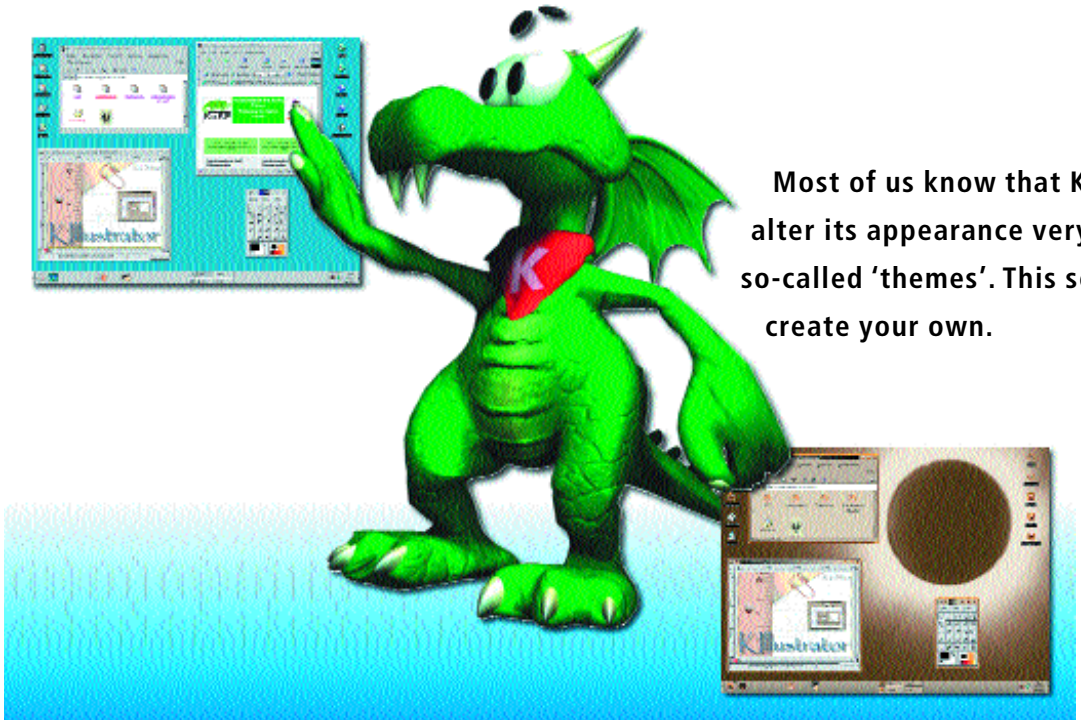
KDE Themes Workshop – Part 3++

PLAYING DRESS UP

BY HAGEN HOEPFNER



Most of us know that KDE has the ability to alter its appearance very rapidly by means of so-called 'themes'. This series describes how to create your own.

This workshop was conceived as a three part series. I now find myself compelled to add a fourth. The notation 3++ is appropriate. This last part takes over where the third one left off – with manual work. In particular, it describes how to port icon themes from KDE 1.1.2 to KDE 2 without having to alter all the icons by hand.

Fig. 1: Icon 32x32 in size

Fig. 2: Icon 32x32 in size enlarged to 48x48

Porting icons by machine

A few things have been happening at KDE. The new version uses *.xpm*, instead of *.png* files as standard for the icons. It no longer stores these by crudely shoving them all into one directory, but into several. Each directory has five subdirectories. The icons must already be classified if an icon set is to be altered via dialog.
To do this, we must firstly use

```
mkdir eclipse2
```

to make a directory with the name of the icon theme. After that we make subdirectories, the names of which reflect the size of the icons. KDE 2 standard icon sizes are 16x16, 22x22, 32x32 and 48x48. As our icons are 32x32 we will now create the directories for icons of this size or smaller (enlarging them would effect their quality).

```
mkdir eclipse2/16x16
mkdir eclipse2/22x22
mkdir eclipse2/32x32
```

In these subdirectories, which classify the icons according to their use, additional subdirectories should now be made. The five classes are:

- actions
- apps
- devices
- filesystems
- mimetypes

That's the long and the short of it. Here are the commands:

```
mkdir eclipse2/GR÷flE/actions
mkdir eclipse2/GR÷flE/apps
mkdir eclipse2/GR÷flE/devices
mkdir eclipse2/GR÷flE/filesystems
mkdir eclipse2/GR÷flE/mimetypes
```

Now, to convert our *.xpm* files into *.png* files we will need a temporary directory. We create this with

```
mkdir xpm2png_temp
```

Listing 1: Script to convert from *.xpm*- into *.png*-files

```
# convert_it
for XPM_FILE in "$@";
do
      PNG_FILE=$(basename $XPM_FILE .xpm)
      convert "$XPM_FILE" "$PNG_FILE".png;
done
```

The directory name in this case can be anything you like.
That completes the basic framework of the icon theme. Next, we copy, using

```
copy ~/eclipse/*.xpm xpm2png_temp
```

Table 1: Differing file names

Filename in KDE 1.1.2	Filename in KDE 2
applications_package.png	package_applications.png
editors_package.png	package_editors.png
games_package.png	package_games.png
graphics_package.png	package_graphics.png
multimedia_package.png	package_multimedia.png
network_package.png	package_network.png
settings_package.png	package_settings.png
system_package.png	package_system.png
utilities_package.png	package_utilities.png
kfm_fulltrash.png	trashcan_full.png
kfm_trash.png	trashcan_empty.png

Listing 2: Script for classifying icons

```
#!/bin/sh
#classify it
DEFAULT_KDE_ICON_TREE=/opt/kde2/share/icons/hicolor
THEME_PATH=../eclipse2

for I in "$@";
do
        convert $I -geometry 16x16 __TEMP_FILE16.png
        convert $I -geometry 22x22 __TEMP_FILE22.png

        DEFAULT_FILE="$DEFAULT_KDE_ICON_TREE/32x32/actions/$I"
        if test -f $DEFAULT_FILE; then
                cp $I $THEME_PATH/32x32/actions/ 2> /dev/null
                cp __TEMP_FILE16.png $THEME_PATH/16x16/actions/$I 2> /dev/null
                cp __TEMP_FILE22.png $THEME_PATH/22x22/actions/$I 2> /dev/null
        else
                DEFAULT_FILE="$DEFAULT_KDE_ICON_TREE/32x32/apps/$I"
        if test -f $DEFAULT_FILE; then
                    cp $I $THEME_PATH/32x32/apps/ 2> /dev/null
                                        cp __TEMP_FILE16.png $THEME_PATH/16x16/apps/$I 2> /dev/null
                                        cp __TEMP_FILE22.png $THEME_PATH/22x22/apps/$I 2> /dev/null
                else
                        DEFAULT_FILE="$DEFAULT_KDE_ICON_TREE/32x32/devices/$I"
                if test -f $DEFAULT_FILE; then
                                cp $I $THEME_PATH/32x32/devices/ 2> /dev/null
                                cp __TEMP_FILE16.png $THEME_PATH/16x16/devices/$I 2> /dev/null
                                cp __TEMP_FILE22.png $THEME_PATH/22x22/devices/$I 2> /dev/null
                        else
                DEFAULT_FILE="$DEFAULT_KDE_ICON_TREE/32x32/filesystems/$I"
                                        if test -f $DEFAULT_FILE; then
                                                cp $I $THEME_PATH/32x32/filesystems/ 2> /dev/null
                                                cp __TEMP_FILE16.png $THEME_PATH/16x16/filesystems/$I 2> /dev/null
                                                cp __TEMP_FILE22.png $THEME_PATH/22x22/filesystems/$I 2> /dev/null
                                else
                                        DEFAULT_FILE="$DEFAULT_KDE_ICON_TREE/32x32/mimetypes/$I"
                                if test -f $DEFAULT_FILE; then
                                                cp $I $THEME_PATH/32x32/mimetypes/ 2> /dev/null
                                                cp __TEMP_FILE16.png $THEME_PATH/16x16/mimetypes/$I 2> /dev/null
cp __TEMP_FILE22.png $THEME_PATH/22x22/mimetypes/$I 2> /dev/null
                                        else
                                                echo "File could not be classified: $I";
                                        fi
                                fi
                        fi
                fi
        fi
done
```

Listings from the article *LinuxMagazine/kthemes/*

the old files into the temporary directory and change, using

```
cd xpm2png_temp
```

to this. The actual conversion functions almost ridiculously simply. We merely write a little script, using our favourite editor (*convert_it*) and release this for execution with

```
chmod +x convert_it
```

and call it up with

```
./convert_it *.xpm
```

After a short time all the *.xpm* files have been converted into *.png*s. We can also leave the classification of the files to a script (cf. *Listing 2*). This gleans the necessary information from the standard icon directory of KDE 2 and classifies our icons. The prerequisite for this is that we are still in the temporary directory when we do it. Take care here, as a few file names have changed. The differences in *Table 1* have been found by testing. These need to be corrected before calling up the *classify_it* scripts by

```
mv old_filename new_filename
```

Once *Listing 2* has been typed, stored in the file *classify_it* and, using

```
chmod +x classify_it
```

has been released for execution, the script is started with

```
./classify_it *.png
```

After that, the names of the files are displayed which could not be classified in any of the above five classes. These are mainly icons which either belong to an application not yet known to KDE 2 or do not in fact fit into these classes (window icons etc).

The script in *Listing 2* also ensures that the icons, correspondingly reduced in size, are sorted into the 16x16 and the 22x22 subdirectory. The scripts may not be resolved in the most elegant way. But they do work. So we now have our icons in the necessarily structured directory tree. All that remains is to create the configuration file *index.desktop* in the *eclipse2*-directory. This has a similar syntax to *eclipse.themerc* and is shown in *Listing 3*. The entries were obtained by re-engineering the "penguin" theme by Ilona Melis (listed as item 4 below). They are consequently not based on "substantiated" research. The individual parameters must therefore not be speculated upon. The ambitious reader should investigate further.

Listing 3: index.desktop

```
[KDE Icon Theme]
Name=Eclipse
Description=Eclipse Icons by Hagen Hoepfner
DisplayDepth=32
Example=exec
Inherits=hicolor
DesktopDefault=32
DesktopSizes=16,22,32
SmallDefault=16
SmallSizes=16
Directories=16x16/apps,16x16/actions,16x16/d➥
evices,16x16/filesystems,16x16/mimetypes,22x➥
22/apps,22x22/actions,22x22/devices,22x22/fi➥
lesystems,22x22/mimetypes,32x32/apps,32x32/ac➥
tions,32x32/devices,32x32/filesystems,32x32➥/
mimetypes,
[16x16/apps]
SIZE=16
Context=Applications
Type=Fixed
[16x16/actions]
SIZE=16
Context=Actionsd
Type=Fixed
[16x16/devices]
SIZE=16
Context=Devices
Type=Fixed
[16x16/filesystems]
SIZE=16
Context=FileSystems
Type=Fixed
[16x16/mimetypes]
SIZE=16
Context=MimeTypes
Type=Fixed
[22x22/apps]
SIZE=22
Context=Applications
Type=Fixed
[22x22/actions]
SIZE=22
Context=Actionsd
Type=Fixed
[22x22/devices]
SIZE=22
Context=Devices
Type=Fixed
[22x22/filesystems]
SIZE=22
Context=FileSystems
Type=Fixed
[22x22/mimetypes]
SIZE=22
Context=MimeTypes
Type=Fixed
[32x32/apps]
Size=32
Context=Applications
Type=Fixed
[32x32/actions]
Size=32
Context=Actionsd
Type=Fixed
[32x32/devices]
Size=32
Context=Devices
Type=Fixed
[32x32/filesystems]
Size=32
Context=FileSystems
Type=Fixed
[32x32/mimetypes]
Size=32
Context=MimeTypes
Type=Fixed
```

Using the icons

Once the icons have been converted and classified, they have to be copied to the right place and then activated. The first is done via a simple copy command. As we are indeed still in the theme directory, we first have to leave this with

```
cd ..
```

Then we send it with

```
cp eclipse2/ ~/.kde2/share/icons/ -rf
```

to where it belongs.

Changing an icon theme in KDE 2 works on a dialog basis. The corresponding dialog is in the start menu under *Settings/Display/Design/Symbols* and is shown in *Figure 3*.

After the dialog has been called up, the icon theme to be used is selected by a left click on the name and confirmed by a left click on the *OK* button. ■

Info

[1] KDE Homepage:http://www.kde.org
[2] The example of an icon theme "eclipse2":
http://kde.themes.org/themes.phtml?cattype=inc&disptype=trad&numthemes=0&boxhide=1&themetxt=eclipse
[3] KDE Themes Homepage: http://kde.themes.org
[4] penguin theme : http://www.ilicon.com

■

The author

Hagen Hoepfner studies computer science at Otto-von-Guericke-Universit‰t in Magdeburg.

Fig. 3: Changing icon themes under KDE 2