

qt

Marcus Comstedt

COLLABORATORS

	<i>TITLE :</i> qt		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Marcus Comstedt	August 23, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	qt	1
1.1	Who's qt?	1
1.2	Overview	2
1.3	Installation	2
1.4	Using qt	3
1.5	Options	4
1.6	License	5
1.7	Bug reports	6
1.8	The author	6
1.9	History	7
1.10	Future	7

Chapter 1

qt

1.1 Who's qt?

qt 1.0

- A QuickTime player for fast Amigas -

©1996 Marcus Comstedt <marcus@lysator.liu.se>
Freely distributable

"We're not bees and we're not cats
or bugs or horses or things like that.
What we are is clear and absolute.
What we are, dear doctor, is CUTE!"
/ The Warner sibs

Overview

Why? When? How?

Installation

How to install qt

Using~qt

Basic principles of operation

Options

Command line switches

License~

Legal stuff

Bug~reports

Bugs? Who me? Naah...

The~author
The man, the myth, the concept

History
What has been

Future
What may yet become

1.2 Overview

"Moral #4: Never ask what hot dogs are made of."
/ Yakko

qt (pronounced cute) is a QuickTime movie player for the Amiga. I wrote this program because I wanted something CPU-intensive to run on my 68060, so it's main purpose is to eat computrons, the fact that it also plays movies rather nicely is just a side effect. :)

The requirements for running qt are:

- 68020 CPU or better
- AmigaOS 3.0 or better
- AGA and/or CyberGFX
- lots of memory

Depending on the actual movie, and the replay mode selected, an 020 may very well be a tad under specced...

QuickTime animation formats currently supported are:

- RLE
- Cinepak
- Road Pizza

Just for the record, here are it's advantages over xanim:

- It is smaller
- It is faster
- It supports 24bit graphics cards
- It has better sound handling
- It has a really neat HAM8 mode (if your CPU is fast enough :)

1.3 Installation

Brain: "Pinky, are you pondering what I'm pondering?"
Pinky: "I think so, Brain, but where will we find a duck and a hose at this hour?"

Not much to say here really. Either run the installer-script, or just copy the binary that best matches your CPU to some convenient place.

There are 5 binaries to select from:

qt - Any CPU from 68020 and up
qt.020fpu - Like abobe but with 68881/68882 FPU
qt.030 - 68030 and up without FPU
qt.030fpu - 68030 and up with FPU
qt.040fpu - 68040 with FPU

There is no principal difference between the 5 versions, they are just compiled with different compiler flags.

1.4 Using qt

```
"Moral #2: If at first you don't succeed, blame your parents!"  
/ Yakko
```

qt is designed to be run from CLI. It is possible to run it from Workbench, but you will not be able to specify files to play and options using the normal Workbench mechanisms. This will probably be fixed in the next version or so.

To play a movie from CLI, run the command qt with the filename of the movie and any options that you feel appropriate. If no filename is specified, a standard ASL file requester will appear.

The movie will now be loaded into RAM. If it is a big movie (>1MB) this may take a little while, and will eat lots of memory. However, loading the entire animation into RAM will ensure the smoothest possible playback, and that's what we all want, right?

If the loading completes without errors, a screen with a (hopefully) appropriate resolution and color-depth will appear and the movie will begin playing. After the entire animation has been played (both video and audio) the program will quit automatically, unless the LOOP option is given in which case it starts all over again.

To manually terminate the program, press Ctrl-C in the CLI window.

1.5 Options

```
"That's a finger. You have five of them on each hand.  
Unless you're in a circus. Then it's negotiable."  
 / Yakko
```

CLI Options:

```
FILE, EVERY/K/N, GRAY/S, HAM8/S, CYBERGFX/T, MODEID/K,  
WCP=WriteChunkyPixels/T, NOSOUND/S, NOWAIT/S, LOOP/S
```

FILE

The filename of the animation to play. If none is given, a file requester will be presented.

EVERY/K/N

A numeric argument limiting the programs desire to skip displaying frames in order to be able to keep up with the intended replay rate (and thus the soundtrack, if any). For example, EVERY=3 will make sure that no more than two subsequent frames are skipped at a time, meaning that at least every 3:rd frame will be displayed. EVERY=1 will cause all frames to be displayed no matter what.

GRAY/S

A switch causing true color animationes to be replayed in 256 grays if the hardware does not support 24bit screens. The default is to use a 8 \times 8 \times 4 color cube.

HAM8/S

A switch causing true-color animationes to be replayed in HAM8 on AGA machines. The conversion from true color to HAM8 is done "on the fly", so the CPU load will increase dramatically in this mode. 68060 recommended.

CYBERGFX/T

A toggle to indicate whether you want the program to select a CyberGFX screenmode or not. The default is to use CyberGFX screenmodes if cybergraphics.library is present, and AGA screenmodes if it is not. So normally there is no need to specify this switch.

MODEID/K

A numeric screenmode ID that you want to use. Prefix with "0x" for hexadecimal. The mode id may be for an AGA screen or a CyberGFX screen, regardless of the CYBERGFX switch.

WCP=WriteChunkyPixels/T

A toggle to indicate whether the program should use its own chunky to planar code, or use the system routines. A value of "on" will cause graphics.library/WriteChunkyPixels() to be used (AmigaOS3.1+ required), and a value of "off" means that the builtin routine should be used. The default is to use WriteChunkyPixels() only if there is chunky to planar hardware present in the system. (This switch has no effect for CyberGFX screens, the appropriate CyberGFX calls will always be used.)

NOSOUND/S

A switch turning off sound replay. It does not affect the timing of video playback in any other way than that it reduces the load on the CPU.

NOWAIT/S

Ignore timing information for video playback and display all frames as quickly as possible. No frames will be skipped, and no waiting will be done. See also EVERY.

LOOP/S

Specify this switch to play the animation over and over and over again... (Press Ctrl-C in the CLI window to abort.)

Workbench ToolTypes:

None. Sorry.

1.6 License

"Ehhhh, but let's not be anal."
/ Slappy

qt may be freely distributed in its unmodified archived form.
The program may not however be sold for profit.

The program is provided on an as-is basis, and there is no guarantee of functionality of any sort.

The author can not be held responsible for any physical or mental damage incurred directly or indirectly from the use or misuse this program.

Now, go ahead and have fun fun fun! :)

1.7 Bug reports

 "The hotel of your mind has many vacancies"
/ Yakko

If you find something which you consider an obvious bug or misfeature you may send me a bug report and perhaps I will actually fix the problem for future releases. :)
My address can be found in the
 next section
 .

If you do decide to send a bug report, please make it as detailed as possible. Especially, if the problem is that qt won't play a particular animation, a pointer to where that animation can be found would be mostly appreciated.

1.8 The author

 "Woah! Dumber than advertised..."
/ Dot

Here is my address and such:

Marcus Comstedt
Alsättersgatan 11A:21
S-584 35 Linköping
SWEDEN

E-mail: marcus@lysator.liu.se
WWW: <http://www.lysator.liu.se/~marcus/>

Feel free to drop a line if you have comments or suggestions. See also the section on

Bug reports

.

1.9 History

```
"Ehhhh, that snake doesn't have a leg to stand on.  
Hey! It's the dawn of time. What did you expect -- new jokes?"  
 / Slappy
```

This is the first official release. So I have absolutely no idea why I put this section in in the first place. :)

Well, here goes anyway:

```
V1.0 1996-08-05  
 - First official release
```

1.10 Future

```
Pinky: "Where are you going, Brain?"  
Brain: "Back to our cage, Pinky. We must plan for tomorrow night."  
Pinky: "Why? What are we going to do tomorrow night?"  
Brain: "Same thing we do every night, Pinky. Try to take over the world!"
```

Although I am pretty pleased with the program the way it is now, here are some improvements that I may decide to implement:

- Workbench interface with ToolTypes et al.
 - More video codecs
 - Replay in a window on a public screen
-