

By SacRat and Imagine

Winamp Skin Specification version 2.0 © by Fyre

Version 1.5.0

TABLE OF CONTENTS:

Introduction for the second edition	5
INTRO	5
THE BEGINNING	5
I'M A NEWBIE. HELP ME! (useful links)	6
Photoshop Tutorials.....	6
Other Winamp Skinning Tutorials.....	6
Skin/Art Zines.....	7
SOFTWARE	7
Graphical Editors (general).....	7
Freeware Graphical Editors	7
Skingenerators	8
Button Extractors.....	8
All-In-One	8
Plain Text Editor.....	8
Configuration files generators (general)	9
Viscolor Utility	9
Region.txt Generator	9
ZIP/UNZIP Tool.....	9
Cursor Editor	9
Skin Reloader	9
Color Picker.....	9
WINAMP SKIN FORMAT	10
WSZ FILES	10
GRAPHICS	10
ANIMATION	11
THE SKINNING PROCESS	11
GRAPHICAL FILES	12
AVS.BMP.....	12
BALANCE.BMP.....	13
CBUTTONS.BMP.....	13
EQ_EX.BMP	14
EQMAIN.BMP	14
MAIN.BMP	15
MB.BMP	16
MONOSTER.BMP.....	16
NUMBERS.BMP or NUMS_EX.BMP	16
PLAYPAUS.BMP	16
PLEDIT.BMP	16
POSBAR.BMP	18
SHUFREP.BMP	18
TEXT.BMP	18
TITLEBAR.BMP	18
VOLUME.BMP.....	19
TEXT FILES	19
Pledit.txt.....	19
Readme.txt.....	20
Region.txt	21
Viscolor.txt	23
Winampmb.txt	24

Mb.ini	25
TRANSPARENCY	25
CURSORS	26
WINAMP SKIN SPECIFICATION (© BY FYRE)	27
Overview	27
Getting Started.....	28
Bitmaps.....	28
avs.bmp (Advanced Visualization Studio)	28
balance.bmp (Left-Right Balance Adjuster)	29
cbuttons.bmp (Control Buttons)	30
eq_ex.bmp (Equalizer Windowshade).....	31
eqmain.bmp (Equalizer Module).....	32
main.bmp (Main Module).....	36
mb.bmp (Minibrowser).....	38
monoster.bmp (Mono-Stereo Indicator)	39
numbers.bmp (Time Display Numbers)	39
playpaus.bmp (Play-Pause Indicator)	40
pledit.bmp (Playlist Editor Module)	40
posbar.bmp (Song Progress Bar)	44
shufrep.bmp (Shuffle, Repeat, Equalizer, & Playlist Toggles).....	44
text.bmp (Winamp Text)	45
titlebar.bmp (Titlebar).....	47
Text Files	50
pledit.txt (Playlist and Minibrowser Text Settings).....	50
region.txt (Transparency Settings).....	51
viscolor.txt (Visualization Color Settings)	52
Notes.....	53
BUGZ (don't forget about testing...)	53
RIPS and COPIRIGHTS	54
SKIN PROTECTION.....	54
SUBMITTING	55
MAKING A PREVIEW	56
COMPABILITY	56
HINTS AND TRICKS	56
Small introduction	56
Skin files	56
Images.....	57
Text files	57
ANOTHER COPYRIGHT STUFF	57
OUTRO	57
ABOUT THE AUTHORS.....	59
SacRat:.....	59
Imagine (Gileva):.....	59
GREET'S'n'RESPECTS.....	59
TESTERS	60
GLOSSARY	60

Introduction for the second edition

Hi, man (girl)! Here's the second (updated) version of our winamp 2.X tut. What's new here? First of all, some new stuff, like tricks, hints and other designing perversions. What else? We've tried to rearrange navigation on this work a bit more in order to help you find all the needful information faster. Some stuff was updated, some was changed, bugs were fixed and so on and so on... Version 1.5.0 now comes with **Fyre's** Winamp skin specification, used from his permission.

Don't forget, that this tutorial is constantly being improved, so wait for updates. Where to find them? As always: <http://sacrat.deviantart.com>. (SacRat)

INTRO

Greets to everyone. SacRat's online. The stuff you are reading now is our first attempt to make a complete winamp skinning tutorial. There are lots of winamp tutorials anywhere, but almost none of them gives you a *complete* information about the skinning process, all the tricks and so on... Here we tried to combine all the needful and useful information about the process of skin creation (skinning), from the idea and creating temporary images with templates to packing and submitting the final work. This manual is based as on our own experience, both as on the experience of other skinners. (SacRat)

Imagine's online. Well, I don't really know if it's going to be *complete* – you still may find some information irrelevant, or missing, or you just have your own ways of skinning. We just tried to help you to solve some 'traditional' problems that you will face when making your first skin and to tell more advanced skinners about different 'tricks'. Hope you find it useful! And if you think you could make this tut better – don't hesitate to mail us! (Imagine)

SacRat's back. If you're a novice, you may find this tutorial useful, as it was created mainly for you. If you're an advanced skinner, you could find some interesting tricks/ideas here and if you're a profi – feel free to share your experience with the rest and help us making it better. (SacRat)

THE BEGINNING

So, you decided to become a skinner, that means started creating your own work (skin). What shall you do then? First you must have an idea/design conception for your new work. In order to make your work outstanding or just non-standard, you should have an original idea. For example, if it's just a repainted default winamp skin, then you could skip reading the tutorial and go ahead doing that crap. If not... Then you should decide, what you wanna get: think about the way you skin would look, think about all the animations/effects and so on. In few words, you need a *style*. Don't bother skinning if you have absolutely no idea about what you want. It's not my aim to teach you about all these special affects, textures and so on, but just know: if you want your skin to be used by lots of people, try to make it both nice looking and having high usability. For example, you might create a groovy skin of "toxic green" color, but no one will want to download it, just because it's too bright and eyes get tired from it very fast. The same is about too dark skins: they are hard to use. Whatever you do, think about *usability* as well as about the beauty. A good skin must have quite a good contrast and it's control elements should be easy to find. I've seen lots of beautiful skins, where you cannot find a button you need if only you're not an advanced user. They were really nice, but almost impossible to use. Remember this and try not to sacrifice usability. Original works *can* be beautiful. (SacRat)

A good idea would be to draw some drafts of your skin on paper first (a way to kill time on a boring lesson or lecture ;)) But sometimes trying some tricks may end up with a cool skin, and sometimes a groovy design that you've done on paper may turn out to a crap after you try to paint it in Photoshop. All I can say is – if you're talented, you can ignore any advice and create a good skin, if you're not – no-one can help you! (Imagine)

I'M A NEWBIE. HELP ME! (useful links)

Photoshop Tutorials

Ok, don't cry, kid. First of all this tutorial was made to tell you about all the parts of skin creation process, it's not just a format description. So, if you're not a complete dumb and have some experience of working with Photoshop or any other graphical program, you will understand "how it works". From the other side, you may need some *graphical tutorials/resources*. Here's a couple of links:

- <http://www.eyeball-design.com/> This resource is useful for all those, who uses Photoshop: lots of free tutorials, plugins and stuff like that.
- <http://www.photoshoproadmap.com/photoshop-tutorials-tips.html> Another nice site for Photoshop fans.
- <http://www.project-boredom.com/> There're no much tutorials here, but they are really interesting.
- <http://www.guistuff.com> You may find lots of interesting designs. All they are free after a registration, which is free too :)
- And, finally, <http://www.deviantart.com/>. Here is all you need: brushes, actions, designs, lots of amazing tutorials (well, there's also enough crap in "Tutorial" sections) and links to some other resource sites.

Here's also set of links to some cool sites with Photoshop tutorials, which were taken from *gremlindesign's* (<http://gremlindesign.deviantart.com/>) tutorial:

- [Designs By Mark](http://www.designsbymark.com/) (<http://www.designsbymark.com/>)
- [Photoshop Lover](http://www.pslover.com/) (<http://www.pslover.com/>)
- [Team Photoshop](http://www.teamphotoshop.com/) (<http://www.teamphotoshop.com/>)
- [Phong](http://www.phong.com) (<http://www.phong.com>)
- [Hands-On](http://www.handson.nu/) (<http://www.handson.nu/>)
- [deaddreamer](http://misery.subnet.at/) (<http://misery.subnet.at/>)
- [DeepSpaceWeb](http://www.deepspaceweb.com/) (<http://www.deepspaceweb.com/>)
- [Gurus Network](http://www.gurusnetwork.com/) (<http://www.gurusnetwork.com/>)
- [Effectlab](http://www.effectlab.com/) (<http://www.effectlab.com/>)
- [DigitalInflux](http://www.digitalinflux.com/) (<http://www.digitalinflux.com/>)
- [Fli7e](http://www.fli7e.de) (<http://www.fli7e.de>)

Keep in mind, that lots of Photoshop tutorials could be adapted for some other design programs, like PaintShop Pro, Painter or GIMP.

Almost all the needful utilities could be found at <http://www.1001winampskins.com> in "software" section.

Other Winamp Skinning Tutorials

Here are few links to some other *Winamp skinning tutorials*. You might look for some information there or just use them instead of my tut, if you dislike it for some reason:

- http://members.aol.com/awm3000/skinning_winamp.htm (**Skinning Winamp 2.x: Information & Tactics by Andrew Mackowski**) – some basic information.
- <http://luna.spaceports.com/~wsdskins/> - quite a good tutorial with lots of information.
- <http://www.winamp.com/nsdn> - a very groovy tut. *Must have* for every skinner :) Why? Just because it's the *official* one :)
- <http://fyreskins.plastiqueweb.com/> - a damned great site! If you only need to be explained the format of winamp skins with no redundant stuff, this place is for you. There you could also find links to pages of some famous skimmers. Don't forget to visit this site anyway!!!

Skin/Art Zines

- **Deviantmag** (<http://www.deviantmag.com/>) -an art magazine from the members of *Deviantart*. Also available in PDF version.
- **Razorart** (<http://www.razorart.com/>) - quite a nice magazine. Mainly for skimmers.
- **Teknidermy** (<http://www.teknidermy.com/>) – Another good skinning magazine. Lots of interviews and stuff like that.
- **Twisted Realmz** (<http://mysticspace.net/~twistedrealmz/july2002/index.html>). - Relatively new magazine, dedicated to skins. Not much info yet.

You may also find other interesting resources elsewhere. (*SacRat*)

SOFTWARE

Here's a small list and a description of software you may need while working on your skin. Some programs have links to places, where they could be downloaded, others – don't. So, where to find them? It's quite easy. Lots of this stuff may be found via www.download.com, www.tucows.com or any search engine, like my favorite google (www.google.com). And don't forget to check *software* section on www.1001winampskins.com! Most of the tools, described below are freeware, so you will only have to pay for Internet connection while downloading them. (*SacRat*)

Graphical Editors (general)

What kind of software do you need to create skins? First of all, a graphical editor. Well, some people think, that it's possible to create a good skin in MS Paint and other "heroes" try to make it, but we're just mortals and we need a *good* flexible program, which allows us to draw our skin and don't limit our imagination. Depending on your needs you might need a *bitmap* drawing program or a *vector* one.

Now there are lots of different raster editors and as for me, the best program of that kind is *Adobe Photoshop*. It has lots of great features and it's a very powerful instrument for every graphics creator, but it's very expensive and therefore almost unavailable for most of non-professional designers (I'm talking about legal copies). *Paint Shop Pro* is also a commercial program, which has many fans and includes much better vector tools, than Photoshop. Nowadays Photoshop is the main graphical tool of most skimmers due to it's power and flexibility: you're not given a set of cool effects, but you're given all the needful tools to make your own.

Vector graphics... It's not used that often in the skinning process, but you may need some tools. Professional editors, like Corel Draw, Adobe Illustrator and so on are very expensive, but you may also find some worthy freeware or shareware tools on the web. (*SacRat*)

Freeware Graphical Editors

As it was already said above, the most popular graphical editors are commercial and very expensive for simple mortals. Below I'll try to tell you about some alternative programs, which are not so powerful, but which are free.

- **GIMP** (GNU Image Manipulation Program) (www.gimp.org) – a very nice graphical editor for Linux. There's also a win32 version, which works good too, but which is less stable and has some limitations. The program is being developed by a group of volunteers, who constantly improve it's abilities. It's still a bit less powerful, than Photoshop, but it's the only real free alternative to that commercial monster. It supports layers, has quite a big set of effects and so on... Almost all you may need. By the way, if you're interested in this program and wanna know more, take a look here: <http://www.dezina.org/teknidermy/issueone/Gimp.html>. All the links you may need could be found in this *Teknidermy* article.
- **PIXIA** – one more interesting tool. Unlike GIMP, which is quite similar to Photoshop this tool uses some more original solutions. Filters, layers... That all exists. As for me, it's interface is not that intuitive, so you may need to spend some time, trying to understand "how does it work?". The best solution is to read documentation :) Anyway, it seems to be a worthy stuff.

- *Irfan_view* (<http://www.irfanview.com/>) – It's not a graphical editor, it's a *viewer*. Although it was originally designed to view pictures, but not to create them, this tiny application has quite a good set of instruments for basic image manipulations, like: Filters, resample function, "smart" screen capture, some tools for color correction and so on... Besides, it allows to make conversions between lots of graphical formats. It might be useful for you to have it loaded while you work with your skin: you could easily make screenshots, do some basic image manipulations and so on.

If you are looking for good freeware graphical tools, don't forget to visit www.nonags.com, *Graphical* section. It's a great site, dedicated to free software. (*SacRat*)

Skingerators

There are lots of tools, created specially to ease your work. The first tool of that kind is famous **skinner** (http://members.tripod.com/ajie_g/skinner.htm). You may find skins, made by this "instrument" almost anywhere, lots of them are located in <http://www.winamp.com/skins>. This skinning tool and some other skingerators are known and damned by almost any skinner. What's wrong with it? The business is that it allows you to take almost any image and create a skin from it in seconds using any predefined template. And lots of little (and big) idiots all over the world create skins this way and even upload them into the skin sites (thank God, some of them decline works of that kind) and other idiots download them. All these work look exactly the same way and have very poor quality as well. The only more-less good use of this stuff is trying to use it instead of a *button extractor*, but it's almost the same, as using a tractor instead of a plane... Anyway, it's better, than nothing...

You can always find a list of more-less popular skingerators at winamp.com. For sure, this software is not needful at all, but... here's a link: <http://www.winamp.com/nsdn/winamp2x/>.

If you think, that you're a groovy designer and don't wanna lose time drawing every image, then go and download this stuff, close this tutorial and delete it, you won't need it anymore. And if you wanna know, how to create more-less professional works, just take a look below. (*SacRat*)

Button Extractors

A really needful thing. From now you don't have to create all of your images manually, as you could force computer to make some of your work. Remember, that it *doesn't create* a skin *instead of you*, like *skingerators* do, but eases your work a lot, making one of the most boring and hard parts of skinning, image cutting, automatically.

So how does this stuff work? Very simple: you make a set of three (main window, PL, EQ) images and then use *button extractor* to extract needful components from them. The most boredom part, creating and adjusting backgrounds is therefore done automatically. Piece of cake!

Lots of *button extractors* could be found on the web, but most of them are too buggy or absolutely useless. The best button extractor I've seen is created by **Arteym Tatarov** (mailto:arteym_tatarov@mail.ru) and could be found at <http://arteym.chat.ru/>. It's quite easy to use and small (256 kb archive). Some bugs still present, but it works :) (*SacRat*)

All-In-One

Another interesting application for creating your own skins is **The Cyanide Workshop** (<http://www.sanchit.8m.com/>). It's not a kind of all these stupid skingerators and is just an application, designed to help you making skins. If you don't have a good graphical program, cursor creating program and stuff like that, you could use this one instead. So, this program could be called all-in-one. As for me, I still prefer to use several standalone programs for skinning, but you might like Cyanide Workshop. (*SacRat*)

Plain Text Editor

Besides the graphical editor you may need a *plain text editor*. You could use standard *Notepad* for Windows or try something small and nice, like **Editor²** (<http://personal-pages.ps.ic.ac.uk/~umeca74>), which is good enough and has less, than 50Kb in size (compressed).

Syntax highlighting ability may be useful too, and if you're experienced enough, you could create a scheme for winamp configuration files. If so, you know, where to get such a text editor, and if you're a beginner, then you don't really need it.

Anyway, text editor is not that critical, until you begin to write poems in readme.txt file. (SacRat)

Configuration files generators (general)

It seems to be true, that laziness is the engine of any progress. Even things, like creating configuration files for winamp are nowadays made with the help of special programs... So, what do they do and what tools shall you try to use? (SacRat)

Viscolor Utility

Viscolor Utility (<http://www.geocities.com/SouthBeach/6857>) by *Eugene Loginov* is a simple utility for creating *viscolor.txt* file (see file's description below). Creating this file manually is not hard at all, but if you're lazy enough to calculate all the colors manually, then go and download it. (SacRat)

Region.txt Generator

Region.txt generator is another groovy program. It was created to help you creating "transparent" skins. Read more about it in *transparency* paragraph. (SacRat)

ZIP/UNZIP Tool

Finally, you may need a compression/decompression program. Most of modern archivers already support zip archives' creation/extraction, so if you already have *WinRAR*, *WinACE* or so, you don't need anything else. You may also use *WinZIP*, a commercial program, which is big and only creates zip archives, but some people like it. As for me, I prefer to use fast and small **7-ZIP** (<http://www.7-zip.com/>). You could also find lots of other freeware compression tools anywhere on the web.

What for do you need an archiving program? Take a look at *wsz files* section. (SacRat)

Cursor Editor

If you're gonna make cursors for your skins, it could be a good idea to find a special program for this aim. I've used **Axialis AX Cursors** (<http://www.axialis.com/>) to create some cursors for my skins. The program is quite easy to use and allows you to import cursors files from bitmap images. So you could just create an image with MS Paint, Photoshop or so and create a groovy cursor from it. It's easier, than using standalone programs for creating cursors pixel-by-pixel.

Someone might also like Stardock's shareware **CursorsXP** program (<http://www.stardock.com/products/cursorxp>) or free **Icon Studio** (<http://www.laguna.by.ru>)...

For sure you could look for something, which might fit your tastes better. (SacRat)

Skin Reloader

One more thing you may need is a small Winamp plugin: **Bluecave Winamp Slider** (<http://www.bluecave.net/>). It's most important (for us) feature is that it allows you to reload skin automatically, so you won't need to do it manually. From the other side, you could always do the same, by just pressing *F5*, when your skin's window is active. (SacRat)

Color Picker

And, finally, a tinny tool you may take a use of: **Colorpad** (http://www.entic.net/~vmoya/download_software.html). It's a small *color picking utility*, which allows you to get a value of any pixel on your screen. You may easily find lots of programs of that kind on www.download.com or somewhere else, but this one is my choice. It's very easy in use, small and don't waste any space on your taskbar (it could be accessed from system tray) and supports skins. So what do you need it for? It's more, than useful, when you make text configuration files and need to know the

exact value of color of your skin's fragment or when you wish to pick up a color you like, but don't want to load an image into your graphical editor and so on. Keep it under the hand. *(SacRat)*

WINAMP SKIN FORMAT

What is a skin? It's a set of images, text and cursor files, packed into a single zip archive, sometimes renamed into *wsz* (winamp skin zipped). Old versions of winamp needed these archives to be extracted manually, but now there's no need to do this: winamp "understands" zip files. Images has all the needful graphics for the skin, cursors are the replacements of ordinary winamp cursors (hm, strange...) and in text files there's some information about the skin and configuration data. If you want to find a detailed description of winamp skin format, take a look below or follow some links in *Useful Links* paragraph. *(SacRat)*

WSZ FILES

Most of modern winamp skins are distributed in *WSZ* format, which is simply a renamed *ZIP* file with a skin content. You could rename it back to zip and work with it's files. (Note: advanced users could work directly with *wsz* file, using any modern file manager, like **FAR** (<http://www.farmanager.com/>)). The *advantages* of using *ZIP* compression in skin format are: *ZIP* format is widespread, and zip/unzip source codes are available to anyone, so there are lots of freeware programs, which allow you to make zip compressed files. Besides, there are versions of *ZIP* archivers for any more-less popular platform and operating system: DOS, Windows, BeOS, Linux and so on. The only *disadvantage* is small compression ratio, relatively to modern archivers, like RAR, ACE, 7-ZIP, CAB and so on. In order to create/unpack *ZIP* files, you might need any compression utility (if you don't have the one), like freeware **7-ZIP** (<http://www.7-zip.org/>).

It doesn't really matter, what kinds of files are included into the package besides your skin's files, but try to remove all the useless ones, like thumbs.db (windows' picture cache file) and all the temporary files, as they only enlarge a size of your skin. Hint: you could create a file association on *wsz* files with your decompression utility, so you could view these files' content by just double-clicking on them. (But then you'll have to install all the skins manually, by moving them into your winamp skins' directory). Always make maximum zip compression in your skins: the smaller is file, the more people will download it. *(SacRat)*

Usually skin's size is about 50-200 KB. If it's 300, it's still okay, but if it's over 500 try to do something with it! Some skimmers include a fitting AVS preset or a wallpaper into the archive. There are no problems with an AVS preset – it's very small – but I definitely wouldn't recommend you to include a not-tiled truecolor wp! *(Imagine)*

If you want to make maximum compression and don't wanna lose any quality at all, just try following: clean up all you images, so no fields, which are not displayed in your skin will present in bitmap images. (This could be done automatically by a program, called *decreaser* (<http://www.1001winampskins.com/>)) This simple trick allows you to save up to 20-30% of the space. For example, after that kind of optimization most of the skins in my collection (more, than 1200 files) are smaller, than 300 Kb. But be careful, it could spoil your skins sometimes.

One thing to remember: your *wsz* file's name is a name of your skin in winamp skin browser (Als+S), so naming your work somewhat like 123_My_-=C00l-\$kiN=-..wsz is useless – it's almost impossible to remember this stuff (especially if no one tries to do this). Give your skins *simple* and *easy to remember* names. Not the worst idea is naming them like "yournick_skinname", so all your skins will be browsed in series. A small hint: if a name of your skin begins from a bracket, tilde or any other non-alphabetical character, it comes before the rest in the list of skins. *(SacRat)*

GRAPHICS

The graphics of almost any winamp skin is stored in the following BMP (windows bitmap format) files: avs.bmp, balance.bmp, eq_ex.bmp, eqmain.bmp, main.bmp, mb.bmp, monoster.bmp, numbers.bmp, playpaus.bmp, plectid.bmp, posbar.bmp, shufrep.bmp, text.bmp, titlebar.bmp,

volume.bmp (character case doesn't really matter). There could be fewer files, if a skinner was lazy enough to create some elements, like minibrowser or AVS, and in some cases – more, - when he wanted to include into the package a skinnable plugin's element. For example, avs.bmp is only needful for those users, who install winamp advanced visualization plugin, but as it usually comes with winamp, many skinners also skin this window. From the other side, mb.bmp, which is a winamp's minibrowser skin is skinned not that often: just most don't use winamp minibrowser. Some skinners may include skins for DFX plugin and so on, but they are only needful, if you a user has that plugin installed and uses it frequently.

Few words about bitmap files. Note, that it's not needful to save your files in true color (24 bit) format if you use a very limited number of colors: you could save your images in 256 colors as well without any visible loss of the quality. The advantage is that you will be able to reduce the size of your images and the final wsz file as well. But if you create a colorful skin, you might prefer not to lose any quality and save images in true color. (*SacRat*)

ANIMATION

Creating animation is a very important part of skinning. Think several times before discarding an idea of making it. Without it a skin looks "dead" and good animation affects could "renew" almost any work. One of the samples of good skins, which don't have any animation is NeonG, which could look *much* better with it. Many buttons in winamp (like eq/pl in the main window, on/auto in equalizer and all the min/max/close) has three states: **inactive**, **pressed** and **active** (or "on", "**pressed**" and "off"). Active and inactive states indicate the status of a button/element. For example, when the "eq" button is **active**, the equalizer is visible and when **inactive**, it's not. **Pressed** is an intermediate between these two states and it's image is displayed, when you click the button. Remember, that there are often 2 **pressed** states: when you click the **active** and **inactive** button, so these images could differ. Three images are quite enough to make an animated button (and we have whole four!). For example, you could draw a small dark lamp for the **inactive** state, a bit lighter one for the **pressed** and the bright one for the **active**. These animations are only limited by your imaginations: knobs, switchers, lamps, wheels and so on...

There are also buttons/elements with only 2 states. They are the "preset" button of the equalizer, control buttons of the main window and all the buttons of playlist. You're a bit more limited in creating animation for these buttons, but... there could be much things done with only *two* images. Just take a look at these works (or any others): Korea, Korea2, Alden2, Alden3. (*SacRat*)

THE SKINNING PROCESS

So, how does a process of creating a skin look like? Often it looks somewhat like that:

- You *get the idea* of the future skin and imagine, how it should look like in general.
- Draw a *draft image* of your skin on a sheet of paper or in your graphical editor.
- *Create your skin's extractable images*: work in your graphical editor on the final look of your skin. Use templates to create background images. At the end you should have one (main window) or three images (main window, EQ, PL), which are ready for extraction. Remember to remove (hide) control elements, like sliders and so on from these images, they're not needful or better create two versions: with and without control elements.
- *Extract images* from your templates, using any button extractor.
- *Work with every standalone image*. If you have extracted two images: background and the one with control elements, use them both as two layers, interacting in any blending mode (<http://www.deviantart.com/deviation.php?id=330478>) to draw these controls on a background with all the needful animations.
- *Edit your TXT configuration files and create readme*.
- *Make cursors*, if it's needful.
- *Test your skin* in order to find mistakes, like displaced buttons/missing elements and so on (It's a good idea to ask someone else to do this). Fix all the bugs.
- *Compile your final work* into a WSZ file.

- *Submit it* to any site, if you want others to see it.

No one forces you to do the same or follow the steps above *exactly*. You may wish to choose your own way.

And now we'll try to tell you about all this stuff in more details. (*SacRat*)

OK, you've read (or skipped ;)) all this stuff and begin skinning. You know what applications to use, what basic rules to follow, you have a great concept of your future skin... but you still don't know what images you'll create! Don't worry – read on.

Most skimmers work on their skins in this order:

Main window:	main.bmp titlebar.bmp cbuttons.bmp shufrep.bmp volume.bmp balance.bmp posbar.bmp nums_ex.bmp playpaus.bmp text.bmp monoster.bmp viscolor.txt (a vis. Setup file)
Equalizer:	eqmain.bmp eq_ex.bmp
Playlist:	pledit.bmp pledit.txt (playlist colors)
All the other stuff (some optional if you're too lazy :)):	region.txt (transparency) mb.bmp avs.bmp readme.txt winampmb.txt cursors some plugin skins

Of course, you may do it in any other order. Here are brief descriptions of all these files (in alphabetical order). (*Imagine*)

GRAPHICAL FILES

The graphical part of a skin usually consists of these 16 bitmap images:

If you need to take a look at *Winamp Base Skin's* images, download them here:

<http://ftp.winamp.com/winamp/nsdn/baseskin.zip>. If you don't like default winamp skin (as I do),

because it's too useless, then go here: <http://www.deviantart.com/deviation.php?id=69246> and

download template skin. A template, needful for extraction is located here:

<http://www.deviantart.com/deviation.php?id=69250>. Note that it's a PSD file, so you might convert it to

any other format (if you don't use Photoshop). Btw, already mentioned Irfan View could do this

converting. Thanks to Regener8ed (<http://regener8ed.deviantart.com/>) for the last two links. And now

you might read a description of (almost) every graphical file, which could be found in a skin's folder.

Btw, don't forget to visit <http://fyreskins.plastiqueweb.com/> for more info about this stuff. (*SacRat*)

AVS.BMP

Advanced Visualization Studio window is one of the easiest windows to skin. It consists of one image only and is very easy to create. Note: unlike MB or PL, AVS uses resize for the borders instead of tiling.



Blue rectangle on the top is the only header (active and inactive). It's non-resizable, so you may use for an inscription or so. Note: a header will always be located top-left. Red square on the right is a header's resizable area. So, when you enlarge your AVS window, this square will be tiled. Rightmost square is AVS "close" button (the leftmost top area is a pressed close button). A row below the footer (left to right): black stripe, magenta and orange stripes are footers, which are located below the header and follow the same rules: the magenta rectangle is resizable and an orange one is the bottom left corner of your AVS window. You may draw something there to show, that it could be used to adjust the windows' size. Two vertical rectangles on the left of an image are left (the left one) and right (the right one) borders of AVS window. Just note, that they are resizable too and located between header and footer. (*SacRat*)

The green square in the right top corner is the 'close' button. Sometimes a pressed 'close' is placed in the top left corner of the window, but it isn't really displayed anywhere on the skin (bug?) (*Imagine*)

BALANCE.BMP

The first interesting control element in winamp skin's format, as it allows you to use do some cool tricks with it. It has 28 background rectangles for different states of balance, and two rectangles below them, which are sliders – green is used for normal display and red is pressed.

The most primitive way to create balance is to make no animation with it. For example, you draw a simple stripe and a slider, which is moving on this stripe. No special knowledge is needful to make it. A very trivial solution.



Classical balance also has a slider or so, but the stripe it's sliding on changes it's size or color while sliding. You could use simple animation, but not too much, as a slider button's image could be a hindrance for this. The principle of making animation is: *center* balance's state (*no balance shift*) is *the highest* in balance.bmp image. *The maximum shift is the lowest* part of the image and there's *no difference* if you're moving a balance slider left or right: they will be the same.

A modern way of creating a balance is used mainly by advanced skimmers and it provides you with a better result, then the previous ones. All you need is just to *crop* your image in order to cut sliders, which are located on the bottom. Now you have 28 positions - independent states of your balance slider. So you are not limited in animation, as before: you could make a slider of any size (inside a form) and shape. For example, I've used this method in my Ballzzz (<http://sacrat.deviantart.com/gallery>) skin series to create a slider in the shape of a ball. There are more interesting solutions, like knob (Nucleo_NLog skin), but most of the people still prefer to use sliders of different kinds. Don't limit your imagination: you could display digits, could make your own slider move in almost any direction, give it almost any shape (not just a rectangle, moving forward-backward).

Hint: don't remove the empty field on the left of the image (marked blue on our illustration): it's needful for the balance to be displayed properly.

Here and below remember one important rule about cropping the images: each image is placed, based on it's left-top corner, so if you wanna reduce the size of a picture painlessly, you can only remove bottom and right parts of it.

If you don't wanna make a balance slider, you could crop the image after extraction, as described above and leave it this way: you'll see no animation while moving the slider. (*SacRat, Imagine*)

If your volume and balance sliders look exactly the same, there's no need to create balance.bmp at all! Winamp will use volume.bmp for the balance slider. (*Imagine*)

CBUTTONS.BMP



This image contains control buttons: previous track, play, pause, stop, next track and eject in two states: normal and pressed.

What you need to remember: normal state buttons are on the top, and pressed – on the bottom.

Note, that *eject button is one pixel more thin*, than the rest of the buttons and that *bottom eject button is one pixel lower* (relatively to other buttons), than the upper eject. So be careful, if you just copy buttons. What kind of animation to make? Anything you want. Most of the skimmers choose one of two animations: “normal button – highlighted button” or “normal lever – activated lever”. There’s not too much to chose from. Hint: you could make your buttons bigger, but the area, which could be changed, will still be limited by a size of cbuttons.bmp image. Also keep in mind, that you can change a form of your buttons (but don’t forget that the true area of each button is still a small rectangle), so they won’t look like classical rectangles, you may even leave them with no borders. Take a look at: Vida, blade, Blornge, Blue jeans, Blueprint and so on. (SacRat)

If you don’t want any animation, you can crop this image to 1 pixel – then only the buttons from main.bmp will be displayed (but read *Animation* paragraph first, - note by SacRat). (Imagine)

EQ_EX.BMP



An image, which has winshade windows of the equalizer. Winshade is a mode, when you minimize the window into a tiny stripe, which doesn’t take much space on your desktop. The “active” state (when it’s in focus) of the window is

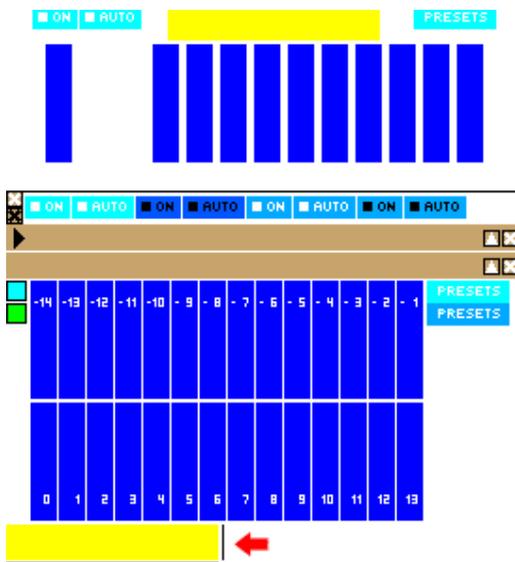
above. The buttons, which are marked white are active and black – inactive. A big black rectangle is a volume slider and the smaller one – balance. Remember, that they are only backgrounds. Pair of blue sets of vertical lines above the buttons are sliders themselves. The principle is easy: when a slider is below the middle it is displayed as the leftmost line (2pixels), when it’s on the middle – the middle one and later – the right. So left sliders are made for volume and right – for the balance. There should be no difficulties while skinning this window. (SacRat)

Notice that ‘maximize’ button has four states – ‘active’, ‘inactive’ and two ‘pressed’ states which is between them. The upper ‘maximize’ button is the ‘pressed’ state for the normal (not winshade) mode and the lower – ‘pressed’ state for the winshade mode. Where are the ‘active’ and ‘inactive’ buttons? They are just on the titlebar of the window. Unlike ‘maximize’, ‘close’ button has two states – ‘normal’ and ‘pressed’ (upper and lower on the image). The ‘close’ button on the titlebar won’t be displayed anywhere. (Imagine)

EQMAIN.BMP

The main window of the equalizer is located here. All the needful elements, including sliders and buttons are included into this file. A pure hell to the skinner. First of all, almost nothing could be done with the damned sliders, so you’ll have to invent something to make ‘em look better. After extraction you may get something like that:

Under the “background” of your EQ there is on/auto and “close” buttons’ animation. White cross is for normal state, black is for pressed. (See EQ_EX.BMP for more info about min/max/close buttons). On/auto buttons are placed in this order: inactive – active – pressed inactive – pressed active. Below there are two titlebar states – active (upper) and inactive (lower). 28 blue stripes are sliders (background), and on the left there are sliders themselves – cyan in normal, green is pressed. On the right there are two states of ‘preset’ button – normal and pressed. The yellow rectangle below is the ‘oscilloscope’ window. Notice two one-pixel black lines on its sides (marked with a red arrow) – the vertical one is used to display the ‘sinusoide’, as I call it, and the horisontal line will be moved whed you move the ‘preamp’ slider. Of course, you can fill the right stripe with a gradient. A good and fast way to create a nice-looking oscilloscope is to copy-paste one-pixel stripes from somewhere on the window and then adjust their hue-saturation to anything you like. (Imagine)

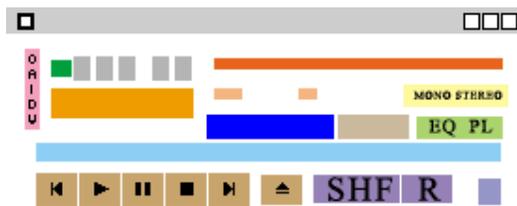


Tricks: you could cut off the region, which indicates the equalizer's state. It's useful sometimes. Other heroes might also cut off sliders and only leave a shape. What for? Many fans of the female sections try to find as much "free" space, as possible, even by a price of usability. I think, that if someone uses the EQ, he also has a right to see it's state at any moment and adjust it as he wants, so try not to leave the empty equalizer. (*SacRat*)

If you're trying to 'hide' your sliders without killing usability, keep in mind that slider stripe changes each time the slider gets two pixels up or down. That means that e.g. if you have some kind of a stripe under your slider, you should rise it by two pixels for each state stripe (look at eqmain.bmp of any skin that has unusual sliders, if you don't understand). It took me a whole lot of time measuring

every pixel to create wheel-sliders for my FuturA skin! (*Imagine*)

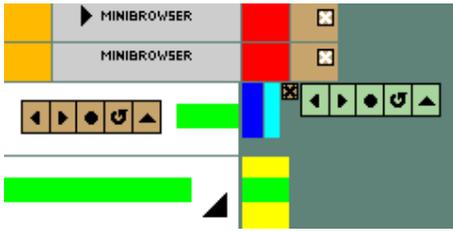
MAIN.BMP



It's the main image in your skin. This window is often made first and then used to extract buttons from it. Just remember, that you better leave some of the elements of this image, like posbar untouched after the extraction: they will be shown, when no music is played. You may also place a logo in the visualization window, which will be shown this way too. This window is often made first and

then used to extract buttons from it.

The elements of this image file are (as shown on the picture): header (gray) on the top *is not* usually *shown* in your final skin, as it's often being replaced by a header from titlebar.bmp (well, 'till you've made something with it). But you may create a header here and then extract it to your titlebar.bmp. The same is true about the buttons: menu (on the left of the header), min/max/close... they are not normally shown but they could be used for extraction. Five gray rectangles below are places, where the time will be shown. The leftmost rectangle is used to display minus or stays empty, depending on the time display mode. Four others show minutes and seconds. The green rectangle on the left shows play/pause/stop sign (which are defined in playpause.bmp). An area below them (orange) is used for visualisation purposes. Keep in mind that it's shown as it is drawn in the main.bmp only when the song is stopped. In other cases it's view is described in viscolor.txt. One small trick is continuing (or reflecting) visualization downwards (you can make a screenshot of your skin's visualization and then fade or transform it), as there's some free space between your visualization area and posbar. A pink area (on the left) is used for some additional buttons (from top to the bottom): options (the same, as right clicking on the main window), always on top, info (the same, as options a view file info Alt+3), doubled mode (Ctrl+D), advanced options. Big red stripe on the right is a song name scroller, where the song title is being displayed. Depending on your winamp *settings* it could be displayed by a default font or font from text.bmp. Two small orange boxes below are used to display song's bitrate (the left one) in kbps (kilobites per second) and the frequency in kHz Mono/stereo indicator's field is indicated by yellow. Blue, light brown and lightgreen rectangles under it are volume, balance and eq/pl placeholders. Big light blue stripe is a position bar. Don't forget: when the song is stopped, you will only see a position bar from your main.bmp window, but not the image from posbar.bmp. Below there are 6 control buttons (from cbuttons.bmp) and shuffle/repeat switchers. The rightmost rectangle field is a place, where normally a winamp logo is shown. This is *not a button*, just an area, clicking which activates winamp help. It's almost never noticed by skimmers due to it's useless. (*SacRat, Imagine*)



MB.BMP

A damned minibrowser's window. Why damned? Just because almost no one uses it. It's very easy to skin (almost like the AVS window) and I won't pay it any special attention, as there's nothing interesting in it. Just don't forget about repeating fragments.

First two rows on the image are headers. The leftmost image (orange) is the left corner, the gray one is a header itself (it's a good idea to place text there), the red rectangle is a repeating fragment and the rightmost (light brown) is a right corner of our header. "Inactive" header is below. Two sets of rectangles, light brown and green, are buttons (back/next/stop/refresh/open URL) The rightmost set (green) displays a "pressed" state of the buttons. A green rectangle, divided into three parts is a place, where the opened URL will be displayed. Just note, that the yellow fragment is repeating. And, as you might have already guessed, the black cross is the pressed state of "close" button. (*SacRat, Imagine*)

MONOSTER.BMP



Mono/Stereo indicator. Most of people don't pay much attention to it, but here I'll try to tell you a small trick I used in some my works. The image is divided into four parts, each of them indicates a state of mono/stereo sign at any time. In many cases skimmers use two states: mono is on or stereo is on, so *both* indicators are visible at the same time. What if we don't want it and only need to know if the song we're listening to is mono or stereo? We create two lines of text (in Photoshop, for example): "mono" and "stereo". Then *rasterize* them and cut each on the middle, so we got 4 text fragments: "ste", "reo", "mo" and "no". Now we place them this way: "reo" is in the left upper corner, "no" is below, "mo" is in the right upper corner and "ste" is below. Now we've got a "steno" indicator, as I call it: it displays only "stereo" while playing a stereo song and "mono" while playing a mono one. But when no song is being played it just indicates meaningful "steno" :). You could use this trick to create your own mono/stereo indicators.

Well, if you don't need that, you could make a classical indicator by following these rules: active "stereo" is left up, "mono" is right up and inactive states are located just *below* them. All is very simple!

Again, if you don't want any mono/stereo indicator, you are free to leave it empty after extraction. (*SacRat*)

☐ 123456789 - NUMBERS.BMP or NUMS_EX.BMP

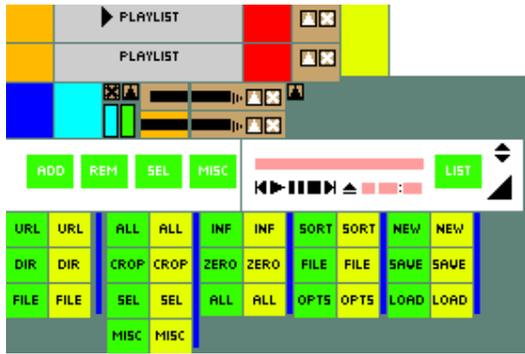
Each of them is used to indicate numbers in your main window's time display. In both cases you should begin from zero and end at 9. I recommend you to leave only *nums_ex.bmp*, as it also has a "minus" sign (on the right), displayed when your winamp's time indicator is set into the "time left" display mode. You might place anything in these images, instead of simple numbers, but remember, that unlike some other indicators, "time" is used very frequently, so it should be quite contrast and understandable. (*SacRat*)



PLAYPAUS.BMP

Play/pause/stop indicator, which is usually located (if someone uses it) in the main window left to the numbers, which display time. Has three states: playing (music is on), paused (the song is paused) and stopped (music is off). One simple small image. Remember, that no one forces you to use the standard shapes on it: you could try something more specific. (*SacRat*)

PLEDIT.BMP



thin stripe on the right (blue), which will be placed on left. ‘Misc’ button, which is below other buttons of the ‘rem’ group, in fact will be placed above all them. To make things even worse, all the buttons are separated with 1 px. stripes. Easy, isn’t it? ;) I dunno what a madman invented this! To make PL look best you would apparently need to make such elements as stripes in the left transparent, and if you’re going to do it manually you’ll spend hell much time on it. I know it, because I did so when designing my first skins. Don’t repeat my mistakes – download a button extractor program, which will save much precious time. And in any case you’ll need a template.

On the very top there are headers, upper is active. Orange and light brown rectangles are left/right corners, grey is the header (it’s wide so you can place the skin’s title or anything else there), and the small red rectangle is the repeating fragment. On the right there is one more repeating element (yellow) from the lower part of the window. The bigger white rectangle will also appear on the bottom as you stretch the window. I think it’s supposed for visualization, but I’ve never seen it there :) Just a good place to put your logo. Below the headers all the elements are cluttered. Blue and cyan rectangles are left/right edges of the window, pressed maximize/close buttons are easy to recognize (maximize button is for ‘normal -> WS’ state). There is a small, but irritating bug with this button – if you’re testing your skin and see that this button is moving in an odd way, blame Nullsoft, not yourself! Below these buttons there are sliders (cyan is normal, green is pressed). Next goes windowshade – the black rectangle is for text, the orange part will be repeated (playlist ws can be stretched, too). There is one left part for active and inactive states, and a right part for each of them (active is above). Apparently you won’t mark active state in any way, but who knows... :) Finally, the rightmost little button is pressed ‘maximize’ (WS -> normal). Wide white stripe with buttons and other stuff will be on the bottom of the window (notice the 1-px stripe where the already described repeating element will be placed). Green buttons with words ‘add’, ‘rem’, ‘sel’ and ‘list’ are, naturally, ‘add’, ‘remove’, select’ and ‘list’ (file commands) buttons. In the pink windows time information will be placed. There are also small not-animated control buttons, which someone may use instead of the main window’s buttons. Anyway, no one does it, so feel free to ‘kill’ them and put anything on their place. The same is with black triangles for scrolling the playlist (everyone uses the slider) and the big triangle that is supposed to prompt that the playlist window is resizable. Below there are that damned buttons that I’ve already described. Phew, that’s all! (*Imagine*)

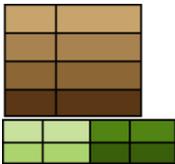
Some people trend to separate add/rem/sel/misc/list buttons on the playlist with the animated popup menus. I don’t think, that it’s a good idea. Popups look “alien” there. Generally speaking there are skins, like Windows 98, where the idea of popups over main buttons is realized very good, but in many cases it’s not true. For example, if you have round buttons and then create rectangle popups over them it may look terrible. You could use a very good and well-known method (Vapor or almost any modern skin), when your buttons slowly turn into popups. It’s not too hard. For example, let’s take the *add* button. When you press it, you get two more: URL/dir/file. So if we make file (the lowest one) look like add, then there’ll be an effect of buttons, which just roll up from “add”. Tricky? Not at all. Another good idea is to create a “transparent” popup. You just give their background the same color, as we have given to your playlist. It’s very easy, but you create an illusion of the lack of background! (*SacRat*)

POSBAR.BMP

Posbar is a winamp's position bar (wittily, right?), which displays a position of the song and allows you to get fast access to almost any song's fragment.

This one has almost no animation and is very easy to create. The only known and usable trick with posbar is that you could make the image a bit thinner, getting a smaller bar this way. This is sometimes useful, when you need to get a smaller slider in order to save space or to make it a bit less visible. You can even cut off sliders of your posbar, but I *don't recommend* you doing that, as it turns your skin's usability into zero. The same is true for all the tricks with smaller sliders (cutting off the right part of posbar.bmp): this only brings you bugs.

On the picture the rightmost green rectangle is the area of a pressed button, the middle one – normal and the big rectangle is a slider's background, which is shown when the song is played (read *main.bmp* paragraph for more details). (*SacRat*)



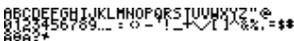
SHUFREP.BMP

This image contains *shuffle/repeat* buttons and *equalizer/playlist* switchers. They both are *3-state* buttons, so there's a lot of possibilities for animating these buttons. You could also *combine* shuffle and repeat buttons into a single one and do the same with EQ/PL. You could see this kind of buttons in *my Ballzzz 3 Honey* skin or somewhere else.

It's important to remember the place of each animated picture. Don't muddle them up. Brown rectangles are shuffle/repeat buttons. "Repeat" image comes *first!* (it's on the left) Nevertheless it's located *on the right* relatively to shuffle in your skin. So don't bother yourself changing their places, just use button extractor. There are four images of each of these buttons: off, pressed and on plus the pressed state for off → on transition. Top rectangles are used for "off" state. Then pressed, on and off → on pressed state.

Green rectangles are eq/pl switchers. EQ is on the left and PL – on the right, so it's easier to use them. Left-top fragment is inactive state, the one below is active and intermediate states are on the right of each of them (so off → on is top-right). (*SacRat*)

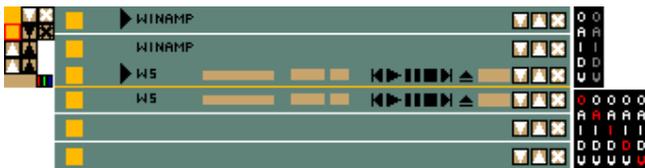
Again, you can cut this image (like almost every element's image) if you want to kill animation and reduce filesize. (*Imagine*)



TEXT.BMP

An image, which has your main window's font, which is used to display artist and song title. It's also used in bitrate/kHz indicator and in playlist to display time. A thing to remember: while perverting with the font, keep in mind, that *the same font* is displayed in several independent windows, so it might be hard to use some effects with it. Another limitation is that a size of a "cell" for each symbol is very small. (*SacRat*)

Note that a one-pixel vertical stripe between letters 'A' and 'B' is also used to fill 'free' space when a song's name is short. You also shouldn't put anything after the '@' symbol – this part of the image is used as spaces in one of the playlist's time indicators. (*Imagine*)



TITLEBAR.BMP

This image contains main window's titlebar, winshades, min/max/close and 'options' buttons animation plus clutterbar. There are two titlebar states – active and inactive (active above), the same with winshade, and below these four stripes there are two more, which aren't displayed anywhere. You may type your name there and protect your skin from ripping.

In the left corner there are all the min/max/close and options buttons in all possible states. 'Options' button is the orange square, stroked with red when pressed. Many skinner prefer not to draw this button

(it's absolutely useless), so you can leave it as it is after extraction. Next there go 'minimize' and 'close' buttons, white are normal, black are pressed. Below them there are four states of the 'maximize' button (from left to right, from top to bottom): normal, normal -> ws, ws, ws -> normal. Usually they are drawn like in this image, as if they have only two states, like 'close' and 'minimize'. But you can use all the four states to make something non-standart, or make them have two states, but not 'normal'-'pressed', but 'active'-'inactive' (like in my skin 'Greenie'). The brown rectangle is the posbar background (winshade) and the three small rectangles (red, green and blue) are the posbar sliders – red one is displayed then the slider is below the middle, green – on the middle, blue – over the middle.

The 'normal' titlebar has nothing special on it – I've left buttons, but they won't be displayed anywhere. Just write there your skin's name if you wish. Winshades have a little bug – that orange 1-px stripe will be displayed in both active and inactive states. Just stroke your winshade with an 1-px border and you'll get rid of the bug. In the brown windows time information, posbar and visualization will be placed. Black icons are the control buttons, bu they don't have animation.

Clutterbar (O A I D V buttons) is on the right of the image. Pressed buttons are marked with red. Note that when you press a button, not only the pressed button, but all buttons are changed. But when the 'always on top' or 'double size' button is active, only the 'pressed' state of this button is shown, the rest of the clutterbar is taken from the above part (where all the buttons are white). In a few words – you can always try to create something unusual, but in most cases you'll get only bugs. Mind that besides 'normal' and 'pressed' states there is one more, 'invisible' state (gray buttons). It appears when you uncheck 'Always show clutterbar' in Winamp display options. You may use it to create some interesting effects, but usually this 'invisible' state is just a copy of the 'normal' state. (*Imagine*)

VOLUME.BMP



All, what was said about skinning a *balance* is right about the volume: you could crop the image to get better animation and get more interesting effects (like in Nucleo_Alien_Mind, Korea), than by using classical volume bars with sliders. On the image the upper (*top*) "cell" is corresponding with the *minimal* volume and the *lowest(bottom)* – with it's *maximum*. Both volume and balance bars provide you with much more animation (up to 28 independent images), than any other skin fragment. (*SacRat*)

TEXT FILES

There are usually five (or less) text files in winamp skin: `pledit.txt`, `readme.txt`, `region.txt`, `viscolor.txt` and `winampmb.txt`. Well, the last one often absents. You could find a detailed description of every file below.

Important note: try not to use characters, which do not present in Latin alphabet to avoid problems: they might be displayed incorrectly on some computers. Why? Winamp is only able to display windows character sets, which differ for every country and if you use some letters of your local alphabet, they will be replaced by according characters from the alphabet of the user, who's looking at your skin's description. The best solution is to try to use *transliteration*. So, if you write your text in Spanish, German or (oh, no!) Russian and so on, try to avoid using these characters where it's possible. Note that symbols, like "©", "%", "\$" and so on will be displayed correctly. The last thing to know about text files is to avoid using non-Windows codetables, like KOI, MAC and so on... Unfortunately Winamp doesn't even "understand" *Unicode*, which could solve all these problems.

So the easiest way to avoid all the problems above is to write all the texts in English and type them in Windows Notepad :). (*SacRat*)

Pledit.txt

This file should look somewhat like that:

```
[Text]
Normal=#A6A6A6
```

```

Current=#404040
NormalBG=#F2F2F2
SelectedBG=#F8F8F8
MbFG=#404040
MbBG=#CFCFCF
Font=Arial

```

A structure of this file is very simple: after the [Text] line there are fields and their values. What do they mean? Normal is a foreground (font) color, which is used to display a song's name in winamp playlist. Current is a foreground color for the currently played song. NormalBG is simply a background of the playlist and SelectedBG – a background of the selected song (the one you have just clicked on). MbFG and MbBG are the minibrowser's foreground and background colors. If you don't want to skin minibrowser, just skip these fields. Font is a name of the font you use in your skin. Try not to use rare or non-standard fonts or, if you do so, place the TTF file into the skin's directory. Note: all the values are given in HEX. So "pure" green color (0R,255G,0B) will look like #00FF00. Use any calculator to convert decimal values into hexadecimal. There's no need to do this, if you use *Colorpad* (take a look into *skinning applications (graphics)* section), as it could give you both HEX and decimal values. (*SacRat*)

Such graphical editors as Photoshop provide a hexadecimal value of any color in the "color picker" box, so you can copy it and paste in your text editor. (*Imagine*)

Readme.txt

Possibly, the most interesting text file in the skin package. It's the text, which is displayed in your winamp skin browser's description window. You may write there whatever you want. Usually skimmers include following information: skin name, skin version, release date, short description, copyright note and the author's contact information. If you're submitting a skin somewhere, then it's not the worst idea to include some information about the creator. The final version could look somewhat like that (a fragment from my *Ballzzz 2 Futura* skin's readme.txt):

```

Ballzzz 2 Futura by SacRat
ver: 2.0.0 final
(skin name and version)

```

```

Ballzzz Futura by Taras Brizitsky (AKA SacRat): sacrat@newmail.ru
Suggestions and bugreports are welcome.
http://sacrat.deviantart.com
(short contact info)

```

Information:

Hope, this is what you were waiting for: Ballzzz 2 Futura. Modern design, minimalistic style...

[skipped]

Small description of the work and a list of new features.

Terms of use:

-Ballzzz Honey for winamp is © 2002 by Taras Brizitsky AKA SacRat.
 -This skin is copyrighted freeware for any noncommercial use or distribution.

[skipped]

A copyright note/terms of use. Just some useful information for those, who might use your skin. In few words it could be described like: I'm the original author; you've got it for free, so you cannot take money from it; you cannot rip it and if you want to place it somewhere, let me know. Note: some skin sites (like <http://www.1001winampskins.com/>) decline submitting skins, which are not freeware.

Greetings to following Deviantart members:

```

-Aggie
-Imagine AKA Gileva
-Iliks
-Jark
-Olya

```



```

;1. Type [Normal]. Real simple that one. (My guess is it is just for forward
compatibility)
;2. Just type the NumPoints= and leave the number blank, we'll fill it in later
;3. OK, now for the fun part.
;   The co-ordinates you should type in are the OUTSIDE pixel co-ordinates in x,y
format.
;   Start at the top lefthand corner and work your way CLOCKWISE around you
polygon.
;   Now WinAmp ONLY accepts the PointList as one line of comma separated variables.
;   You can use spaces, but DONT GO ONTO A NEW LINE. Clear? Good. :)
;   One last thing, don't type the first position again at the end, WinAmp joins
them.
;   Putting a space between each pair is simply common sense, right?
;4. Once your done, count the number of co-ords, and fill in that number for
NumPoints.
;
;Oh, as a side note, the x variables go from 0 to 275, and the y from 0 to 116.
;So if you look at the first example you can see I've gone across the second top
line, gone down a bit, ducked in one pixel, gone down to the bottom, and across and
back up the other side, putting another ledge on the other side.

;This does the outside border, and chops some bits out on the top and edges
;[Normal]
;NumPoints=8      ;Do this LAST!
;PointList=0,1, 275,1, 275,14, 274,14, 274,115, 1,115, 1,14, 0,14

;Cool heh? Very subtle effect. Now lets try a more complex one, with multiple
polygons
;For your first hard(ish) mask, I suggest defining each element as a separate mask.
;It makes them a bit easier to think about.

;First, lets define an area that JUST does the titlebar.
;[Normal]
;NumPoints=4
;PointList=0,1, 275,1, 275,14, 0,14

;Simple as can be :)
;Go ahead, uncomment it and have a look. Just remember to recomment everything
above it
;Doing your areas one at a time does speed the process up a bit

;Now lets define JUST the area sort of inside the outer ring
;[Normal]
;NumPoints = 4
;PointList = 3,15, 272,15, 272,113, 3,113

;Right, so say we want a mask that has, the titlebar AND the main area.
;We just add them together
; [Normal]
; NumPoints = 4, 4 ;Make sure to get the order right(although it doesn't matter
here =P )
; PointList = 0,1, 275,1, 275,14, 0,14,      3,15, 272,15, 272,113, 3,113

;Don't forget to add that extra comma after the first set.
; Having that bigger gap that lets you "keep it in your head" easier, it's a good
idea

;OK, now lets try something tricky. We are going to make a mask for just one green
line around the outside of the skin
;This introduces the other "big thing" with masks. Have a look at the first three
co-ordinates below. You can see the first two moves we make are to go one to the
side and then go down to the bottom. Looking at the last co-ordinate you can see we
will come back up at the end, making a line two wide, right?
;WRONG!! Here comes the big important phrase.

```

```

;IF YOU ARNT FOLLOWING THE OUTSIDE EDGE, YOUR CO-ORDINATES MARK TRANSPARENT SPACE
LIMITS
;That's right. Because I don't make a square, because I have to turn "inside" the
box, I am only marking space, and what I end up with is a mask with one green line.
;WARNING: Leave your skins dialog open when you do this one. :)
;[Normal]
;NumPoints = 8
;PointList = 1,14, 2,14, 2,114, 273,114, 273,14, 274,14, 274,115, 1,115

;OK, as the final touch lets add lets the border we just made to the other two.
;You should be able to work this out on your own
;[Normal]
;NumPoints = 4, 4, 8
;PointList = 0,1, 275,1, 275,14, 0,14, 3,15, 272,15, 272,113, 3,113, 1,14,
2,14, 2,114, 273,114, 273,14, 274,14, 274,115, 1,115

;hmm... that still looks a bit tacky down the bottom right
;So I'm going to modify it a bit
;See if you can work out what I've had to alter just by looking at the display of
it
;[Normal]
;NumPoints = 4, 4, 8
;PointList = 0,1, 275,1, 275,14, 0,14, 3,15, 272,15, 272,80, 3,80, 1,14,
2,14, 2,81, 273,81, 273,14, 274,14, 274,115, 1,115

; Justin's whacked trans skin :) in one, big lame messy line
; [Normal]
; NumPoints=20,4,4,4,8,4,4,6,6,4,4,4,4
;
PointList=0,0,19,0,19,11,114,11,114,0,156,0,156,11,243,11,243,0,275,0,275,13,266,13
,266,22,264,22,264,13,111,13,111,22,109,22,109,13,0,13,
109,22,266,22,266,36,109,36, 16,88,130,88,130,105,16,105,
136,89,157,89,157,104,136,104,
22,13,22,62,102,62,102,13,100,13,100,24,24,24,24,13, 0,0,275,0,275,3,0,3,
16,72,264,72,264,81,16,81, 0,13,0,78,16,78,16,75,4,75,4,13,
275,13,275,78,264,78,264,75,272,75,272,13, 14,78,16,78,16,105,14,105,
130,81,132,81,132,105,130,105, 146,81,146,89,148,89,148,81,
130,96,136,96,136,98,130,98

;THE END
;
;Post-Script
;The other good thing about doing your mask in bits and pieces like this is that
can make multiple versions so you can change as your whims change
You could also read our small format description below, in the transparency section.
Modern skimmers prefer to use various applications to create this file. One of them is already noticed
winamp region.txt generator by Maxim. As for me, it's the best program of that kind, as it allows you to
create transparency regions fast and easy, unlike using the rest of the programs or creating them
manually. (SacRat)
I (Imagine) created a region.txt file manually (for my FuturA skin) and I say there was not much fun
;) What I did - I opened the image in good old MS-Paint, zoomed it in, and typed cursor coordinates
which are displayed in the status bar, even if the window is inactive. But I wouldn't recommend you to
do the same! (Imagine)

```

Viscolor.txt

Visualization colors. When you don't use any visualization plugin, simple oscilloscope/spectrum analyzer is usually turned on and it's located on the left part of your main winamp window (below the numbers, which display song's time). The way it looks is described in this file.

It might look somewhat like that:

```
223,223,223, // color 0 = light gray
```

```

234,234,234, // color 1 = lighter gray for dots
210,210,210, // color 2 = top of spec
205,205,205, // 3
200,200,200, // 4
195,195,195, // 5
190,190,190, // 6
185,185,185, // 7
180,180,180, // 8
175,175,175, // 9
170,170,170, // 10
165,165,165, // 11
160,160,160, // 12
155,155,155, // 13
150,150,150, // 14
145,145,145, // 15
140,140,140, // 16
135,135,135, // 17 = bottom of spec
210,210,210, // 18 = osc 1
174,174,174, // 19 = osc 2 (slightly dimmer)
138,138,138, // 20 = osc 3
102,102,102, // 21 = osc 4
66,66,66, // 22 = osc 5
200,200,200, // 23 = analyzer peak dots

```

Some comments: the file has 24 lines with color values in RGB (decimal). All the text from double-slash “//” to the end of the line is a comment.

- The first line of the file has a background color of the visualizer. It’s often taken the same, as the average background’s color of main.bmp on that place.
- Line two has the second background color. It’s made to give your visualizer grid-looking style. If you don’t need any grid, just set it the same value, as the previous line.
- Lines 3-18 are spectrum analyzer’s color values (from top to the bottom). Trick: you could make every second, for example, line black (or give it any other indexed value) to make the analyzer look like it was build from small blocks.
- Lines 19-23 are the oscilloscope colors (from inside to outside).
- Line 24 is storing analyzer peaks’ color value.

Creating this file manually is not too hard, but you may use any utilities for this (look at *skinning applications (configs)* paragraph). (*SacRat*)

Hint: you can create a 16x1 px. file in your graphix program and fill it with any gradient, or draw anything you like in this limited space :) If you use Paint Shop Pro, choose ‘color picker’ tool shows the RGB value of the pixel under the cursor in a ‘color’ box (on right) even when inactive, so you can type it in your text editor. In Photoshop this will be a bit less convenient, still nothing too hard. You could also use any *color picker*, like Colorpad for this aim (look at *skinning applications (graphics)*). (*Imagine*)

Winampmb.txt

I file I hate so much... A file with content, which is displayed in winamp minibrowser by default. Theoretically. In fact some crappy skins use it to display a minibrowser window, if it’s inactive. May look somewhat like that. Hope, you’ll understand the idea (taken from the *template amp* skin):

```

<html>
<head>
<title>mbboxster</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<script language="JavaScript">
<!--
function MM_reloadPage(init) { //reloads the window if Nav4 resized
  if (init==true) with (navigator) {if
((appName=="Netscape")&&(parseInt(appVersion)==4)) {
    document.MM_pgW=innerWidth;    document.MM_pgH=innerHeight;
onresize=MM_reloadPage; }}

```

```

else if (innerWidth!=document.MM_pgW || innerHeight!=document.MM_pgH)
location.reload();
}
MM_reloadPage(true);
// -->
</script>
</head>

<body bgcolor="#FFFFFF" text="#A0A0A0" leftmargin="0" topmargin="0" marginwidth="0"
marginheight="0" scroll="no">
<div id="Layer1" style="position:absolute; left:0; top:0; width:255; height:88; z-
index:1"></div>
<div id="Layer2" style="position:absolute; left:1px; top:79px; width:244px;
height:74px; z-index:2">
  <p align="right"><font face="Arial, Helvetica, sans-serif"><b><font size="-
2">Version
  1.0<br>
  (c) 2001 Mischa Klement</font></b></font><br><br>
  <font face="Arial, Helvetica, sans-serif" size="-2">created with Macromedia
Fireworks 4.0<br>
  MikroAmp skinned<br>
  DFX skinned<br>
  <br>
  enjoy<br>
  cyana</font></p>
  <p align="right"><font face="Arial, Helvetica, sans-serif" size="-
2"><a href="http://www.omnislab.com"><font
color="#A0A0A0">www.omnislab.com</font></a></font></p>
  <p align="right"><font face="Arial, Helvetica, sans-serif" size="-2"> <br>
  <br>
  </font></p>
</div>
</body>
</html>

```

If you think, that you're a good in HTML, you could write your own, but I never do that. (*SacRat*)

Mb.ini

One more file, useless for almost any normal skinner. It defines the default page, which would be opened in winamp MB window.

The format is very simple:

```

[Winamp]
MBOpen=URL;

```

where "URL" is the address of your page to be opened.

TRANSPARENCY

Well, it's not a secret, that winamp 2 skin format supports transparency which abilities are nevertheless very limited (well, guys, the long-awaited winamp 3 will be able to make really transparent skins with all the opacity effects). How does it work? It's useful to know to know some limitations of this "trick" and it's disadvantages.

So-called "transparency" in winamp is just a way to make some regions of your skin "transparent" (interesting phrase), that means you could see through them and they wouldn't have any control elements. Note, that not all the windows of your skin can be "transparent", only the main window and equalizer. This creates some difficulties... For example, you'll have to make your "transparent" skin look OK with all the windows on and it's not a trivial task. A good idea is to leave the most part of your skin visible, so it would be easier for anyone to use it, so don't create too large transparent regions. Next, think a bit about a simple questions: what the hell do you need this transparency? It's often used to underline the shape of a skin, but not an end in itself. I've seen lots of interesting skins, whose authors

were trying to make their work look original by just making almost all the skin transparent and leaving few space for control elements... They were looking nice, but had terrible usability.

Creating a transparent skin manually is rather tricky: you create a new file, called *region.txt* in your skin's directory and then edit it to look like that:

```
[Normal]
NumPoints=4,4,4
PointList=0,0 275,0 275,113 0,113 0,113 48,113 48,116 0,116 226,113 275,113
275,116 226,116

[Equalizer]
NumPoints=4,4,4
PointList=0,0 48,0 48,116 0,116 226,0 275,0 275,116 226,116 48,4 275,4 275,113
48,113

[WindowShade]
NumPoints=8
PointList=3,0 0,3 0,11 3,14 272,14 275,11 275,3 272,0

[EqualizerWS]
NumPoints=8
PointList=3,0 0,3 0,11 3,14 272,14 275,11 275,3 272,0
```

(This fragment is taken from my Bend v3 skin, windowshade's transparency is made specially for this tutorial :)). Looks difficult? It's not that hard, as it seems for the first time... A [Normal] section has transparency description for the main window and [Equalizer] – for the equalizer. [WindowShade] and [EqualizerWS] are just transparency regions for the window's winshade modes (main and EQ). NumPoints=4,4,4 indicates, that There are 3 transparent regions in our skin's EQ and main window. Each of these regions consists of 4 points. For example, for the main window two first transparent areas look like: 0,0 275,0, 275,113 0,113 0,113. Fill this field *last*, when you know a complete number of your points and transparent regions in the skin. PointList= is an arrow of point coordinates in X,Y format. Note, that they are the *outside* pixels of your skin, so don't point skin elements here. Please note, that all the three regions are not divided by anything, so it's quite easy to make a mistake filling them manually. The same is right for the EQ...

It's not the best idea ever to make this file manually, as it takes a lot of time when making really original shapes. So you could use one of the programs for generating this file. I've seen several programs, which were designed to help you in this, but my favorite is Maxim's *Winamp regions.txt generator*, which could be found at <http://winamp.mwos.cjb.net/> (try looking through any search engine, if this link won't work). If you need a better explanation about creating editing this file, take a look at *regions.txt* section.

Well, there's one more «trick» to create half-transparent skins. Sometimes people just delete some skin's pixels as on a chessboard, getting a visibility of «transparency» this way. I personally dislike this perversion, which only make skins look worse and wouldn't recommend anyone use it. Just know, that it exists :) (*SacRat, Imagine*)

CURSORS

If you need it, you can also replace default winamp cursors.

Cursors usually have 32x32-pixel size and could be one of two types: animated and not animated. If you need to use animated cursors, just rename an ANI file into CUR and use it. Winamp will “understand” it as well.

Here's the description of every file:

CLOSE.CUR	<i>Main window's close</i>
EQCLOSE.CUR	<i>EQ close</i>
EQNORMAL.CUR	<i>EQ normal (default) cursor</i>
EQSLID.CUR	<i>EQ vertical sliders</i>
EQTITLE.CUR	<i>EQ titlebar</i>

MAINMENU . CUR	Main window's top-left button (menu)
MIN . CUR	Main window's MIN button
NORMAL . CUR	Main window's default (normal) cursor
PCLOSE . CUR	PL close
PNORMAL . CUR	PL normal cursor
POSBAR . CUR	Main window's posbar
PSIZE . CUR	PL resizable corner
PTBAR . CUR	PL top bar
PVSCROLL . CUR	PL vertical slider
PWINBUT . CUR	PL WS mode
PWSNORM . CUR	PL WS normal cursor
PWSSIZE . CUR	PL WS MAX button
SONGNAME . CUR	Main window's songname scroller
TITLEBAR . CUR	Main window's titlebar
VOLBAL . CUR	Main window's volume balance
VOLBAR . CUR	Main window's volume switch
WINBUT . CUR	Main window's WS button
WSCLOSE . CUR	WS close
WSMIN . CUR	Main window's WS MIN
WSNORMAL . CUR	WS normal cursor
WSPOSBAR . CUR	WS posbar
WSWINBUT . CUR	WS MAX

In order to change these cursors you could create your own files with any program, like **Axialis AX Cursors** (<http://www.axialis.com>) or use cursors, made by someone else (read *rips and copyrights* paragraph first), which could be found anywhere in Internet. (*SacRat*)

WINAMP SKIN SPECIFICATION (© BY FYRE)

Well, buddies, here's one more needful thing. Skin format description. Above you've read about the components of any skin and now you may find detailed information about any file. This specification was taken from Fyre's homepage (take a look a *links* section) and used by his authority. Why not anyone's else? Just because it's **the best** format description you may find on the Net! Even the official Winamp specification is much worse. Thanks, Fyre!

Overview

So what exactly is Winamp Skin Specs? It is a compilation of bitmaps and documents to help Winamp skimmers create bug-free skins. To novice skimmers, this is a useful tool in creating their your Winamp skin. It provides you with a blueprint to work with so that you are not starting empty handed. To advanced skimmers, this can be a reference to check to see whether your skin has any blatant bugs.

This version of the specs are arranged in HTML format for those who like to see pictures. A simple-text version can be downloaded also at <http://fyreskins.tripod.com/>.

First of all, a short description on how Winamp skins work. Each Winamp skin is actually a group of bitmaps and text files. There are 16 bitmaps and 3 text files (older versions of Winamp may differ - but why not download the latest version for free at Winamp.com?). These 19 files are bunched up in a zip file for easy downloading. There is no programming involved, just graphics. All you have to do is to draw the skin as you would want it to look like, and place paste them as separate pieces into the correct bitmaps and correct positions. It's that easy!

This version of the specs is arranged so that each web page you see describes one and only one bitmap or text file. Below are the links to each page.

Getting Started

Each page will feature a picture of the entire bitmap at the top. Then there will be a table of the different parts of that bitmap. The table will include information such as location, size, and a description. The location and size entries were obtained using the selection tool in MS Paint. Some programs may have a different selection system, thus you may have to adjust. However, most programs like Adobe Photoshop will follow the selection system I have used.

A very important note: the pictures are color coded for your convenience! Notice there are four colors. Green, white, red and grey. Green signifies a border. That means Winamp differentiates between one region and another at those green coordinates. White and red mean drawable (is that a word?) space. That means you can draw freely on the white and red regions. The red is simply for decoration or that you can tell which button is which. The grey regions are the ignored regions. Winamp does not use (display) any of the grey regions whatsoever. This space can be used for things like your personal logo or credits. Just make sure any doodles you do don't fall beyond the grey region.

Bitmaps

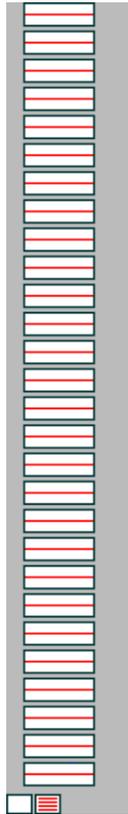
avs.bmp (Advanced Visualization Studio)



Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	avs.bmp	(0, 0)	9 x 9	close button pressed
	avs.bmp	(15, 0)	50 x 15	upper-left corner
	avs.bmp	(66, 15)	14 x 15	top filler
	avs.bmp	(81, 0)	16 x 15	upper-right corner
	avs.bmp	(0, 16)	7 x 172	left filler

	avs.bmp	(8, 16)	6 x 172	right filler
	avs.bmp	(15, 16)	50 x 5	lower-left corner
	avs.bmp	(66, 16)	14 x 5	bottom filler
	avs.bmp	(81, 16)	16 x 5	lower-right corner

balance.bmp (Left-Right Balance Adjuster)



Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	balance.bmp	(9, 0)	38 x 13	centered
	balance.bmp	(9, 15)	38 x 13	filler
	balance.bmp	(9, 30)	38 x 13	filler
	balance.bmp	(9, 45)	38 x 13	filler
	balance.bmp	(9, 60)	38 x 13	filler
	balance.bmp	(9, 75)	38 x 13	filler

	balance.bmp	(9, 90)	38 x 13	24% (left or right)
	balance.bmp	(9, 105)	38 x 13	filler
	balance.bmp	(9, 120)	38 x 13	33% (left or right)
	balance.bmp	(9, 135)	38 x 13	filler
	balance.bmp	(9, 150)	38 x 13	filler
	balance.bmp	(9, 165)	38 x 13	40% (left or right)
	balance.bmp	(9, 180)	38 x 13	filler
	balance.bmp	(9, 195)	38 x 13	49% (left or right)
	balance.bmp	(9, 210)	38 x 13	filler
	balance.bmp	(9, 225)	38 x 13	58% (left or right)
	balance.bmp	(9, 240)	38 x 13	filler
	balance.bmp	(9, 255)	38 x 13	66% (left or right)
	balance.bmp	(9, 270)	38 x 13	filler
	balance.bmp	(9, 285)	38 x 13	filler
	balance.bmp	(9, 300)	38 x 13	74% (left or right)
	balance.bmp	(9, 315)	38 x 13	filler
	balance.bmp	(9, 330)	38 x 13	82% (left or right)
	balance.bmp	(9, 345)	38 x 13	filler
	balance.bmp	(9, 360)	38 x 13	91% (left or right)
	balance.bmp	(9, 375)	38 x 13	filler
	balance.bmp	(9, 390)	38 x 13	filler
	balance.bmp	(9, 405)	38 x 13	100% (left or right)
	balance.bmp	(0, 422)	14 x 11	balance slider pressed
	balance.bmp	(15, 422)	14 x 11	balance slider unpressed

cbuttons.bmp (Control Buttons)



Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	cbuttons.bmp	(0, 0)	23 x 18	previous button unpressed
	cbuttons.bmp	(23, 0)	23 x 18	play button

				unpressed
	cbuttons.bmp	(46, 0)	23 x 18	pause button unpressed
	cbuttons.bmp	(69, 0)	23 x 18	stop button unpressed
	cbuttons.bmp	(92, 0)	22 x 18	next button unpressed
	cbuttons.bmp	(114, 0)	22 x 16	open button unpressed
	cbuttons.bmp	(0, 18)	23 x 18	prev button pressed
	cbuttons.bmp	(23, 18)	23 x 18	play button pressed
	cbuttons.bmp	(46, 18)	23 x 18	pause button pressed
	cbuttons.bmp	(69, 18)	23 x 18	stop button pressed
	cbuttons.bmp	(92, 18)	22 x 18	next button pressed
	cbuttons.bmp	(114, 18)	22 x 16	open button pressed

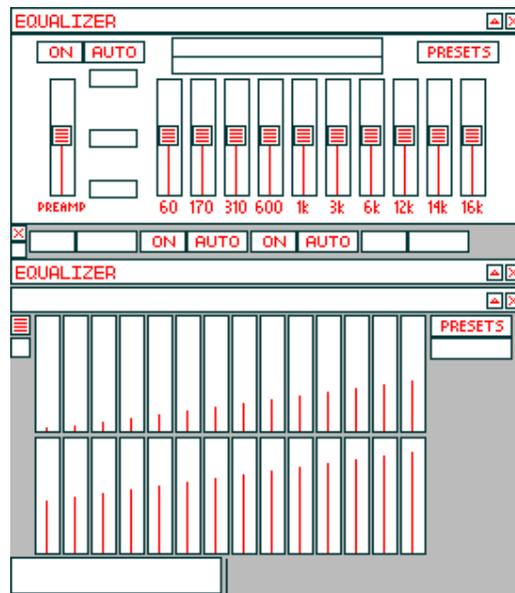
eq_ex.bmp (Equalizer Windowshade)



Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	eq_ex.bmp	(0, 0)	275 x 14	equalizer windowshade active
	eq_ex.bmp	(61, 4)	97 x 7	volume control active
	eq_ex.bmp	(164, 4)	42 x 7	balance control active
	eq_ex.bmp	(0, 15)	275 x 14	equalizer windowshade inactive
	eq_ex.bmp	(61, 19)	97 x 7	volume control inactive
	eq_ex.bmp	(164, 19)	42 x 7	balance control inactive
	eq_ex.bmp	(1, 30)	3 x 7	volume slider low
	eq_ex.bmp	(4, 30)	3 x 7	volume slider medium
	eq_ex.bmp	(7, 30)	3 x 7	volume slider

				high
	eq_ex.bmp	(11, 30)	3 x 7	balance slider left
	eq_ex.bmp	(14, 30)	3 x 7	balance slider middle
	eq_ex.bmp	(17, 30)	3 x 7	balance slider right
	eqmain.bmp	(1, 38)	9 x 9	windowshade button pressed
	eq_ex.bmp	(11, 38)	9 x 9	close button unpressed
	eq_ex.bmp	(1, 47)	9 x 9	maximize button pressed
	eq_ex.bmp	(11, 47)	9 x 9	close button pressed

eqmain.bmp (Equalizer Module)



Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	eqmain.bmp	(0, 0)	275 x 14	equalizer titlebar
	eqmain.bmp	(0, 14)	275 x 102	equalizer main display
	eqmain.bmp	(86, 16)	113 x 20	equalizer minidisplay
	eqmain.bmp	(14, 18)	25 x 12	equalizer on
	eqmain.bmp	(39, 18)	33 x 12	automatic

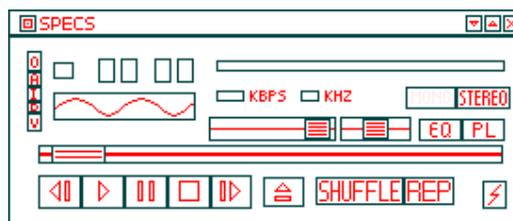
				equalization
	eqmain.bmp	(217, 18)	44 x 12	open preset menu
	eqmain.bmp	(42, 33)	26 x 10	20 db reset
	eqmain.bmp	(21, 38)	14 x 63	preamp bar
	eqmain.bmp	(78, 38)	14 x 63	60 hz bar
	eqmain.bmp	(96, 38)	14 x 63	170 hz bar
	eqmain.bmp	(114, 38)	14 x 63	310 hz bar
	eqmain.bmp	(132, 38)	14 x 63	600 hz bar
	eqmain.bmp	(150, 38)	14 x 63	1 khz bar
	eqmain.bmp	(168, 38)	14 x 63	3 khz bar
	eqmain.bmp	(186, 38)	14 x 63	6 khz bar
	eqmain.bmp	(204, 38)	14 x 63	12 khz bar
	eqmain.bmp	(222, 38)	14 x 63	14 khz bar
	eqmain.bmp	(240, 38)	14 x 63	16 khz bar
	eqmain.bmp	(42, 65)	26 x 10	0 db reset
	eqmain.bmp	(42, 92)	26 x 10	-20 db reset
	eqmain.bmp	(0, 116)	9 x 9	close button unpressed
	eqmain.bmp	(10, 119)	25 x 12	equalizer off unpressed

	eqmain.bmp	(35, 119)	33 x 12	auto off unpressed
	eqmain.bmp	(69, 119)	25 x 12	equalizer on unpressed
	eqmain.bmp	(94, 119)	33 x 12	auto on unpressed
	eqmain.bmp	(128, 119)	25 x 12	equalizer off pressed
	eqmain.bmp	(153, 119)	33 x 12	auto off pressed
	eqmain.bmp	(187, 119)	25 x 12	equalizer on pressed
	eqmain.bmp	(212, 119)	33 x 12	auto on pressed
	eqmain.bmp	(0, 125)	9 x 9	close button pressed
	eqmain.bmp	(0, 134)	275 x 14	equalizer titlebar active
	eqmain.bmp	(0, 149)	275 x 14	equalizer titlebar inactive
	eqmain.bmp	(0, 164)	11 x 11	equalizer slider unpressed
	eqmain.bmp	(13, 164)	14 x 63	-20 db bar
	eqmain.bmp	(28, 164)	14 x 63	-18.57 db bar
	eqmain.bmp	(43, 164)	14 x 63	-17.14 db bar
	eqmain.bmp	(58, 164)	14 x 63	-15.71 db bar
	eqmain.bmp	(73, 164)	14 x 63	-14.29 db bar
	eqmain.bmp	(88, 164)	14 x 63	-12.86 db bar
	eqmain.bmp	(103, 164)	14 x 63	-11.43 db bar

	eqmain.bmp	(118, 164)	14 x 63	-10 db bar
	eqmain.bmp	(133, 164)	14 x 63	-8.57 db bar
	eqmain.bmp	(148, 164)	14 x 63	-7.14 db bar
	eqmain.bmp	(163, 164)	14 x 63	-5.71 db bar
	eqmain.bmp	(178, 164)	14 x 63	-4.28 db bar
	eqmain.bmp	(193, 164)	14 x 63	-2.85 db bar
	eqmain.bmp	(208, 164)	14 x 63	-1.42 db bar
	eqmain.bmp	(224, 164)	44 x 12	preset button unpressed
	eqmain.bmp	(0, 176)	11 x 11	equalizer slider pressed
	eqmain.bmp	(224, 176)	44 x 12	preset button pressed
	eqmain.bmp	(13, 229)	14 x 63	1.42 db bar
	eqmain.bmp	(28, 229)	14 x 63	2.85 db bar
	eqmain.bmp	(43, 229)	14 x 63	4.28 db bar
	eqmain.bmp	(58, 229)	14 x 63	5.71 db bar
	eqmain.bmp	(73, 229)	14 x 63	7.14 db bar

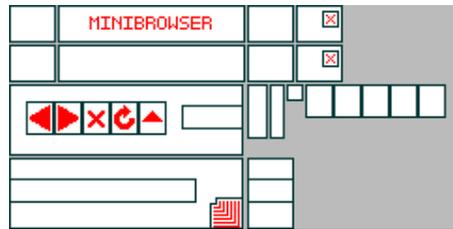
	eqmain.bmp	(88, 229)	14 x 63	8.57 db bar
	eqmain.bmp	(103, 229)	14 x 63	10 db bar
	eqmain.bmp	(118, 229)	14 x 63	11.43 db bar
	eqmain.bmp	(133, 229)	14 x 63	12.86 db bar
	eqmain.bmp	(148, 229)	14 x 63	14.29 db bar
	eqmain.bmp	(163, 229)	14 x 63	15.71 db bar
	eqmain.bmp	(178, 229)	14 x 63	17.14 db bar
	eqmain.bmp	(193, 229)	14 x 63	18.57 db bar
	eqmain.bmp	(208, 229)	14 x 63	20 db bar
	eqmain.bmp	(0, 293)	113 x 20	actual equalizer minidisplay
	eqmain.bmp	(115, 294)	1 x 19	spline line
	eqmain.bmp	(0, 314)	113 x 1	preamp line

main.bmp (Main Module)



Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
--------	---------------------	----------------------	-----------	-------------

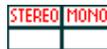
	titlebar.bmp	(0, 0)	275 x 14	titlebar
	main.bmp	(0, 14)	275 x 102	main display
	titlebar.bmp	(10, 22)	8 x 43	clutterbar
	numbers.bmp	[1] (48, 26) [2] (60, 26) [3] (78, 26) [4] (90, 26)	9 x 13	time display
	main.bmp	(111, 27)	154 x 6	song info display
	playpaus.bmp	(24, 18)	11 x 9	play-pause indicator
	monoster.bmp	(212, 41)	27 x 12	mono indicator
	monoster.bmp	(239, 41)	29 x 12	stereo indicator
	main.bmp	(111, 43)	15 x 6	kbps display
	main.bmp	(156, 43)	10 x 6	khz display
	main.bmp	(24, 44)	76 x 15	visualization window
	volume.bmp	(107, 57)	68 x 13	volume adjuster
	balance.bmp	(177, 57)	38 x 13	balance adjuster
	shufrep.bmp	(219, 58)	23 x 12	equalizer toggle
	shufrep.bmp	(242, 58)	23 x 12	playlist toggle
	posbar.bmp	(16, 72)	248 x 10	song progress bar
	cbuttons.bmp	(16, 88)	23 x 18	previous song
	cbuttons.bmp	(39, 88)	23 x 18	play song
	cbuttons.bmp	(62, 88)	23 x 18	pause song
	cbuttons.bmp	(85, 88)	23 x 18	stop song
	cbuttons.bmp	(108, 88)	22 x 18	next song
	cbuttons.bmp	(136, 89)	22 x 16	open song
	shufrep.bmp	(164, 89)	47 x 15	shuffle songs (randomize)
	shufrep.bmp	(211, 89)	27 x 15	repeat songs
	main.bmp	(253, 91)	13 x 15	Winamp info

mb.bmp (Minibrowser)

Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	mb.bmp	(0, 0)	25 x 20	upper-left cornerpiece active
	mb.bmp	(26, 0)	100 x 20	minibrowser titlebar active
	mb.bmp	(127, 0)	25 x 20	top fillers active
	mb.bmp	(153, 0)	25 x 20	upper-right cornerpiece active
	mb.bmp	(0, 21)	25 x 20	upper-left cornerpiece inactive
	mb.bmp	(26, 21)	100 x 20	minibrowser titlebar inactive
	mb.bmp	(127, 21)	25 x 20	top fillers inactive
	mb.bmp	(153, 21)	25 x 20	upper-right cornerpiece inactive
	mb.bmp	(0, 42)	125 x 38	bottom-left control bar
	mb.bmp	(127, 42)	11 x 29	left-side fillers
	mb.bmp	(139, 42)	8 x 29	right-side fillers
	mb.bmp	(148, 42)	9 x 9	close button pressed
	mb.bmp	(158, 42)	15 x 18	back button pressed
	mb.bmp	(173, 42)	15 x 18	forward button pressed
	mb.bmp	(188, 42)	15 x 18	stop button pressed
	mb.bmp	(203, 42)	15 x 18	reload button pressed
	mb.bmp	(218, 42)	15 x 18	options button

				pressed
	mb.bmp	(9, 51)	15 x 18	back button unpressed
	mb.bmp	(24, 51)	15 x 18	forward button unpressed
	mb.bmp	(39, 51)	15 x 18	stop button unpressed
	mb.bmp	(54, 51)	15 x 18	reload button unpressed
	mb.bmp	(69, 51)	15 x 18	options button unpressed
	mb.bmp	(92, 53)	33 x 13	URL display bar (left end)
	volume.bmp	(0, 81)	125 x 38	bottom-right control bar
	mb.bmp	(127, 81)	25 x 38	bottom filler
	mb.bmp	(0, 92)	100 x 13	URL display bar (right end)
	mb.bmp	(127, 92)	68 x 13	URL display bar (filler)
	mb.bmp	(107, 102)	18 x 17	resize control

monoster.bmp (Mono-Stereo Indicator)



Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	monoster.bmp	(0, 0)	29 x 12	stereo active
	monoster.bmp	(29, 0)	27 x 12	mono active
	monoster.bmp	(0, 12)	29 x 12	stereo inactive
	monoster.bmp	(29, 12)	27 x 12	mono inactive

numbers.bmp (Time Display Numbers)



Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	numbers.bmp	(0, 0)	9 x 13	0
	numbers.bmp	(9, 0)	9 x 13	1
	numbers.bmp	(18, 0)	9 x 13	2

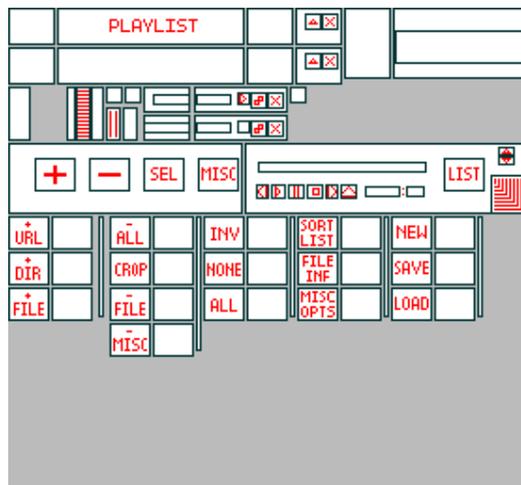
	numbers.bmp	(27, 0)	9 x 13	3
	numbers.bmp	(36, 0)	9 x 13	4
	numbers.bmp	(45, 0)	9 x 13	5
	numbers.bmp	(54, 9)	9 x 13	6
	numbers.bmp	(63, 9)	9 x 13	7
	numbers.bmp	(72, 9)	9 x 13	8
	numbers.bmp	(81, 15)	9 x 13	9
	numbers.bmp	(90, 18)	9 x 13	blank

playpaus.bmp (Play-Pause Indicator)

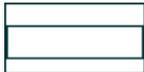


Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	playpaus.bmp	(1, 0)	8 x 9	play indicator
	playpaus.bmp	(9, 0)	9 x 9	pause indicator
	playpaus.bmp	(18, 0)	9 x 9	stop indicator
	playpaus.bmp	(27, 0)	2 x 9	filler
	playpaus.bmp	(36, 0)	3 x 9	start of song indicator
	playpaus.bmp	(39, 0)	3 x 9	end of song indicator

pledit.bmp (Playlist Editor Module)



Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	pledit.bmp	(0, 0)	25 x 20	upper-left cornerpiece active
	pledit.bmp	(26, 0)	100 x 20	playlist titlebar

				active
	pledit.bmp	(127, 0)	25 x 20	top fillers active
	pledit.bmp	(153, 0)	25 x 20	upper-right corner piece active
	pledit.bmp	(179, 0)	25 x 38	bottom fillers
	pledit.bmp	(205, 0)	75 x 38	visualization miniscreen
	pledit.bmp	(0, 21)	25 x 20	upper-left cornerpiece inactive
	pledit.bmp	(26, 21)	100 x 20	playlist titlebar inactive
	pledit.bmp	(127, 21)	25 x 20	top fillers inactive
	pledit.bmp	(153, 21)	25 x 20	upper-right cornerpiece inactive
	pledit.bmp	(0, 42)	12 x 29	left-side fillers
	pledit.bmp	(31, 42)	5 x 29	right-side fillers (left bar)
	pledit.bmp	(36, 42)	8 x 29	right-side fillers (scroll groove)
	pledit.bmp	(44, 42)	7 x 29	right-side fillers (right bar)
	pledit.bmp	(52, 42)	9 x 9	close button pressed
	pledit.bmp	(62, 42)	9 x 9	windowshade button pressed
	pledit.bmp	(72, 42)	25 x 14	windowshade (left end)
	pledit.bmp	(99, 42)	50 x 14	windowshade (right end) active
	pledit.bmp	(150, 42)	9 x 9	maximize button pressed
	pledit.bmp	(52, 53)	8 x 18	scroll button unpressed
	pledit.bmp	(62, 53)	8 x 18	scroll button pressed
	pledit.bmp	(72, 57)	25 x 14	windowshade (middle fillers)
	pledit.bmp	(99, 57)	50 x 14	windowshade (right end) inactive
	pledit.bmp	(0, 72)	125 x 38	bottom-left control bar

	pledit.bmp	(126, 72)	150 x 38	bottom-right control bar
	pledit.bmp	(261, 74)	9 x 5	page up button
	pledit.bmp	(261, 79)	9 x 5	page down button
	pledit.bmp	(14, 80)	22 x 18	add button
	pledit.bmp	(43, 80)	22 x 18	remove button
	pledit.bmp	(72, 80)	22 x 18	selection button
	pledit.bmp	(101, 80)	22 x 18	miscellaneous button
	pledit.bmp	(232, 80)	22 x 18	list button
	pledit.bmp	(133, 82)	90 x 6	time status display
	pledit.bmp	(257, 89)	19 x 21	resize control
	pledit.bmp	(132, 94)	7 x 8	previous control
	pledit.bmp	(140, 94)	8 x 8	play control
	pledit.bmp	(149, 94)	9 x 8	pause control
	pledit.bmp	(159, 94)	9 x 8	stop control
	pledit.bmp	(169, 94)	7 x 8	next control
	pledit.bmp	(177, 94)	9 x 8	open control
	pledit.bmp	(190, 95)	19 x 6	current time display (minutes)
	pledit.bmp	(212, 95)	10 x 6	current time display (seconds)
	pledit.bmp	(0, 111)	22 x 18	add URL button unpressed
	pledit.bmp	(23, 111)	22 x 18	add URL button pressed
	pledit.bmp	(48, 111)	3 x 54	decoration bar (add)
	pledit.bmp	(54, 111)	22 x 18	remove all button unpressed
	pledit.bmp	(77, 111)	22 x 18	remove all button pressed
	pledit.bmp	(100, 111)	3 x 72	decoration bar (remove)
	pledit.bmp	(104, 111)	22 x 18	invert selection

				button unpressed
	pledit.bmp	(127, 111)	22 x 18	invert selection button pressed
	pledit.bmp	(150, 111)	3 x 54	decoration bar (selection)
	pledit.bmp	(154, 111)	22 x 18	sort list button unpressed
	pledit.bmp	(177, 111)	22 x 18	sort list button pressed
	pledit.bmp	(200, 111)	3 x 54	decoration bar (misc)
	pledit.bmp	(204, 111)	22 x 18	new list button unpressed
	pledit.bmp	(227, 111)	22 x 18	new list button pressed
	pledit.bmp	(250, 111)	3 x 54	decoration bar (list)
	pledit.bmp	(0, 130)	22 x 18	add DIR button unpressed
	pledit.bmp	(23, 130)	22 x 18	add DIR button pressed
	pledit.bmp	(54, 130)	22 x 18	crop button unpressed
	pledit.bmp	(77, 130)	22 x 18	crop button pressed
	pledit.bmp	(104, 130)	22 x 18	select none button unpressed
	pledit.bmp	(127, 130)	22 x 18	select none button pressed
	pledit.bmp	(154, 130)	22 x 18	file info button unpressed
	pledit.bmp	(177, 130)	22 x 18	file info button pressed
	pledit.bmp	(204, 130)	22 x 18	save list button unpressed
	pledit.bmp	(227, 130)	22 x 18	save list button pressed
	pledit.bmp	(0, 149)	22 x 18	add file button unpressed
	pledit.bmp	(23, 149)	22 x 18	add file button

				pressed
	pledit.bmp	(54, 149)	22 x 18	remove file button unpressed
	pledit.bmp	(77, 149)	22 x 18	remove file button pressed
	pledit.bmp	(104, 149)	22 x 18	select all button unpressed
	pledit.bmp	(127, 149)	22 x 18	select all button pressed
	pledit.bmp	(154, 149)	22 x 18	misc options button unpressed
	pledit.bmp	(177, 149)	22 x 18	misc options button pressed
	pledit.bmp	(204, 149)	22 x 18	load list button unpressed
	pledit.bmp	(227, 149)	22 x 18	load list button pressed
	pledit.bmp	(54, 168)	22 x 18	misc remove button unpressed
	pledit.bmp	(77, 168)	22 x 18	misc remove button pressed

posbar.bmp (Song Progress Bar)



Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	posbar.bmp	(0, 0)	248 x 10	song progress bar
	posbar.bmp	(248, 0)	29 x 10	song slider unpressed
	posbar.bmp	(278, 0)	29 x 10	song slider pressed

shufrep.bmp (Shuffle, Repeat, Equalizer, & Playlist Toggles)



Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	shufrep.bmp	(0, 0)	28 x 15	no repeat unpressed
	shufrep.bmp	(28, 0)	47 x 15	no shuffle unpressed
	shufrep.bmp	(0, 15)	28 x 15	no repeat pressed
	shufrep.bmp	(28, 15)	47 x 15	no shuffle pressed
	shufrep.bmp	(0, 30)	28 x 15	repeat unpressed
	shufrep.bmp	(28, 30)	47 x 15	shuffle unpressed
	shufrep.bmp	(0, 45)	28 x 15	repeat pressed
	shufrep.bmp	(28, 45)	47 x 15	shuffle pressed
	eqmain.bmp	(0, 61)	23 x 12	no equalizer unpressed
	pledit.bmp	(23, 61)	23 x 12	no playlist unpressed
	eqmain.bmp	(46, 61)	23 x 12	no equalizer pressed
	pledit.bmp	(69, 61)	23 x 12	no playlist pressed
	eqmain.bmp	(0, 73)	23 x 12	equalizer unpressed
	pledit.bmp	(23, 73)	23 x 12	playlist unpressed
	eqmain.bmp	(46, 73)	23 x 12	equalizer pressed
	pledit.bmp	(69, 73)	23 x 12	playlist pressed

text.bmp (Winamp Text)

ABCDEFGHIJKLMNOPQRSTUVWXYZ "0
 123456789... : @ - ! _ \ |] ^ & * + =
 ~ ` ~`

Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	text.bmp	(0, 0)	5 x 6	A
	text.bmp	(5, 0)	5 x 6	B
	text.bmp	(10, 0)	5 x 6	C
	text.bmp	(15, 0)	5 x 6	D
	text.bmp	(20, 0)	5 x 6	E
	text.bmp	(25, 0)	5 x 6	F
	text.bmp	(30, 0)	5 x 6	G
	text.bmp	(35, 0)	5 x 6	H
	text.bmp	(40, 0)	5 x 6	I
	text.bmp	(45, 0)	5 x 6	J

K	text.bmp	(50, 0)	5 x 6	K
L	text.bmp	(55, 0)	5 x 6	L
M	text.bmp	(60, 0)	5 x 6	M
N	text.bmp	(65, 0)	5 x 6	N
O	text.bmp	(70, 0)	5 x 6	O
P	text.bmp	(75, 0)	5 x 6	P
Q	text.bmp	(80, 0)	5 x 6	Q
R	text.bmp	(85, 0)	5 x 6	R
S	text.bmp	(90, 0)	5 x 6	S
T	text.bmp	(95, 0)	5 x 6	T
U	text.bmp	(100, 0)	5 x 6	U
V	text.bmp	(105, 0)	5 x 6	V
W	text.bmp	(110, 0)	5 x 6	W
X	text.bmp	(115, 0)	5 x 6	X
Y	text.bmp	(120, 0)	5 x 6	Y
Z	text.bmp	(125, 0)	5 x 6	Z
"	text.bmp	(130, 0)	5 x 6	quotation mark
@	text.bmp	(135, 0)	5 x 6	at symbol
	text.bmp	(142, 0)	5 x 6	time display background
	text.bmp	(150, 0)	5 x 6	title display space
0	text.bmp	(0, 6)	5 x 6	0
1	text.bmp	(5, 6)	5 x 6	1
2	text.bmp	(10, 6)	5 x 6	2
3	text.bmp	(15, 6)	5 x 6	3
4	text.bmp	(20, 6)	5 x 6	4
5	text.bmp	(25, 6)	5 x 6	5
6	text.bmp	(30, 6)	5 x 6	6
7	text.bmp	(35, 6)	5 x 6	7
8	text.bmp	(40, 6)	5 x 6	8
9	text.bmp	(45, 6)	5 x 6	9
...	text.bmp	(50, 6)	5 x 6	ellipsis ...
.	text.bmp	(55, 6)	5 x 6	period .
:	text.bmp	(60, 6)	5 x 6	colon :
(text.bmp	(65, 6)	5 x 6	left parenthesis (
)	text.bmp	(70, 6)	5 x 6	right parenthesis)
-	text.bmp	(75, 6)	5 x 6	dash -
'	text.bmp	(80, 6)	5 x 6	apostrophe '

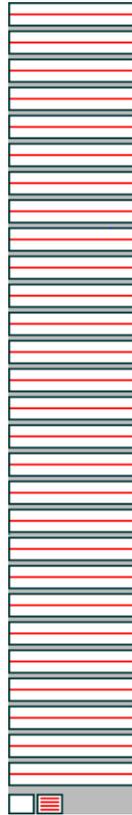
!	text.bmp	(85, 6)	5 x 6	exclamation mark !
-	text.bmp	(90, 6)	5 x 6	underscore _
+	text.bmp	(95, 6)	5 x 6	plus symbol +
\	text.bmp	(100, 6)	5 x 6	back slash \
/	text.bmp	(105, 6)	5 x 6	slash /
[text.bmp	(110, 6)	5 x 6	left bracket [
]	text.bmp	(115, 6)	5 x 6	right bracket]
^	text.bmp	(120, 6)	5 x 6	carrot ^
&	text.bmp	(125, 6)	5 x 6	ampersand &
%	text.bmp	(130, 6)	5 x 6	percent %
,	text.bmp	(135, 6)	5 x 6	comma ,
=	text.bmp	(140, 6)	5 x 6	equal symbol =
\$	text.bmp	(145, 6)	5 x 6	dollar symbol \$
#	text.bmp	(150, 6)	5 x 6	pound symbol #
â	text.bmp	(0, 12)	5 x 6	â
ö	text.bmp	(5, 12)	5 x 6	ö
ä	text.bmp	(10, 12)	5 x 6	ä
?	text.bmp	(15, 12)	5 x 6	question mark ?
*	text.bmp	(20, 12)	5 x 6	asterisk *
	text.bmp	(100, 12)	5 x 6	kbps display space

titlebar.bmp (Titlebar)



Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	titlebar.bmp	(0, 0)	9 x 9	menu button unpressed
	titlebar.bmp	(9, 0)	9 x 9	minimize button unpressed
	titlebar.bmp	(18, 0)	9 x 9	exit button unpressed
	titlebar.bmp	(27, 0)	275 x 14	normal titlebar active

	titlebar.bmp	(304, 0)	8 x 43	clutterbar shown
	titlebar.bmp	(312, 0)	8 x 43	clutterbar hidden
	titlebar.bmp	(0, 9)	9 x 9	menu button pressed
	titlebar.bmp	(9, 9)	9 x 9	minimize button pressed
	titlebar.bmp	(18, 9)	9 x 9	exit button pressed
	titlebar.bmp	(27, 15)	275 x 14	normal titlebar inactive
	titlebar.bmp	(0, 18)	9 x 9	windowshade button unpressed
	titlebar.bmp	(9, 18)	9 x 9	windowshade button pressed
	titlebar.bmp	(0, 27)	9 x 9	maximize button unpressed
	titlebar.bmp	(9, 27)	9 x 9	maximize button pressed
	titlebar.bmp	(27, 29)	275 x 14	windowshade titlebar active
	titlebar.bmp	(0, 36)	17 x 7	song progress bar
	titlebar.bmp	(17, 36)	9 x 7	song progress controls
	titlebar.bmp	(27, 42)	275 x 14	windowshade titlebar inactive
	titlebar.bmp	(304, 44)	8 x 43	clutterbar [o] pressed
	titlebar.bmp	(312, 44)	8 x 43	clutterbar [a] pressed
	titlebar.bmp	(320, 44)	8 x 43	clutterbar [i] pressed
	titlebar.bmp	(328, 44)	8 x 43	clutterbar [d] pressed
	titlebar.bmp	(336, 44)	8 x 43	clutterbar [v] pressed
	titlebar.bmp	(27, 57)	275 x 14	alternative titlebar active
	titlebar.bmp	(27, 72)	275 x 14	alternative titlebar

volume.bmp (Volume Adjuster)

Bitmap	Corresponds To File	Starting Coordinates	Dimension	Description
	volume.bmp	(0, 0)	68 x 13	0%-1%
	volume.bmp	(0, 15)	68 x 13	3%-5%
	volume.bmp	(0, 30)	68 x 13	7%-9%
	volume.bmp	(0, 45)	68 x 13	11%-13%
	volume.bmp	(0, 60)	68 x 13	15%-17%
	volume.bmp	(0, 75)	68 x 13	19%-21%
	volume.bmp	(0, 90)	68 x 13	23%-25%
	volume.bmp	(0, 105)	68 x 13	27%-29%
	volume.bmp	(0, 120)	68 x 13	31%
	volume.bmp	(0, 135)	68 x 13	33%-35%
	volume.bmp	(0, 150)	68 x 13	37%-39%
	volume.bmp	(9, 165)	68 x 13	41%-43%
	volume.bmp	(0, 180)	68 x 13	45%-47%
	volume.bmp	(0, 195)	68 x 13	49%-50%
	volume.bmp	(0, 210)	68 x 13	52%-54%
	volume.bmp	(0, 225)	68 x 13	56%-58%
	volume.bmp	(0, 240)	68 x 13	60%-62%
	volume.bmp	(0, 255)	68 x 13	64%
	volume.bmp	(0, 270)	68 x 13	66%-68%

	volume.bmp	(0, 285)	68 x 13	70%-72%
	volume.bmp	(0, 300)	68 x 13	74%-76%
	volume.bmp	(0, 315)	68 x 13	78%-80%
	volume.bmp	(0, 330)	68 x 13	82%-84%
	volume.bmp	(0, 345)	68 x 13	86%-88%
	volume.bmp	(0, 360)	68 x 13	90%-92%
	volume.bmp	(0, 375)	68 x 13	94%-96%
	volume.bmp	(0, 390)	68 x 13	98%
	volume.bmp	(0, 405)	68 x 13	100%
	volume.bmp	(0, 422)	14 x 11	volume slider pressed
	volume.bmp	(15, 422)	14 x 11	volume slider unpressed

Text Files

pledit.txt (Playlist and Minibrowser Text Settings)

```

pledit - Notepad
File Edit Search Help
[Text]
Normal=#FF0000
Current=#000000
NormalBG=#FFFFFF
SelectedBG=#0000FF
Font=Arial
mbBG=FFFFFF
mbFG=FF0000

```

Line	Corresponds To File	Description
Normal	pledit.bmp	normal (not playing) text color
Current	pledit.bmp	current (playing) text color
NormalBG	pledit.bmp	normal (not selected) background
SelectedBG	pledit.bmp	selected background
Font	pledit.bmp & mb.bmp	font name
mbBG	mb.bmp	minibrowser background color
mbFG	mb.bmp	minibrowser text color

NOTE: The colors in pledit.txt are expressed in hexadecimal form.

region.txt (Transparency Settings)

```

region - Notepad
File Edit Search Help
[Normal]
NumPoints=20
PointList=7,1,269,1,270

[WindowShade]
NumPoints=13
PointList=7,1,269,1,270

[Equalizer]
NumPoints=20,20
PointList=7,1,269,1,270

[EqualizerWS]
NumPoints=13
PointList=7,1,269,1,270

```

Line	Corresponds To File	Description
[Normal]	main.bmp	indicates the transparency is for the main module
[WindowShade]	titlebar.bmp	indicates the transparency is for the windowshade
[Equalizer]	eqmain.bmp	indicates the transparency is for the equalizer
[EqualizerWS]	eq_ex.bmp	indicates the transparency is for the equalizer windowshade
NumPoints	n/a	the number of points that will be defined
PointList	n/a	list of coordinate points

NOTE: The series of points indicate the area that will be seen, NOT the hidden area. Also, if NumPoints does not match the number of points in PointList, this will cause Winamp to have an error.

viscolor.txt (Visualization Color Settings)

```

viscolor - Notepad
File Edit Search Help
255,255,255 // 0
255,236,236 // 1
255,0,0 // 2
255,15,15 // 3
255,30,30 // 4
255,45,45 // 5
255,60,60 // 6
255,75,75 // 7
255,90,90 // 8
255,105,105 // 9
255,120,120 // 10
255,135,135 // 11
255,150,150 // 12
255,165,165 // 13
255,180,180 // 14
255,195,195 // 15
255,210,210 // 16
255,236,236 // 17
255,0,0 // 18
255,59,59 // 19
255,118,118 // 20
255,177,177 // 21
255,236,236 // 22

```

Line	Corresponds To File	Description
// 0	n/a	background color
// 1	n/a	background dots
// 2	n/a	top spectrum color
// 3	n/a	spectrum color 2
// 4	n/a	spectrum color 3
// 5	n/a	spectrum color 4
// 6	n/a	spectrum color 5
// 7	n/a	spectrum color 6
// 8	n/a	spectrum color 7
// 9	n/a	spectrum color 8
// 10	n/a	middle spectrum color
// 11	n/a	spectrum color 10
// 12	n/a	spectrum color 11
// 13	n/a	spectrum color 12

// 14	n/a	spectrum color 13
// 15	n/a	spectrum color 14
// 16	n/a	spectrum color 15
// 17	n/a	bottom spectrum color
// 18	n/a	brightest oscilloscope color
// 19	n/a	oscilloscope color 2
// 20	n/a	oscilloscope color 3
// 21	n/a	oscilloscope color 4
// 22	n/a	dimpest oscilloscope color

NOTE: The colors in viscolor.txt are expressed in RGB (red,green,blue) form.

Notes

There are many reasons I created the specs. One of them of for myself, of course. I found a need to be sure that the skin was accurate rather than count on another skin which may have many mistakes. Another one was that after tons of people saw my most popular skins, I started getting 10 e-mails a day asking me how I made the skins and if I could teach them how to start. The most important reason is that I have seen too many messed up skins out there that would be otherwise good if corrected. What a shame to make a great looking skin but have it full of bugs.

There are a few bugs that nobody can avoid making. In fact these bugs have nothing to do with the skinner but every to do with Winamp itself. Until the Winamp guys can correct Winamp's code for displaying the skin these bugs will be seen everywhere:

- **main windowshade** - the bottom pixels for the active windowshade are the same as the top pixels for the inactive windowshade.
- **main module time display** - in the main window, there is a designated space for a negative sign if you like to see a count down instead of a count up; Winamp incorrectly shifts the negative sign out of place.
- **playlist windowshade button** - when the playlist windowshade is on, try clicking the maximize button; once your mouse button is down, the button will magically move one pixel to the right.

Enough complaining from me now. If you have any complaints, comments, questions, even a "well done" message you wanna send me, my address is:

mt_101@yahoo.com

or...

fyreskins@yahoo.com

BUGZ (don't forget about testing...)

Few words about some bugs and misses, which could often be found even in skins, made by professionals. Before submitting, check these things:

- **Buttons displacements and animation bugs:** press any button and without releasing your mouse's left button (if you're not a left-hander) move cursor over all the buttons in your skin.

It's easy to find "lost" animation this way. Besides, if some buttons are displaced even one px aside, you will find the bug.

- **Headers:** sometimes you could make a mistake, while creating your headers in *titlebar.bmp* or so. For example, it (or any header's element, like title) could be displaced.
- **Sliders:** sliders are, possibly, the hardest things to troubleshoot, as they have lots of animations. Just move your slider slowly from minimum to maximum position and look at the slider carefully. If you've missed only a pixel in your *volume.bmp* (or any other slider), it could spoil all your work. Sliders trend to "climb down" very easily, so pay a lot of attention to them.
- **Stop!:** if you think, that a winamp, where your skin is applied is always playing music, you're not right: sometimes it's useful to press *pause* or *stop* and take a look at changes: posbar displacements, missing background in animation window, hell-knows-what on the place of your digits... this all is more, than real.
- **Resize:** resize your playlist, MB, AVS window and find out, that there are clear joints in tiling elements... or not... Anyway, it's better to find it out now.
- **WS:** for some reasons people usually forget, that someone could use it. One of the most "popular" bugs is a displacement of min/max/close buttons.
- **Is that all?** Not... Check all the modes of visulizer (spectrum/oscilloscope) and be sure, that you have tested *all* the windows, including AVS and MB (if you skinned them) in *all* the modes. Check readme file and start all over again (joke). Now that's all... (SacRat)

RIPS and COPIRIGHTS

There are some important things *any* winamp skinner must know. It doesn't matter is you're a member of any NetArt community, like deviantart or just a single, you better follow some basic rules... First of all, respect others works: don't rip them. Ripping is taking a work of any author and showing it as your own. It's not a good idea to do this: you will only get satisfaction from the high rating of the work if it's yours, besides in most of the communities ripping is "a heavy sin", so you will immediately be excluded from it (and will be marked as a ripper). As for me, taking even a part of the original work is ripping; I've seen lots of skins with completely the same buttons and so on... I think, they were looking terrible.

From another side, *"stealing" the idea is not ripping*: you might make your skin's elements look similar to the ones you've seen somewhere before and no one will blame you for that.

Another interesting thing is creating a *"remix"* or your own version of a work. For example, you have seen an interesting skin, but dislike it's color and equalizer, so you decided to change the color and make your own EQ. What shall you do? First of all contact the author. If you wouldn't do this, he could lodge a complaint on you and you could be blamed in ripping and this is not the thing you were going to get. Right? Just know, that most of the authors, whose works I've tried to remake agreed to give me a permission on creating my version of their skin if I note them in readme file. There are some situations, when you cannot find the author of the original work. For example, he hasn't noticed his name anywhere and he hasn't created any readme. Then try to look for a skin with that name (it's often a name of the archive, sometimes authors show their skin's name on the titlebar) on www.winampskins.com, www.1001winampskins.com or at any other more-less popular skin site. You may find a skin there and therefore know who's the author. If nothing helps (you haven't found him), just write something like "the work is based on orig_skin_name winamp skin" in your description after it's finished.

All above could be said by a very short phrase: respect others and they will respect you. (SacRat)

SKIN PROTECTION

Well, you've done a work and you don't want anyone to grab it. If you have submitted it under your name, it's an evidence of your authorship already (most of the sites, where you could submit your work, automatically set the skin's submission date). And... don't forget to include readme.txt file ;) (SacRat)

If you want to protect your skin from ripping (well, everyone wants it!) there's a good way to do it. The point is that most of the images of your skin have some 'free' space on them, so you can write there something like "skin by Imagine and SacRat" and "I've made this skin! Don't rip it!!!" There's a lot of free space in avs.bmp, mb.bmp, eqmain.bmp, eq_ex.bmp, plectit.bmp and titlebar.bmp. Most users don't unpack skins in search of your signature, but many rippers don't, too (who knows, who knows... nevertheless this method is used very often - a note by SacRat). So, if you see that your skin was ripped, you can easily prove your authorship. You can also draw your logo in the visualization window - there it will be seen. (*Imagine*)

Well, sooth to say there are even more tricky methods. As it was said, the easiest way is just to include a readme.txt or any other file with your signature into the work (you could also add a comment to wsz archive (don't forget, that it's just a zip file) or place your own signature file into it, but someone, who wants to grab your work might simply repack the archive and remove all the signatures this way. A method, which was suggested by Imagine, is a bit better, but these signs could also be removed in any editing program, even in MS Paint.

So, if you're a truly maniac and want something special, you could try one of the methods below. The first one is simple... There are programs, which allow you to add some redundant information to your bitmap files, but no changes will be visible. How it works: you create a file (text files are smaller) with your signature and then imbed it into any image of your skin in encrypted form. No visible changes will occur and the size of your images (and skin, for sure...), won't enlarge much (it's usually a couple of Kb), so if someone just take your images, you will be able to prove your authorship by extracting the hidden information. Disadvantages of this method are obvious: this embedded information can be lost after some manipulations with the file.

Another idea is to include the authorship information by using much more complicated programs, like Digimark's products, which include all the author's information directly into your image, so they will be readable even after a set of manipulations. Disadvantages: watermarks may "spoil" your image a bit, and as I know, these programs are not free at all.

Remember: whatever you do it's *impossible* to create 100% protection of you skin's files, but you could prove your authorship if you have submitted your skin first...

And at the end I'd recommended you to read one interesting article about skin protection and copyrights. It's located here: (<http://www.teknidermy.net/issue6/artcopyright.html>). (*SacRat*)

SUBMITTING

Here are some useful links of the sites, where you can place your own skins (don't forget, that you could also download skins from them):

-<http://www.deviantart.com/>

-<http://www.winamp.com/skins>

-<http://www.1001winampskins.com/>

-<http://www.skinz.org/>

-<http://www.lotsofskins.com/>

-<http://www.skinbase.org/>

While submitting a skin to most of these sites you should follow some rules: a work you submit must be original (not ripped) and created by you; it does not contain any nudity, violence and so on... (well, the rule about nudity is not always right, just take a look to celebs or female skin section); at least three windows must be skinned: main, equalizer, playlist. Some sites, like <http://www.winamp.com/> also recommend skinning minibrowser (as it's included into the preview thumbnail).

Some information: a submitting process may take *some time*: from few seconds on www.deviantart.com or one day www.skinz.org to several weeks or even more, as in www.winamp.com. (*SacRat*)

MAKING A PREVIEW

Most of the sites, where you submit your works need a preview picture. The only exclusions I know are www.winamp.com and www.1001winampskins.com, which generate this preview automatically (the second one generates a preview only once, later you will be allowed to do this manually).

What is it (a preview)? It's just a *shot* of your skin. Nothing less, nothing more. If you're not a dumb, you may make this shot in a second and if you are, then it might take much, much more time :). You could just press *Print Screen* button or use any special program. I prefer **Irfan view** (<http://www.irfanview.com>), as you could crop the skin from a shot of your desktop with no need to run any other program.

When you make a shot, remember, that it's the first thing a user sees and it influences on his wish to download your skin greatly: most people take a look at the picture first and only then decide if they need this skin, isn't it? ;) Make shots when winamp is in focus and it's playing music; *try to make your skins look the best on the shot*. You're not limited in the position of your skin's windows, but remember, that the most commonly used placement is (top to bottom): main, EQ, PL, so it's better to make preview when your windows are placed in this order. If you also place there an AVS or minibrowser window, then it doesn't really matter, that your desktop wallpaper is seen through it, just try to select a wallpaper looking according to your skin or make it in Photoshop (or any other bitmap graphical editor).

Make thumbnail's file size as small as possible, but try not to lose quality. Keep in mind, that JPG format is better for compressing photos and GIF is the best for pictures with a limited set of colors. Sometimes sites could accept PNG images, then use it instead of GIF (but don't forget to set maximum PNG compression first!). Don't submit a whole image of your desktop, even if it looks great with your skin. Remember, that the size of a winamp's window is quite small relatively to the desktop space, so it might be unseen on a small preview (in most cases a user won't see your preview in full size, only a small thumbnail). (SacRat)

COMPABILITY

It's not a secret, that now there are lots of winamp clones, working at almost any operating system and there are many programs, which also support winamp skin format. If you want to make a compatible skin, you should remember following (details may differ for any specific program): don't take a step aside from the classical winamp skin standard and don't use any transparency. The best solution is to create a transparent skin, which looks good even without regions.txt. What about classics: lots of programs of that kind use classical winamp skin specification, which does not allow to create some tricks, like cropping images here and there. (SacRat)

HINTS AND TRICKS

Small introduction

This is not a self-dependent paragraph. It's just a collection of hints and tricks, already described above. I bet, you, dude, was lazy enough to read the whole tutorial, so take a look at this section. If you're an experienced skinner, even then these hints might be useful for you.

Skin files

- *WSZ* file is a simple *ZIP* archive.
- Try to make your skin *as small, as possible*. It shouldn't be more, than 300Kb. An average skin in True Color is usually less, than 200Kb.
- Use *maximum* *ZIP* compression to make smaller skins.
- *Remove* all the useless files, like *.bak, thumbs.db, *.psd and so on from skin archive: the less useless stuff you add, the smaller is file.
- Most skins could be converted to 256 colors *without a visible loss of quality*. This could help you to reduce the size.

- A name of your file is a name of your skin in winamp skin browser. Note, that if this name starts from digit or any other non-alphabetical sign, it's placed *before* the rest.

Images

- Many images (volume, balance, EQ) in your skin could be cropped in order to use some special effects.
- All the images are positioned relatively to their left top corner. Don't forget it, when cropping: only delete bottom and right regions.
- Use *button extractor* to ease your work in creating skins' backgrounds.
- Don't forget, that some buttons, like *shuffle*, *repeat*, *PL/EQ* and so on have different intermediate (pressed) states for each direction: on→off and off←on.
- Unlike in most other windows, AVS stripes are *not tiled*, but *stretched*. Remember this to avoid bugs.
- AVS window doesn't have working *close* button's *press* image.
- While working with *cbuttons.bmp* remember, that Eject button, unlike the rest is one px smaller and therefore displaced.
- Winshade sliders have 3 states: before the middle, middle and after the middle.
- When the *stop* button is pressed, *main.bmp*'s background is displayed on a place of posbar and visualization window.
- It's useless to draw anything on *main.bmp*'s header: it is normally replaced by a title from *titlebar.bmp*.
- There could be two files with digits in your skin: *numbers.bmp* and *nums_ex.bmp*.
- Tiny stripes on the left of playlist's file/+/-/misc buttons could be used for design purposes. Don't forget to use button extractor, it's almost impossible to deal with them manually.
- Work with *templates* first: you will be able to edit details after extraction.
- Be careful with effects on *text.bmp*: a size of a single symbol's cell is quite small. And try not to replace "unnecessary" spaces – they're needful.

Text files

- All the text files are displayed using windows codepage, so try not to use any other (this is especially actual for *readme.txt*).

This section is not 100% completed yet, so wait for updates... If you have any tricks to share, just tell us.

ANOTHER COPYRIGHT STUFF

*This winamp 2.X skinning tutorial is © 2002 by Taras Brizitsky [AKA SacRat] and Maria (Mary) Gileva [AKA Imagine AKA Gileva] (excluding some parts, where other authors are mentioned). Some skin illustrations are based on *Template Amp* Version 1.0 by Mischa Klement. The front cover illustration is based on SacRat's Ballzzz 2 Futura skin, all the other illustrations in this text are made by SacRat and Imagine. Winamp Skin Specification version 2.0 is © by Fyre.*

You may use this tutorial for free, may make any amount of copies you want and even send it to your friends or upload it to your homepage, but you'll need to know some things first.

If you upload/send it you *must* leave this tutorial *exactly* in the form you got it: *no modifications* are allowed. Next, you must leave the authors' names near the link (when posting) and give us a e-mail note.

If you'd liked to make a translation, change something or just fix errors or grammar, feel free to contact us. *(SacRat)*

OUTRO

Here we finish our tutorial. Hope it was useful for you. Now you should know enough about skinning to start creating your own works right now.

This is the very first version of tutorial and we hope to continue developing and adding even more hints and tricks.

If you have found any mistakes here or something is not clear or you have any suggestions – feel free to contact us.

P.S.: we're getting congratulations: version 1.0.0 of this tutorial has been added to www.1001winampskins.com and could be downloaded from (http://www.1001winampskins.com/winamp_skinning_tutorials.html). All the updated versions should be there too. Great thanks to Max Bloch (max@1001winampskins.com) for that.

New versions and updates will follow soon! (SacRat)

ABOUT THE AUTHORS

SacRat:



Real name: Taras Brizitsky.

Age: 20 (11.11.1981).

Sex: male.

Location: Sarov, Russia.

Skins: B-Amp, FiorAmp, FiorAmp 1.1, AggieAmp 2K+2 (*remix of AggieAMP 2000 by Aggie*), Freqamp g2, Freqamp g3, Ballzzz Minimal, Ballzzz Minimal Colorpack, Ballzzz 2 Futura, Ballzzz 2 Futura Colorpack, Bend v3 (*remix of Bend v2 by The-Joker*), Ballzzz 3 Honey (*in collaboration with Imagine*), The Rusty One (*in collaboration with Imagine*).

Homepage: <http://sacrat.deviantart.com>,

E-mail: sacrat@newmail.ru

ICQ: [128620050](http://www.icq.com/128620050) (<http://www.icq.com/128620050>)

MSN: t_martian@hotmail.com (messenger only)

Imagine (Gileva):



Real name: Mary Gileva.

Age: 16 (2.12.1985).

Sex: female.

Location: Krasnoyarsk, Russia.

Skins: The Orange Amp, NOT_NAMED, FuturA, Greenie, Ballzzz 3 Honey (*in collaboration with SacRat*), The Rusty One (*in collaboration with SacRat*).

Homepage: <http://gileva.deviantart.com/>

E-mail: gileva@icm.krasn.ru

SHAMELESS AD

If you need a winamp skin with your company's advertisement, logo and so on, we could consider your offers (commercial skinning).

GREET'S'n'RESPECTS

SacRat:

I'd liked to thank the following people (in no order):

- All the international [Hugi](#) staff and personally Adok.
- All the [Deviantart](#) community members, all those, who commented my works, sent me notes and so on, but especially: c-specter, eikasia, Gaston, **Gileva**, girlfromverona, ice-hf, Iliks, Jark, jonnywedge, lockheart7ca, mach5, mariko, Olya, pigletteus, Poetess, princessleia, shiiizzam, Tanya, The-Joker.
- Members of all other art/skinning communities.
- All my pen-pals from all over the world.
- Special thank goes to my Hispanic friends: Tania Zamora, Rosaluz Salazar, Fabio, Yanet, Kath.

- Personal greets to Fiorela Agustí.
- Huge thanks to Fyre for his specs.
- Few more greets to my friends and pals IRL: Alex, Pavel, Alina, Artem, Olesya, Vlad and all the FIDONet staff of Sarov.
- Webmaster of www.1001winampskins.com for publishing my stuff.
- Anyone I've forgotten.
- *You* for reading this stuff.

Imagine:

- Everyone at DeviantArt, especially **SacRat**, vbdude, ice-hf, k1m, my great respect to jark, olya, torquema70, tangia, once more ice-hf :)
- Everyone who knows me.
- Everyone who doesn't know me.
- Everyone who has read this tutorial and didn't fall asleep.
- Just EVERYONE! :)

TESTERS

- Jan Klimo (jankoklimo@hotmail.com)
- C-Specter (<http://c-specter.deviantart.com/>)
- Iliks (<mailto:iliks@pochtamt.ru>)
- PanteraSuNo2 (<http://panterasuno2.deviantart.com/>)
- *You?*

Wanna see your name here?

Send a suggestion/bugreport, drop us few lines if you think, that any paragraph is too superficial.

GLOSSARY

A list of new words and abbreviations, used in text:

- 7-ZIP – a freeware archiving program (www.7-zip.org) and an original compression format (great for compressing PSD files).
- ANI – animated cursor file's extension.
- AVS (advanced visualization studio) a visualization plugin for winamp, which is often included into a standard program's distributive. AVS window could be skinned.
- BMP – windows bitmap file's extension. A popular graphical format with no data compression.
- Button extractor – a program, designed to help skinner cut template images into elements, which could be edited separately then.
- Clutterbar – a set of buttons on the left of time/vis area of the main window.
- Codetable – a way of storing characters. One-byte codetables often differ by a way of coding non-English characters. A text typed in one codetable and displayed in another often shows incorrectly.
- Color picker – a tool, designed to get a value (hex or decimal) of any pixel, displayed on your screen.
- CUR – cursor file's extension.
- Deviantart or Devart – a Netart community, located at www.deviantart.com.
- EQ – winamp equalizer.
- GIF – popular graphical format. Creates quite small images with a limited (256 max) amount of colors.
- HEX –hexadecimal.
- JPEG – a popular graphical format, good for storing photo quality data. Compression with losses allows 1:10 compression with no visible changes.
- JPG – the same, as JPEG.

- MAX – maximize.
- MB – winamp minibrowser. A window, which is used to display Internet content or browse Internet. Due to it's useless (it's too small to display any page normally), it's almost never used for what it was made for, but plugins may display some info their (like tags, lyrics and so on).
- MIN – minimize.
- MIN/MAX/CLOSE – three (or two, without MIN) buttons on the right top of Main window, EQ and PL.
- PL – winamp playlist.
- PNG – quite a new graphical format, created, as an alternative to GIF. Main features: smaller (than GIF) file size, true color support, loseless compression, and alpha channel transparency.
- PSD – Photoshop's file extension.
- PSP – Paintshop Pro's file extension.
- Posbar – position bar. Look at posbar.bmp
- Readme - a file, called readme.txt. Can often be found in skin's archive.
- Resample – unlike resize, this function makes better changes of the picture's size. A use of some special algorithms even allows you to make 2X image zooming without a critical loss of quality.
- RGB – (red, green, blue) – a method of storing color data, which will be displayed on the screen then.
- Skingenerator – a program, made to generate skins automatically from your image/photo. Unlike *button extractors* skingenerators are often used as the only skinning tool, as it's possible to generate skins, based on predefined templates.
- Scroller – the same as slider.
- Skin – a changeable program's interface.
- Skinner – someone, who create skins.
- Skinner.exe – one of the most popular skingenerators.
- Teknidermy – online magazine for designers/skinners. (<http://www.dezina.org/teknidermy/>)
- Tut – tutorial.
- Transliteration – a method of writing, when national characters are replaced by characters of Latin alphabet. For example, Russian “з” could be replaced by “z” (the same sound) and so on... In some cases people replace symbols, like “á” or “â” by simple Latin “a”.
- Truecolor – a graphical mode, which shows an “unlimited” number of colors (about 16 millions). Used to display high quality images.
- Unicode – a codetable, which allows you to write text, containing characters of several languages at the same time. For example, you could write an Arabian text with comments in English and Russian. Nowadays 2-byte (two bytes for any symbol) Unicode table is widely used in MS Windows 2000/XP.
- Vis – visualization.
- Viscolor – visualization color.
- Volbal – Volume/balance sliders.
- Win32 – 32 bit versions of Windows (from Windows95 to Windows XP, i.e. excluding windows 3.1).
- **Winamp** (<http://www.winamp.com/>) – a popular skinnable media player, created by **Nullsoft** (<http://www.nullsoft.com/>).
- Winshade or windowshade – look WS.
- WP – wallpaper.
- WS – windowshade mode, when a window is minimized into a small stripe to save space.
- WSZ (Winamp Skin Zipped) – winamp skin's extension.
- ZIP – a zip archived file's extension. ZIP is one of the most popular archiving formats.