CyberQT

Thore Böckelmann

COLLABORATORS

	TITLE : CyberQT		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Thore Böckelmann	August 22, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

CyberQT

1.1   main

CyberQT 1.2

Copyright ©1996–1997 by Thore Böckelmann

CyberQT is a fast QuickTime animation player for 020+ machines with AmigaOS 3.0
or higher and a graphic board with CyberGraphX or AGA.

Background
Requirements
Installation

Supported encodings

Shell parameters
Workbench parameters

Operation

Frequently asked questions

Contact info

History
Bugs
To do

XAnim
Argue

Distribution
Credits
Disclaimer

CyberQT is Freeware
really want to I won't stop you from doing that :). But I would be glad if you

send me an EMail if you like CyberQT.

1.2   background

The main reason to develop CyberQT was because I was not very pleased with the
existing programs to view QuickTime animations. Although Marcus Comstedt's
program QT works very well with graphics boards and CyberGraphX, it has some
big disadvantages. Before playback the whole animation is decoded. By this a
very precise timing during playback is archived, but the animation must
completely fit in your machine's memory. Therefore the maximum size of
animations is limited by the free memory. Additionally it takes quite a long
time to decode an animation with "just" 10MB size, even with an MC68060!

So I began to program CyberQT. Development upto now (V1.0) took just two weeks.
This is because I could reuse big parts of CyberAVI (my program to view AVI
animations) and because Apple provides a very good QuickTime file format
specification. I wish I had something similar from Microsoft while developing
CyberAVI.

CyberQT was developed on:
  – A4000 with CyberStorm MK2 with MC68060/50
  – AmigaOS 3.1
  – 2MB ChipRAM
  – 16MB FastRAM
  – FastLaneZ3 SCSI–Host (disk capacity about 1.3GB)
  – CyberVision64 with 4MB graphic memory
  – Philips 17B 17" monitor

1.3   requirements

minimum hardware requirements:
  –Amiga with AmigaOS 3.0 (V39)
  –MC68020
  –graphic board with CyberGraphX (eg. CyberVision64, Picasso II, Retina, etc)
   or AGA chipset
  –2MB of FastRAM, 512K of ChipRAM

software requirements
  –CyberGraphX 2.21 or higher (if used with a graphic board)
  –asyncio.library V39 (included)

recommended hardware requirements:
  –Amiga with AmigaOS 3.1 (V40)
  –MC68040 at 25MHz minimum
  –graphic board with CyberGraphX 2.21 or higher

I recommend a registered version of CyberGraphX since the unregistered version

doesn't support 24 bit screenmodes.

1.4   install

Just use the supplied installer script.

If you are very experienced you can install CyberQT "by hand". Just copy
asyncio.library to LIBS: and copy CyberQT to where ever you want to. I
recommend to use the installer script.

1.5   encodings

CyberQT's implementation of QuickTime decoding is based on the source code to
Mark Podlipec's XAnim program.
I will add more decoders as I need them and get any animations to test them.

Supported encodings until now:

```
  Video:

    Radius CinePak      (CVID)                  8 bit (only gray)
    Radius CinePak      (CVID)          16/24/32 bit
    Kodak Photo CD      (KPCD)          16/24/32 bit
    Microsoft Video 1   (MSVC)             8/16 bit
    Apple Animation     (RLE)             1/4/8 bit (color and gray)
    Apple Animation     (RLE)           16/24/32 bit
    Apple Uncompressed (RAW)              1/4/8 bit (color and gray)
    Apple Uncompressed (RAW)            16/24/32 bit
    Apple Video         (RPZA)               16 bit
    Apple Graphics      (SMC)                 8 bit
    Component Video     (YUV2)          16/24/32 bit
    Intel Raw           (YUV9)          16/24/32 bit


  Audio:

    PCM         8/16 bit Mono/Stereo
    PCM0        8/16 bit Mono/Stereo
    TWOS        8/16 bit Mono/Stereo
    IMA ADPCM    16 bit Mono/Stereo
```

I am still looking for other encodings. Please contact me if you have an
animation with one of the following compressions:

```
  Video:
    - KPCD (Kodak PhotoCD)
    - or any other unsupported format, except "Intel Indeo"


  Audio:
    - MAC3
    - MAC6
```

- $\mathrm{\mu}$$-Law
- or any other unsupported format

You can send me any animation with one of the above compressions. Just compress it (with e.g. LhA) and send it uuencoded to one of my EMail addresses.

My worst problem is to support the Intel Indeo codec. Intel charges $5000 (in words: five thousand US-Dollars) for a licence to their source code. So it seems to be impossible that Indeo is ever supported :(

1.6 shell

The command template for CyberQT is...

```
FILES/M,PUBSCREEN/K,SCREENMODEREQ=SMR/S,FORCE24/S,GRAY=GREY/S,AGA/S,HAM8/S,
DITHER/S,BUFFERSIZE=BUF/K/N,DELAY/K/N,LOOP/S,MAXFPS/S,NOSOUND/S,NOVIDEO/S,
SKIP/S,AUDIOPRELOAD=APL/S,MAGNIFY/K/N,STATS/S,QUIET/S,DEBUG/S
```

FILES   One or more QuickTime files you want to view. If you don't specify a filename an ASL file requester will pop up to let you choose one or more animations. Wildcards, like ?, #? or *, are allowed.

PUBSCREEN   This options forces playback to a window on a public screen. Both the screen and the animation must have at least a depth of 15 bits. If this screen does not exist or if the screen's or animation's depth is lower than 15 bits or CyberQT will use its own screen.

                NOTE:
                    This option only works with CyberGraphX!

SCREENMODEREQ   Enables screenmode requester. If the screen is to be opened for playback you can choose a screenmode here. This requester will appear everytime CyberQT tries to open a screen.

FORCE24   Forces 16 bit animations to be displayed on a 24 bit screen. This may increase playback speed a bit.

                NOTE:
                    If CyberQT is running on a non-CyberGraphX system this option is totally useless and will be ignored.
                    This option only works for 16 bit animations.
                    For 8 bit animations this option will be ignored!!
                    For 24 bit animations this option is (of course) senseless.

GRAY/GREY   This switch will set a grayscale palette instead of the normal color palette. This may speed up playback, because less data have to be transfered.

AGA   Enables AGA support. This options doesn't need to be specified
                if you don't own a graphic board. CyberQT will recognize itself
                if either AGA or CyberGraphX is available. It is just meant to
                force playback on AGA if you want. If this option is specified
                or only AGA chipset is available all frames will be reduced to
                256 colors without doing any dithering, just color reduction.

                NOTE:
                    Do not expect too much from AGA support. AGA chipset is damn
                    slow compared to recent CyberGraphX compatible boards. And
                    since there are still some bugs in OS3.x's graphics.library
                    display may look corrupt on AGA screens.

HAM8   Forces playback of truecolor animations (16 bits or more) to a
                HAM8 screen. This may result in better looking pictures, but
                playback speed will slow down!

                NOTE:
                    This option only works with AGA!
                    This option is mutual exclusive with GRAY!

DITHER   This option enables ordered dithering of truecolor animations
                instead of just simple color reduction. This may result in
                better looking pictures, but playback speed will slow down!
                Here is a list of all encodings that support dithering.

                NOTE:
                    This option only works with AGA!
                    This option is mutual exclusive with GRAY!

BUFFERSIZE   This option sets the buffersize for asynchronous reading.
                asyncio.library allocates two buffers of the given size! Default
                value is 64K per buffer. The given value has to be between 1 and
                4096.

                NOTE:
                    This value is interpreted as the given amount of blocks of
                    1024 bytes. So a value of 50 means 51200 bytes and NOT 50
                    bytes!!

DELAY   CyberQT will wait the given amount of seconds before displaying
                the first frame. This is very useful if you have a slow
                synchronizing monitor, so you don't miss a few frames at
                beginning of the animation. Default value is 2 seconds. The
                given value has to be between 0 and 10.

LOOP   Enables looping of the animation when it finishes. The default
                is to exit at the animation's end.

MAXFPS   Set playback speed to maximum possible. In fact a delay of 0
                seconds per frame is set, so there won't be any delay and
                playback will be as fast as your machine can do.

NOSOUND   Disables sound playback. Only the video frames will be shown.

NOVIDEO   Disables playback of the video frames. If the animation also has
                audio frames these will be played alone. If the animation has no
                audio frames nothing will be done.

SKIP   This option enables skipping of video chunks if playback speed
                is too slow.

AUDIOPRELOAD   This option enables decoding of audio sample ahead of playback.
                If you have an animation with non-interleaved audio and video
                data and enough free memory you can set this option to avoid big
                seeks on your harddisk that may lead to asynchronous playback.

                NOTE:
                    This option is very memory intensitive. If you don't have
                    enough free memory playback will fail. CyberQT uses any free
                    memory to decode the audio samples, so it is not limited to
                    the amount of free ChipRAM.

MAGNIFY   This option sets a scaling factor for the display window's size.
                Possible values ust be between -4 and +4. Negative values cause
                smaller display, positive values cause bigger display than
                normal. no additional scaling. You should not set higher values
                than 2, because CyberGraphX gets really slow with factors of 3
                or 4. Values of 0 or -1 also cause no scaling.

                NOTE:
                    This option only works with CyberGraphX and on public
                    truecolor sceens! It has no effect when playback is done on
                    an own screen.

STATS   This option causes some statistical information to be displayed
                after playback. This will include the number of frames, the
                number of displayed frames, expected playback speed and real
                playback speed.

QUIET   This options will switch off all message printing. This does not
                affect error messages!

DEBUG   Enables debug mode. When switching debug mode on CyberQT will
                print much stuff to the output window describing the work being
                done.

Attention!! The following combinations of options are illegal and will cause
one option to be switched off:

    – AGA and PUBSCREEN   (AGA is switched off)
    – HAM8 and PUBSCREEN   (HAM8 is switched off)
    – HAM8 and GRAY   (GRAY is switched off)
    – DITHER and GRAY   (GRAY is switched off)
    – NOVIDEO and NOSOUND (NOVIDEO is switched off)

Examples:

  CyberQT blabla.mov

    This will just show blabla.mov with sound (if supported).

  CyberQT blabla.mov nosound maxfps

    This will show blabla.mov sound and at maximum possible speed.

  CyberQT blubb.mov screenmodereq buffersize=128

    This will show blubb.mov, but before playback starts CyberQT will ask you
    for a screenmode. A buffersize of 128K is used for reading.

  CyberQT brabbel.mov delay=5 loop stats

    This will show brabbel.mov with a delay of 5 seconds before starting
    playback. The loop option causes CyberQT to restart playback as soon as the
    animation's end is reached. Playback can be aborted by pressing a
    mousebutton, ESC or CTRL–C. After playback some statistical data will be
    shown.

  CyberQT veryslow.mov skip quiet gray

    This will show veryslow.mov without printing any information. If your
    machine isn't fast enough then delayed frames will be skipped if possible.
    A grayscale palette will be used instead of the normal colors.

  CyberQT verysmall.mov magnify 2 pubscreen ThisOne

    This will show verysmall.avi on the public screen "ThisOne" and all frames
    will be scaled by a factor of 2, this means double width and height.

1.7 workbench

The following ToolTypes are available for CyberQT's and any QuickTime ←
   animation's
icon:

WINDOW   This specifies the output window CyberQT uses. "NIL:" is the
                default value, so no text output will be visible. This option
                only exists if CyberQT was started from Workbench


```
PUBSCREEN        \
SCREENMODEREQ    |
FORCE24          |
GRAY or GREY     |
AGA              |
HAM8             |
DITHER           |
BUFFERSIZE       |
DELAY            | These options are identical with the shell options,
LOOP             |
MAXFPS           | so please have a look there for further explanation.
NOSOUND          |
NOVIDEO          |
SKIP             |
AUDIOPRELOAD     |
MAGNIFY          |
STATS            |
QUIET            |
DEBUG           /
```


NOTE:
     The DEBUG option only makes sense in combination with the WINDOW option. So
     if you specify DEBUG but not WINDOW you won't see any debug information.

1.8 dither

These encodings support ordered dithering with AGA:

```
    Radius CinePak     (CVID)          16/24/32 bit
    Kodak Photo CD     (KPCD)          16/24/32 bit
    Microsoft Video 1  (MSVC)                 16 bit
    Apple Video        (RPZA)                 16 bit
    Component Video    (YUV2)          16/24/32 bit
    Intel Raw          (YUV9)          16/24/32 bit
```

All other encodings do NOT support dithering!

1.9 operation

CyberQT will try to show the animation at the correct speed, but this mostly

depends on your machine. Faster processors (e.g. MC68040 or MC68060) and fast
graphic board will result in more smooth playback.

To quit CyberQT or stop playback you can do one of the following things:

  - press either mousebutton or ESC
     This will just terminate playing the current animation and is the only way
     to quit CyberQT if the LOOP option has been specified.
     Aborting playback by pressing a mousebutton only works if CyberQT is
     running on its own screen. When playback is done on a public screen the
     mousebuttons will have NO effect!

  - press CTRL-C
     This will quit CyberQT completely

  - send a Break to CyberQT's task with tools like Scout or XOper
     This is the same as pressing CTRL-C

During playback you can use the function keys to adjust playback speed:

  F1 - as fast as possible, same as MAXFPS option
  F2 - 1000% of normal speed
  F3 - 500% of normal speed
  F4 - 300% of normal speed
  F5 - 200% of normal speed
  F6 - normal speed
  F7 -   75% of normal speed
  F8 -   50% of normal speed
  F9 -   20% of normal speed
  F10 -   10% of normal speed

The real speed you get may differ from the speed it should be, but this depends
on you machine. The faster you machine is, the more exact playback speed will
be.

1.10   faq

Q: Why is "Intel Indeo" not supported? Most of my QuickTimes are saved with
   this encoding!

A: Intel charges $5000 (or more) for a non-disclosure agreement and the source
   to their Indeo technology. I am just a student and can't afford so much
   money. Now you can imagine?

Q: I have an animation whose video or audio compression is unknown to CyberQT.
   What can I do?

A: If this one is NOT an Intel Indeo (see above) animation, then please send it
   to me. I will try to support that compression in CyberQT's next release.

Take a look here for which encodings are currently supported (or not supported).

Q: Why does the sound sometimes listen very bad?

A: Either your machine isn't fast enough to decode the video frames in time and so the sound can't be played correctly, or it is my fault. Sound support still is not perfect. You can try the SKIP option to allow skipping of video frames if your machine is not fast enough.

If sound listenes much too slow (eg. voices of women listen like voices of men) the animation's audio frequency is probably higher than 28kHz. The Amiga's audio hardware has a limit of 28kHz, if you use screenmode like PAL or NTSC. With doublescanned modes (like DblPAL or DblNTSC) audio frequencies upto 56kHz are possible. Try starting CyberQT with option SCREENMODEREQ and choose a doublescan mode. If sound is then still listening the same way as before CyberQT has a bug :(

Q: With some of my animations CyberQT tells me, that the ".rsrc fork" is missing. Is there a way to view them nevertheless?

A: QuickTime animations were originally created on Apple Macintosh machines, whose filesystem splits a file into a data- and a resource-fork. When copying such animations the resource fork is often forgotten. Without this fork it is not possible to view this animation, because the resource fork contains all important information about the animation (encoding, resolution, etc).
Maybe you just forgot to rename the resource fork correctly. It must either have the same name as the animation and must be copied to the ".resource" directory, or it must have the extension ".rsrc" or ".rs".
One example: animation's name is "blafasel.mov"
The following files are possible to contain the resource fork:
  – .resource/blafasel.mov
  – blafasel.mov.rsrc
  – blafasel.mov.rs

Q: CyberQT doesn't display anything or crashes the machine!

A: Please contact me! It seems you have found a bug in CyberQT.

Q: CyberQT refuses to work on my A500 with MC68000/7 bought back in 1988! Why?

A: <sigh>!!

Q: CyberQT refuses to work on my 200MHz Pentium with Win95? Should I buy a faster processor and more memory?

```
A: YOU DAMN IDIOT!!
```

1.11 contact

```
My adresses:

  Snail mail:
    Thore Böckelmann
    Entgelhof 11
    D-32278 Kirchlengern    Tel: +49-5744-1309 and +49-5744-1323
    Germany

    Thore Böckelmann
    Stephanusstraße 82
    D-33098 Paderborn       Tel: +49-5251-730837
    Germany


  Electronic mail:
    tboeckel@uni-paderborn.de
    tboeckel@guardian.fido.de (prefered)
    FidoNet:  2:2432/230.15
    AmigaNet: 39:170/410.15


 NOTE:
   I prefer guardian.fido.de and my  FidoNet  account  for  "just  talking"  and
   bugreports.  If  you  want  to  send any uuencoded stuff (like animations for
   testing) then PLEASE send them to tboeckel@uni-paderborn.de, because  I  have
   (nearly) unlimited mailquota at the university.


The most recent version of CyberQT is always available by filerequest at  Blind
Guardian BBS. Just request "CyberQT.lha" at one of the following lines:

   Line 1: 38400 - 64000  Elink 310  +49-5742-920340  (2:2432/231.0)
   Line 2:  2400 - 28800  Elsa TQV   +49-5742-920341  (2:2432/230.0)
   Line 3:  2400 - 19200  ZyXEL EG+  +49-5742-920342  (2:2432/232.0)
```

1.12 history

```
V1.0:

   27-Feb-97   - first release on Aminet



V1.1:

   28-Feb-97   - fixed a bug in RPZA decoder with AGA that produced wrong
                 pixels/blocks of pixels
```

```
                        - fixed "too slow video playback" when video timescale is not
                          equal to anim timescale
                        - fixed delayed reaction when changing playback speed

        4-Mar-97        - CVID depth 32 is now supported, too

        5-Mar-97        - added support for IMA4 audio compression. Unfortunately I was
                          not able to test IMA4's stereo part because of no animation of
                          that kind
                        - improved handling of multi-track QTs

        6-Mar-97        - implemented SKIP option to skip delayed frames and keep audio
                          playback synchronous.

        8-Mar-97        - fixed sound loop bug when playing more than one animation at a
                          time. Reported by Thomas Börkel and many other people.
                        - improved error handling in "out of memory" situations

       14-Mar-97        - added support for KPCD video compression. Thanks to Lars Weigl
                          for his NASA animation.

       15-Mar-97        - fixed bug that made playback stuck because of wrong calculated
                          time offsets

       16-Mar-97        - fixed bug with wrong interpreted elst chunks
                        - small speed improvement for KPCD

       17-Mar-97        - hopefully fixed last problems with screenmode selection.
                          eg: 240x300 animations should be shown on a 640x480 screen
                              instead of a 320x240 screen.
                        - fixed some skipping problems with "keyframe-only" QTs

       19-Mar-97        - fixed problems with TWOS 16bit stereo sound. SamplesPerChunk
                          entry was misinterpreted as "bytes per chunk" instead of real
                          "samples per chunk", so not enough data were read.


  V1.2:

       22-Mar-97        - fixed some problems with oversized audio samples, although
                          audio playback may still stumble a bit (happened with trailer
                          to "Aliens 2")
                        - added support for PCM0 audio encoding

       23-Mar-97        - window on public screen is now resizeable on special request
                          by Thomas Börkel :)

       28-Mar-97        - fixed bad bug in RAW32 decoder
                        - added some frame size checks to avoid crashes with corrupted
                          frames
                        - fixed some problems with resource fork recognition
                        - fixed some problems with multi video tracks and different
                          encodings
                        - BIG speed improvement when converting YUV -> RGB in grayscale
                          mode
```

29-Mar-97   - fiddled around with depth selection of animation with mixed
              truecolor and colormapped encodings. I hope it is correct now.

31-Mar-97   - did many changes on timing calculations
            - audio samples are now buffered in a queue instead of playing
              them directly. Thus playback should not stumble anymore if
              audio and video data are interleaved.

 1-Apr-97   - Sorry, no jokes happened :)

 2-Apr-97   - animations with more audio frames than video frames per second
              are now played correctly again
            - animations with with very small audio samples (eg. one sample
              per frame instead of one sample per 5 frames) are now more
              often played correctly. Playback may still listen bad because
              of too much overhead with these very small samples.
            - decoding of mono audio samples sped up by using just one
              buffer instead of two. Two buffers are now used for stereo
              samples only.

 3-Apr-97   - added option AUDIOPRELOAD, which enables decoding of all audio
              samples before playback, to keep playback smooth with slow
              harddisks or non-interleaved animations.

 4-Apr-97   - fixed bug with PCM0 decoder
            - finally made animations with old style STSC chunk work

 7-Apr-97   - CyberQT now handles samples of ANY size. There is no limit of
              128K per sample for audio.device anymore.

 8-Apr-97   - fixed bug when animations with and without sound were played
              within one call of CyberQT (eg selected via requester)

 9-Apr-97   - fixed bug with RLE1 decoder, RLE4 decoder was called
              accidentially
            - fixed bug with wrong grayscale palette for CVID animations

10-Apr-97   - fixed bug with TWOS 16bit stereo audio that occured when
              reading sample sizes
            - changed FPS rate in stats from integers to floats to be more
              precise

14-Apr-97   - added option MAGNIFY to allow scaling of the display window on
              public by factors of 1..4

15-Apr-97   - CVID decoder produced wrong pixel blocks in grayscale mode.
              This is fixed now.

20-Apr-97   - CyberQT got a new icon. Much thanks to Bastian Kinne for
              painting it.

22-Apr-97   - added option DITHER for ordered dithering with AGA

25-Apr-97   - optimized checks for corrupted files. Truncated files are not
              always rejected, but recognized.

 2-May-97   - did some optimizations for RPZA dithering, looks better now

          3-May-97   - hopefully fixed the last elst-chunk problems

1.13 bugs

Known bugs:
  - Memory usage increases when playing more than one animation at a time.
    To avoid this just start CyberQT with only one animation at a time.
  - animations with audio frequency higher than ~28kHz are played much too slow
    due to Amiga's audio hardware. If you have an AGA machine or an ECS machine
    with graphics board you should enable a doublescan mode (DblNTC, DblPAL,
    etc), to allow audio frequencies upto 56kHz.

If you should find any additional bugs or if you have any suggestions please
contact me immediately.

If you should find some Enforcer hits, please report them to me.

Please include a capture of CyberQT's output when started with the DEBUG option
if there are any problems while playback (eg. black screen, immediate stop
after start, crashes, etc).

1.14 todo

Very urgent things to to (maybe already finished in the next release):
  - better sound support (will this ever be finished?)
  - support AHI for sound playback. This package offers much easier access to
    different types of audio hardware.

To do:
  - support for more video and audio encodings
  - find more bugs

1.15 xanim

The QuickTime encodings supported by CyberQT are based on the source code to
Mark Podlipec's XAnim program. XAnim supports a number of other QuickTime video
encodings, as well as AVI files.

  The key differences between CyberQT and XAnim for QuickTime files are:

    - XAnim is more powerful and more general than CyberQT.

    - CyberQT requires either a CyberGraphX compatible graphic board or AGA
      chipset. ECS is not and will not be supported!

    A gzip compressed archive of Mark Podlipec's XAnim can be obtained from:

```
     http://www.portal.com/~podlipec/home.html   "The XAnim Home Page"


   /*
    * xanim.c
    *
    * Copyright (C) 1990,1991,1992,1993,1994,1995,1996,1997 by Mark Podlipec.
    * All rights reserved.
    *
    * This software may be freely copied, modified and redistributed without
    * fee for non-commerical purposes provided that this copyright notice is
    * preserved intact on all copies and modified copies.
    *
    * There is no warranty or other guarantee of fitness of this software.
    * It is provided solely "as is". The author(s) disclaim(s) all
    * responsibility and liability with respect to this software's usage
    * or its effect upon hardware or computer systems.
    *
    */
```

1.16 argue

It was around 1992 when Commodore released their new Amiga OS 2.0. With this,
there were amazing changes for developers and users. All looked a bit more
professional, and a lot of things were just easy and better to handle than in
former times.

Earlier, developers had to write their own argument reading system. Often it
was really unpractically and difficult to understand.

The guys at Commodore knew that and thought about a new standard for argument
parsing to avoid confusion about all that. What they finally got was
ReadArgs(), a system function that parses arguments automatically.

Developers now only had to write a template to specify, what arguments they
would like to have. A template looks like this: FILE/A,SWITCH/S...

From now on, all the users could have a look at this template by adding a
question mark to the program's name to execute.

But all in all, there was a problem. Folks still had to go "down" into a shell
and type in all the arguments by hand.

So there are still a lot of people that write external interfaces for a
specific tool. Some of them are even shareware!

This was really annoying as there was no tool that could manage ALL tools.

In early 1996 I developed a GUI layout system called NiceGUI. It was crap, but
on this way I created the first version of Argue.

Argue's job was and is to read other tool's argument templates and prepare a
nice user interface where the user can decide what he would like to have as
arguments. Argue 0.3 was quite bad, but it was the first basis for further

development.

Some months later I invented how to write MUI applications. It was very easy, and I implemented a new version of Argue with it. This was called Argue 0.6 and released to some BBSs here in Germany.

From then on Argue made giant steps towards user friendliness and efficiency. New features were added in masses, and now, at the time of Argue 1.0, there is a (near ;) complete interface creation system.

1.17 distribution

CyberQT is Copyright ©1996-1997 by Thore Böckelmann.

CyberQT may be freely distributed as long as the following conditions are met:

 - all files have to be kept together
 - no file may be modified or crunched/packed
 - the only official way to distribute this program as archive is the original LhA archive

1.18 credits

CyberQT was written by Thore Böckelmann using Amiga Oberon 3.10 and GCC 2.7.2

Thanks to Matthias Greim for his "Peter Gabriel" CD-ROM with CVID animations.

Thanks to Peter Kunath for the source to his DrWho genie for chunky->planar conversion.

Thanks to all the other people who send me some animations for debugging, testing and enhancing CyberQT.

Thanks to Bastian Kinne (kinne@zfn.uni-bremen.de) for his new icon.

The XAnim program is written by Mark Podlipec. XAnim6 on the Amiga is a port by Terje Pedersen. XAnim is ©1990-1997 by Mark Podlipec.

The QT program was written by Marcus Comstedt . QT is ©1996-1997 by Marcus Comstedt.

asyncio.library was written by Martin Taillefer, Magnus Holmgren and Olaf Barthel

Argue is ©1996 by Thorsten Stocksmeier

1.19 disclaimer

No warranty, either express or implied, is made with respect to the fitness or merchantability of CyberQT.

Thore Böckelmann (referred to as "the author"), reserves the right to not develop any future versions of CyberQT.

The author will try to make a good faith attempt at correcting any problems if any are discovered, but is in no way required, nor bound to correct them.

The author neither assumes nor accepts any responsibility for the use or misuse of these programs. He will also not be held liable for damages or any compensation beyond the original registration fee due to loss of profit or any other damages arising out of the use, or inability to use this program.

The author will not be liable for any damage arising from the failure of this program to perform as described, or any destruction of other programs or data residing on a system attempting to run the programs.

The user of this program uses it at his or her own risk.