color is called hue. The lower the saturation of a color
the more greyish it is. The really greys are all placed on the axis,
they have zero saturation. Therfore the color at the tip of the cone is
the color with zero saturation and no brightness, black. The hue is
mesaured in degree ranging from 0\textdegree{} (red) over 120\textdegree{} (green) ←
    and 240\textdegree{}
(blue) to 360\textdegree{} (red again), saturation and value are measured in
percent.

   To simplify things TPaint maintains four independent color-pots at
the right side of the colorsliders. The current selected pot is marked
and the position of the sliders represents the color of this pot. You
can select another pot by just clicking it.

   Another way to select a color is the gadget [Pick]. If you activate
this, you can pick a color directly out of the picture. The color
pointed at by the mouse is shown in the currently selected color-pot.
If you click the left mousebutton, this color is fixed. You can use ESC
to abort the colorselection.

   In most cases the selected color doesn't match exactly a color used
to display your picture on screen. For displaying the picture the color
best matching your choice is choosen but the 24bit picture in memory
matches exactly your choice.

   On an AA-equipped Amiga you have the possibility to choose 16777216
different colors. I think, that's enough.

1.82 TPaint.guide/menu0303

```
'Toolbox/Tool'
--------------
```

   After activating this part of the toolbox you see a list of the
available painting tools. Each tool is symbolized by a little pictogram
in this list. You can scroll through the list by the scrollbar at the
right of the list. The current selected tool is marked at the left of
the pictogram of the tool. Some of the tools allow special parameters
to be set. If such a tool is selected, the gadget [Tool Spec] is
selectable. If you click this gadget a requester will appear. These
requesters will be described along with the description of the tools.

```
  Points
  Freehand
  Lines
  Box
  Filled Box
  Box with Round Corners
  Filled Box with Round Corners
  Circle
  Filled Circle
  Ellipse
  Filled Ellipse
  Ellipse Arc
  Airbrush
  Filling Areas
  Text
```

1.83   TPaint.guide/Points

```
Points
......
```

   This is the simples tool, used to place single points. About
choosing the shape of the points see 'Toolbox/Pencil'.

1.84   TPaint.guide/Freehand

```
Freehand
........
```

   This tool draws a connected set of lines following the movements of
the mouse. About choosing the shape of the linesegments see
'Toolbox/Pencil'.

1.85 TPaint.guide/Lines

Lines
.....

   Draws a straight line. The starting point of the line is selected by
pressing the left mousebutton, the endpoint by releasing the
mousebutton. While moving the mouse a line is displayed that shows the
position of the line if you would release the mousebutton in this
moment. About choosing the shape of the line see 'Toolbox/Pencil'.

1.86 TPaint.guide/Box

Box
...

   Draws a rectangular upright box. You define the box by defining two
corners in a similar fashion like drawing Lines. The box is not filled
in.

1.87 TPaint.guide/Filled Box

Filled Box
....

   The same as the simple Box except that the interior of the box is
filled in. There exists different possibilities for filling, see
'Toolbox/Filling'.

1.88 TPaint.guide/Box with Round Corners

Box with Round Corners
....

   This tool draws boxes with round corners. The box is defined
analogue to the normal Box. The corners are actually quarters of an
ellipse. The radii are selected by you via the tool-special-requester
which appears if you click the gadget [Tool Spec]. If you call this
requester you can enter the x-radius and the y-radius of the ellipse.

1.89 TPaint.guide/Filled Box with Round Corners

Filles Box with Round Corners
....

   The same as Box with Round Corners except that the interior of the
box is filled in. About filled areas see 'Toolbox/Filling'.

1.90 TPaint.guide/Circle

Circle
......

   Using this tool you can draw circles. You specifiy the width of the
circle, the height is computed by TPaint. The ratio width/height is
specified by the screentype. On a correctly adjusted monitor circles
should appear really round, that means, if you use a ruler the width
and the height og the circle should be the same. For the ratio
width/height depends on the selected screentype, circles may get
ellipses if you reload the picture on another screentype. The shape of
the circle may be changed at 'Toolbox/Pencil'.

   You specify the midpoint of the circle by pressing the left
mousebutton and the width by releasing the mousebutton. While you are
moving the mouse and holding down the mousebutton a circle is drawn to
show you what the shape of the circle would be if you would release the
mousebutton in this moment.

1.91 TPaint.guide/Filled Circle

Filled Circle
....

   The same as Circle except that the interior of the circle is filled
in according to 'Toolbox/Filling'.

1.92 TPaint.guide/Ellipse

Ellipse
....

   This tool is similar to Circle, but you specify the height of the
ellipse as well as the width by using the mouse.

1.93 TPaint.guide/Filled Ellipse

Filled Ellipse
....

   Like Ellipse but filled.

1.94 TPaint.guide/Ellipse Arc

Ellipse Arc
....

   This tool is an extension to Ellipse. First you draw an ellipse the
same way like you do using the tool Ellipse. After specifying the
ellipse you have to press the left mousebutton again to mark the
starting point of the arc. By releasing the mousebutton you specify the
enpoint of the arc. Try it.

1.95 TPaint.guide/Airbrush

Airbrush
....

   This tool draws points. The points are not drawn exactly were the
mouse points to but randomly distibuted on a disk. If you call
[Tool Spec] you can change the radius of the disk using the gadget at
the top of the requester. In the middle of the requester you see a a
diagram showing you how the points to be set are distributed over the
disk. The x-axis of the diagram represents the distance form the center
of the disk and the y-axis the probability that a point is set a the
corresponding distance from the center. The more white you see on a
specific column the more probable it is that a point is placed at the
corresponding distance. You can change the shape of this diagram and the
distribution by moving the five knobs visible in the diagram. They can
be moved up and down. The preset distribution is very near to the
behavior of an actual airbrush but you can choose other very strange
distributions if you want to.

   The shape and the [Intensity] of the points drawn may be changed at
'Toolbox/Pencil'. If you set the [Intensity] to a value about 20% to
50% (or even lower) the behavior of this tool gets more likely an
airbrush.

1.96 TPaint.guide/Filling Areas

Filling Areas
....

    Using this tool you can fill irregular shaped areas. An area is
defined by a connected set of points having the same color. Remember
that you have 16777216 different colors at your choice. For this it may
be possible that you want to fill an area that has nearly the same
color at all points but not exactly the same. The problem is that a
computer is very exact so if want to fill this area the computer would
distinguish between the different colors int this area and fill only
the area having exactly the same color. To say the computer to be fuzzy
you can select a fuzziness by the [Tool Spec].

    Now you have two choices how to tell TPaint about the fuzziness.
First, you may specify the fuzziness in general if you have activated
the option [Fix Fuzziness]. Then you enter the general fuzziness at
[Fuzzines]. This value is used for all filling operations. A
Fuzziness of 0 means now fuzziness, all the points of the area have to
have exactly the same color, the higher the value the mor the colors
may differ. The second way to specify the fuzziness is to deactivat the
option [Fix Fuzziness]. In this case evereytime you start a filling
operation a requester pops up asking you for your choice of fuzziness.
Here you may enter the desired value numerically at [Fuzziness] or you
can select [Add Color]. If you do this you can select a second point of
your picture. The fuzziness is choosen that way that the color of this
additional point belongs to the area to fill as well as the first point.
The value to reach this aim is now displayed at [Fuzziness]. You may
add more points using [Add Color]. If you click [OK] the operation
starts.

    TPaint uses a recursive algorithm for filling areas. This means that
TPaint needs lots of stack if filling great or very irregular shaped
areas. If the stack is to small TPaint takes care of this and stops
filling effecting that not the whole area may be filled. If this happens
you should start TPaint with a larger stack (using the
Shell-stack-command, see Amiga-DOS manual). If you start
TPaint from WorkBench, a stack of 20000 bytes is used which should
be enough for most areas, if not change the stack-value if the
TPaint-pictogram and restart TPaint.

    Remember: TPaint looks in his 24bit memory picture for the colors of
the pixel. Points that seem to have the same color on screen need not to
have exactly the same color in the actual picture in memory.

1.97 TPaint.guide/Text

Text
....

    This tool is used to add text to your picture. First you have to
specify the point where the letters should appear by using the mouse.
Note that you don't specify the lower or upper edge of the letters but
the baseline of the letters. This is the line where the line of

underlined letters is drawn.

    After specifying this point you can start entering your text. RETURN
gets you to the next line, ESC stops this tool. You have to use ESC
before you are able to select any menuitem or tool. If you click
somewhere with the mouse while typing text the following text will
appear where you clicked.

    Using [Tool Spec] you may choose the shape of the letters. After
selecting this gadget a requester comes up. This could take a momemt
because TPaint has to read the list of available fonts.

    This requester has a couple of gadgets. At the upper left the list of
available font shapes is shown. You may scroll through this list using
the scrollbar at the right of the list. The current selected shape is
marked at the left of the fontname. At the right of the fontlist a list
of the available sizes of that fontshape is shown. This list is used
similar to that of the fontnames. Below this to lists there are four
more gadgets. The left three are used to algorithmically generate
additional attributes of the choosen fontshape. The topmost is used to
generate [Bold], the middle one to genrate [under.]=underlined and the
last to generate [slanted] letters. If the selected font shape is
already bold, underlined or slanted these gadgets may have no effect.
If you select the big gadget right at the right of these three gadgets
a example of the selected font shape is shown.

    TPaint uses the standard fonts of your Amiga. This means that all
fonts supported by WorkBench are automatically supported by TPaint too.

1.98   TPaint.guide/Tool Spec

[Tool Spec] - Tool Special Paramters
....

    If this gadget is selectable the current tool has special
parameters. A requester to select these parameters will appear if you
click on this gadget. For a description of these requesters see the
description of the tools at 'Toolbox/Tool'.

1.99   TPaint.guide/menu0305

'Toolbox/Pencil'

    TPaint knows different ways to draw points. The selection done using
the requester invoked by this menuitem takes effect on most of the
Painting Tools.

    At the upper left of this requester is a gadget that is used to
select the thing TPaint draws if you say him to draw a point. The

possible things are:

  * [Point]: A simple pixel is set as point.

  * [Ellipse]: The thing drawn as point is an ellipse. Using the
    gadgets below this gadget you can choose the width and the height
    of the ellipse. Furthermore you can choose the shape of the
    ellipse. Using the diagram at the right you can choose how
    intensity is distributed over the ellipse. The left edge of this
    diagram corresponds to the the center of the ellipse. (see
    Airbrush)

  * [Rectangle]: The same is the previous choice except that the shape
    is not an ellipse but a rectangle.

  * [Brush]: This choice is only availble if a brush exists. If this
    choice is taken the current brush is used as point by the tools.
    This choice is selected automatically if you cut a new Brush.

   At the bottom of this part of the toolbox you find a slider named
[Intens.] which is used to select the intensity that is used by all
drawing actions, that are all Tools and all actions from the
'Picture'-menu altering the picture.

   The higher the intensity is choosen the stronger the tools work. If
you choose an intensity of 100% every action simply overwrites the
affected points of the picture (except the effects of the
'Picture/Alphachannel'). An intensity of 0% is a very effective way to
prevent the picture against any changing. In general the new color of a
point is composed of the old color of the point and the color the tool
tries to give the point. The intensity is the part of the new color
that is contributed by the tool.

   The difference to the 'Picture/Alphachannel' is that the intensity
choosen here is the same for the whole picture while the
'Picture/Alphachannel' has a different value for each point of the
picture.

1.100   TPaint.guide/menu0306

'Toolbox/Filling'

   Also different ways to fill areas are available. The choice made
affects all Painting Tools that fill any area and the command
'Picture/Fill Colorrange'. The following selections are possible:

  * [Color]: An area to be filled is simply filled by the currently
    selected color (see 'Toolbox/Color').

  * [Pattern]: Also fills the area by the currently selected color,
    but additionally the filling is done through a pattern. The
    pattern is a 16 X 16 rectangle containing informations about the
    strength by which the filling is done at the different points. A

new pattern is selected using [Cut Areapattern]. After selecting
this gadget the requester is disabled and you have two choose a
region of the picture that is to be used as new pattern by using
the mouse. The brighter points in the choosen region are the
stronger gets the pattern at the corresponding point. Points of
the pattern with full strength effects the area to be filled with
full intensity at the corresponding points, points of the pattern
with 0 strength prevents the corresponding points from being
affected by the filling operation.

* [Repeated Brush]: This is only selectable if a brush exists. If
  this is selected the area is filled with the brush. The brush
  keeps its original size. If the area is greater than the brush the
  brush is repeated to fit in the area, if the area is smaller the
  brush is cut off.

* [Stretched Brush]: This is also only avaible if a brush exists.
  Similar to [Repeated Brush] an area is filled with the brush. But
  instead of repeating or cutting off the brush it is stretched to
  fit exactly the area to be filled. This stretching could take a
  momemt since TPaint has to compute the shape of the region before
  filling in the brush.

1.101 TPaint.guide/menu0302

`Toolbox/Zoom'

   The part of the toolbox corresponding to this menuitem contains two
gadgets.

   The left one is used to invoke the zoom of TPaint. After activating
this gadget a box is moved along with the mouse showing the size of the
area that will be zoomed. You select the area to zoom by clicking with
the mouse. The interior of the box at mousebutton-releasing-time will
be the zoomed area. After selecting the are the zoom will show up at
the right two third of the current picture seperated from the rest of
the picture by a white line. You are able to perform all operations in
the zoom that may be performed in the unzoomed picture.

   The zoom disappears if you click on this gadget again.

   The right gadget is used to enter the magnification of the zoom. The
value is entered using a slider gadget, the currently selected
magnification is shown at the right of the slider. A value of 5 means
that the width and the height of a pixel becomes 5 times the width and
the 5 times the height of the unzoomed pixel. This means a 25 times
zoomed area of the pixel.

1.102 TPaint.guide/menu0304

'Toolbox/Geometrie'

   This part of the toolbox curretnly allows you to define a grid for
painting. This means that if the grid is activatet you are only allowed
to set points with coordinates that are integral multiples of the
defined x- and y-width of the grid, defined by the sliders at the right
of the toolbox. In fact all tools are restricted to the gridpoints if
this option is active.

1.103   TPaint.guide/menu0307

'Toolbox/Status'

   This menuitem activates the statuswindow of TPaint. It shows the
currently selected tool or operation currently performed, the position
of the pointer or the progression of an operation that takes longer to
perform.

   The statuswindow is opened by default during startup of TPaint.

1.104   TPaint.guide/Key Commands

Key Commands
****

   Additionally to the commands available through the menu and the
toolbox TPaint knows two maybe very helpful commands only reachable by
keys.

   * The first one is ` (at the upper left corner of your keyboard,
     right below ESC). If you have changed the colors of the display
     using 'Picture/Display/Colors' you may have changed the first two
     colors, normally black and white. Maybe you have changed them that
     way that you are now unable to read the menus and the requesters.
     If you type `, the first color is set to black and the second to
     white so that you are now able to read the menus and requesters.

   * The second key command is ~ (found as the shift of `). This is
     the opposite of `, it reactivates the colors you have choosen.

   If you change the displaycolors manually keep these commands in mind.
You'll be surprised how often you will use them.

1.105   TPaint.guide/Error Messages

Error Messages
****

   During startup of TPaint you may get a message like Can't open
xxxxx.library. This means that TPaint can't open the xxxxx.library due
to low memory condition or for you haven't installed the right version
of xxxxx.library. In case of teddi.library, dither.library or
losapr.library the right versions come along with the distribution of
TPaint and are automatically installed if you use the installation
script.

   The first three errors are fatal. If they occur TPaint quits.

   *   Not enough memory for screen
     TPaint can't allocate enough memory to open a new screen. Because
     the screen is very essential for a painting program this is a
     fatal error that causes TPaint to quit. See Screen-Requester.

   *   Multitasking overload
     This error message should be impossible. Each program on th Amiga
     has a set of signals for comunication with other programs. This
     error message says that TPaint is running low on signals. I don't
     now any circumstances under which this could occur since each
     program has its own signal set. This error is fatal.

   *   Not enough memory for menustrip
     There is not enough memory to create the menu. Because the menu is
     essential (it contains the only way to quit TPaint) this is fatal
     and TPaint quits.

   The following errors are not fatal. Perhaps you can't do what you
wanted to but TPaint doesn't quit.

   *   Brush to large
     The current brush is to large for the requested operation. This is
     the case if the result of the brush would be a brush greater than
     16384 X 16384 pixels. I think this size should be enough.

   *   Can't open colorrangerequester
     There is not enough memory to open the requester for defining the
     fuzziness of a colorrange. See Filling Areas about fuzziness.

   *   Can't open controlcenter
     A controlcenter couldn't be opened.

   *   Can't open filerequester
     The File-Requester can't be opened.

   *   Can't open font selector
     The requester to select a font for the tool Text can't be opened.

   *   Can't open inforequester
     The requester of the commands 'System/Info' or 'System/About'
     can't be opened.

   *   Can't open IO-requester

The requester called by 'System/IO-Control' can't be opened

*   Can't open palette
   The colorrequester can't be opened. See 'Picture/Display/Colors'.

*   Can't open specials-requester
   A requester called by [Tool Spec] can't be opened.

*   Can't open shell
   The 'System/Shell' can't be opened.

*   Can't open sizerequester
   The requester to choose the size of a 'Picture/New Picture' can't
   be opened.

*   Can't open transitionrequester
   The requester opened by 'Picture/Make Transition' can't be opened.

*   Can't open requester
   Some other requester can't be opened.

*   Not enough memory, can't change colornumber
   This error occurs if you reopen the screen via 'System/Screen' and
   TPaint can't allocate additional memory for the displays of the
   pictures and brushes if you have increased the number of colors.

*   Not enough memory for computing palette
   The function 'Picture/Display/Auto. Colors' needs memory for
   counting colors and can't get it. Try a larger fuzziness for
   counting colors (see 'Picture/Display/Controlcenter')

*   Not enough memory for new pencil
   The memory for new shape of the thing used as point can't be
   allocated. See 'Toolbox/Pencil'.

*   Not enough memory for new picture
   The new picture requested by you via 'Picture/New Picture' or
   'Picture/IO/Load' can't be created. Remember that the 24bit memory
   picture maintained by TPaint is very memory consuming.

*   Not enough memory for brush
   The memory for the brush requested by you can't be allocated.
   Similar to the error above.

*   Not enough memory for toolbox
   A part of the toolbox can't be opened (see 'Toolbox'-menu).

*   Not enough memory for undobuffer
   The undobuffer can't be allocated. See 'Picture/Undo' and
   'Picture/Controlcenter'.

*   Not enough memory to stretch brush
   TPaint can't allocate memory for stretching a brush to fit in an
   area to be filled. See 'Toolbox/Filling'.

*   Not enough memory for temp. buffer
   TPaint needs some temporary memory to 'Picture/Filter' the

picture, but it can't get the memory.

Most errors occur caused by low memory. If this happens you should
free some memory (disposing pictures or brushes, closing parts of the
toolbox, ending other programs ...) and try again. In the worst case
you should buy some new memory for you Amiga.

The next errors occur in connection with loading a picture or brush.
Additionally to these errors any DOS-Errors may occur while loading or
saving a picture or brush. For an explanation of the DOS-Errors refer
to the AmigaDos-Manual.

*   IFFERR_NO_IFF_FILE
     The file you tried to load isn't a picturefile inf IFF-ILBM format
     and TPaint can't load it.

*   IFFERR_CMAP_BEFORE_BMHD
     The picturefile you tried to load is corrupt. The hunk CMAP
     occured before the hunk BMHD.

*   IFFERR_BODY_BEFORE_BMHD
     Analogue to the one above.

*   IFFERR_NO_BMHD
     The picturefile doesn't contain a BMHD hunk which contains the
     size of the picture and other ifnormations required by TPaint.

*   IFFERR_NO_BODY
     The picturefile contains no imagedata, perhaps it is a palettefile.

*   IFFERR_NO_CMAP
     The file choosen by you to load a palette (see
     'Picture/Display/Colors') doesn't contain any color-data.

The next block of error may occur in connection with printing. The
error starting with PDERR are returned by the printer.device which is
the part of the Amiga that drives your printer.

*   PDERR_NO_MEM
     The printer.device is low on memory.

*   PDERR_CANCEL
     The print has been canceled by you.

*   PDERR_NOTGRAPHICS
     Your printer is not a graphics-printer! Perhaps you have choosen
     the wrong printer in your WorkBench-preferences. Refer to the
     Amiga-Manual.

*   PDERR_BADDIMENSION
     The size you have choosen for the print is to large.

*   PDERR_INTERNALMEMORY
     The printer.device is low on memory.

*   PDERR_BUFFERMEMORY
     The printer.device is low on memory.

```
*       ERR_CANT_GET_PRTDEVICE
   TPaint can't contact the printer.device. Perhaps another program
   is currently using the printer.
```

1.106   TPaint.guide/Bug Reports and Comments

```
Bug Reports and Comments
************************

   I hope TPaint is nearly bug free but I'm sure it is not. I would
like if you could report the bugs you find to me. You are also wellcome
to comment the program either positive or negative or both. It would be
very nice if I could get a feedback.

   If you use paper mail:

     Jens Bender
     Am Krüppershaus 58
     42111 Wuppertal
     Germany

   If you use e-mail:

     bender@wrcs3.urz.uni-wuppertal.de

   One last word: Remember this program is shareware. If you use it
think about registration.

   To register write a letter or an email. It would be nice if you
could add some money -- that would help to keep things going. If you
add some more money for mailing you get the newest version of TPaint
and a printed manual (Currently this doc in printed form). Write it in
your letter if you want to. I don't state anything about the amount of
money, I think you'll make the right decision.

   That's all. Good luck using TPaint.
```