

## *A Work-In-Progress report by Paul Obermeier*



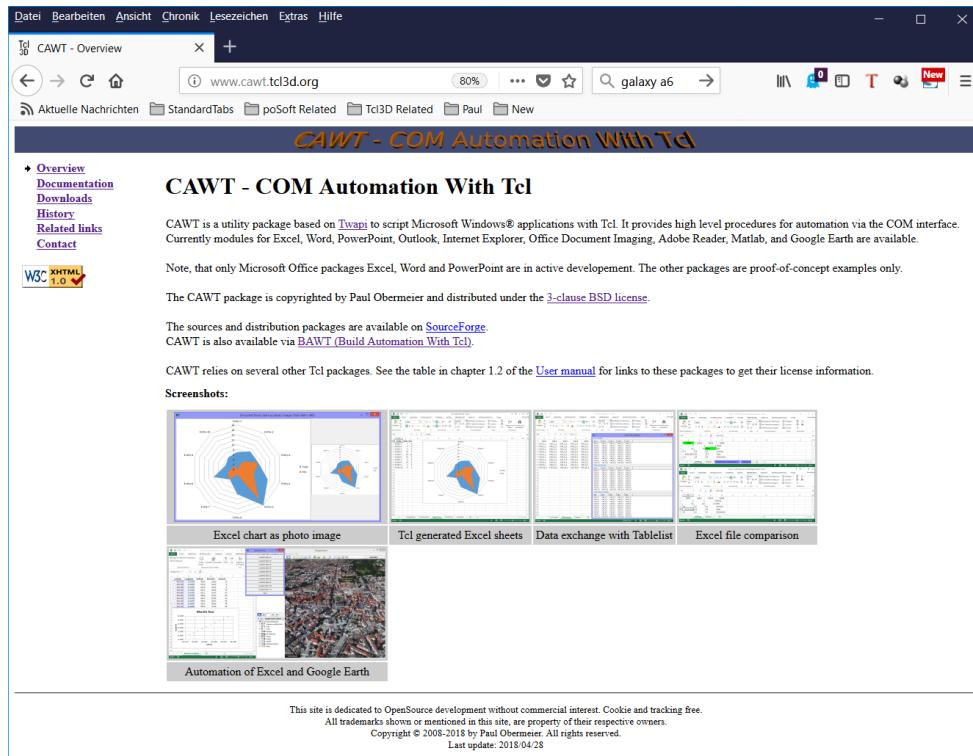
- **Introduction to Automation With Tcl**
- **MAWT Architecture**
- **MAWT Internals**
- **MAWT Status**
- **Summary**

# Introduction - Automation With Tcl

poSoft

## Episode 1: CAWT – COM Automation With Tcl

- Introduced at EuroTcl 2012. Currently at version 2.4.2.
- High-level Tcl interface for scripting Windows applications having a COM interface.
- Available at <http://www.cawt.tcl3d.org>
- Also available as part of BAWT-Tcl.

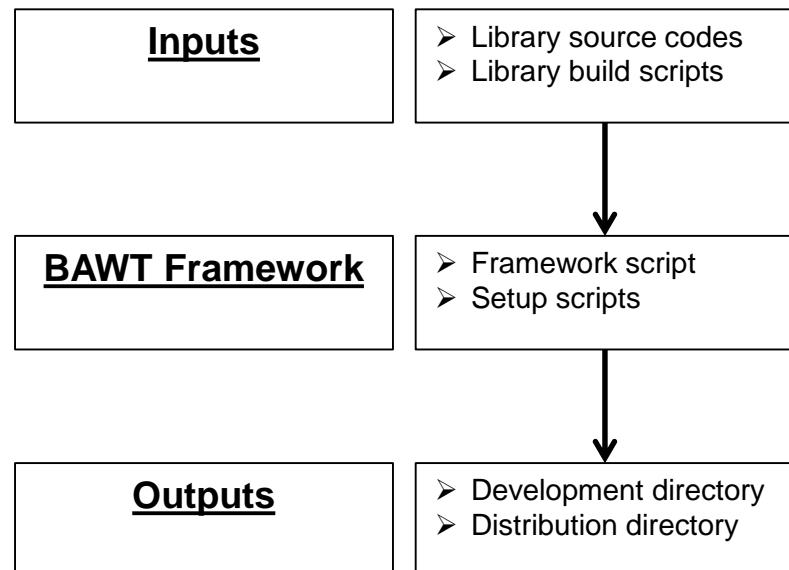


# Introduction - Automation With Tcl

poSoft

## Episode 2: BAWT – Build Automation With Tcl

- Introduced at EuroTcl 2016. Currently at version 0.8.0.
- Tcl based framework for automatically building C/C++ based libraries from source code.
- Available at <http://www.bawt.tcl3d.org>

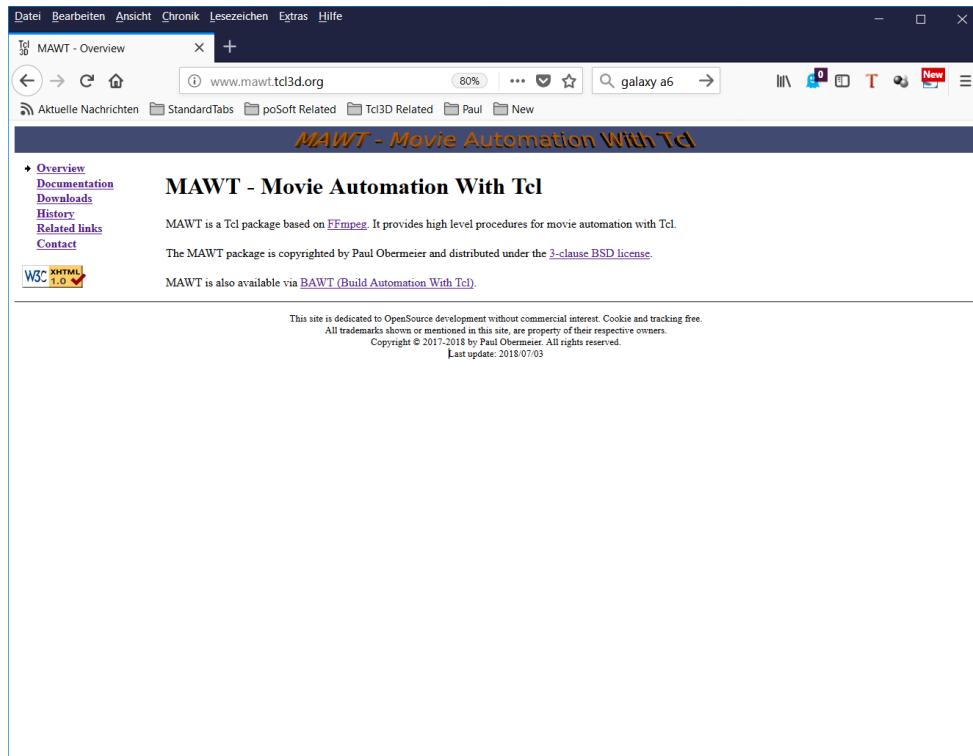


The screenshot shows the BAWT website interface. The top navigation bar includes "Datei", "Bearbeiten", "Ansicht", "Chronik", "Lesezeichen", "Extras", and "Hilfe". The address bar shows the URL [www.bawt.tcl3d.org](http://www.bawt.tcl3d.org). The main content area has a header "BAWT - Build Automation With Tcl" and a sub-header "BAWT is a Tcl based configurable framework for automatically building C/C++ based software libraries from source code." It mentions that it's used for Windows, Linux, and Darwin, and can be used on Linux and Darwin as well. The copyright information states it's copyrighted by Paul Obermeier and distributed under the 3-clause BSD license. Below this, a section titled "Supported Libraries" lists various libraries with their names, versions, platforms, and URLs. The table includes Boost, CMake, CppUnit, Eigen, Giflib, GLEW, Img, InnoTools, InnoSetup, and Tk.

#	Name	Version	Platform	Homepage
1:	Boost	1.62.0	Windows Linux Darwin	<a href="http://www.boost.org/">http://www.boost.org/</a>
2:	CMake	3.12.2	Windows Linux Darwin	<a href="https://cmake.org/cmake-Buildpack/">https://cmake.org/cmake-Buildpack/</a>
3:	CppUnit	0.12.0	Windows Linux Darwin	<a href="https://github.com/mpButcher/CppUnit">https://github.com/mpButcher/CppUnit</a>
4:	Canvas3d	1.2.2	Windows Linux	<a href="http://3dcanvas.tcltk.net/">http://3dcanvas.tcltk.net/</a>
5:	cawt	2.4.2	Windows	<a href="http://www.poSoft.de/html/extCawt.html">http://www.poSoft.de/html/extCawt.html</a>
6:	CMake	3.10.0	Windows	<a href="https://cmake.org/">https://cmake.org/</a>
7:	ChkWin32	3.10.0	Windows	<a href="http://www.chkwin32.com/">http://www.chkwin32.com/</a>
8:	crtcl	3.1.17	Windows Linux Darwin	<a href="http://andreas-kupries.github.com/crtcl/">http://andreas-kupries.github.com/crtcl/</a>
9:	curl	7.54.1	Windows Linux Darwin	<a href="https://curl.haxx.se/libcurl/">https://curl.haxx.se/libcurl/</a>
10:	DiffUtil	0.4.0	Windows Linux Darwin	<a href="https://github.com/rephuth/DiffUtilTcl/">https://github.com/rephuth/DiffUtilTcl/</a>
11:	Eigen	3.3.1	Windows Linux Darwin	<a href="http://eigen.tuxfamily.org/">http://eigen.tuxfamily.org/</a>
12:	Eigen	3.4.2	Windows Linux Darwin	<a href="http://eigen.tuxfamily.org/">http://eigen.tuxfamily.org/</a>
13:	ffmpeg	3.4.2	Windows Linux Darwin	<a href="http://www.ffmpeg.org/">http://www.ffmpeg.org/</a>
14:	fftw	3.3.7	Windows Linux Darwin	<a href="http://www.fftw.org/">http://www.fftw.org/</a>
15:	Freeglut	3.0.0	Windows Linux Darwin	<a href="https://sourceforge.net/projects/freeglut/">https://sourceforge.net/projects/freeglut/</a>
16:	FreeType	2.10.2	Windows Linux Darwin	<a href="https://sourceforge.net/projects/freetype/">https://sourceforge.net/projects/freetype/</a>
17:	FTGL	2.1.3	Windows Linux Darwin	<a href="https://sourceforge.net/projects/ftgl/">https://sourceforge.net/projects/ftgl/</a>
18:	gdal	2.2.0	Windows Linux Darwin	<a href="http://www.gdal.org/">http://www.gdal.org/</a>
19:	GeographicLib	1.46	Windows Linux Darwin	<a href="http://geographiclib.sourceforge.net/">http://geographiclib.sourceforge.net/</a>
20:	GeographicLibData	1.46	Windows Linux Darwin	<a href="http://geographiclib.sourceforge.net/">http://geographiclib.sourceforge.net/</a>
21:	giflib	4.2.3	Windows Linux Darwin	<a href="http://giflib.sourceforge.net/">http://giflib.sourceforge.net/</a>
22:	G12P	1.4.0	Windows Linux Darwin	<a href="http://www.geuz.org/g12p/">http://www.geuz.org/g12p/</a>
23:	GLEW	2.0.0	Windows Linux Darwin	<a href="http://glew.sourceforge.net/">http://glew.sourceforge.net/</a>
24:	Img	1.4.7	Windows Linux Darwin	<a href="https://sourceforge.net/projects/kimg/">https://sourceforge.net/projects/kimg/</a>
25:	InnoTools	0.9.0	Windows Linux Darwin	<a href="http://www.kinimil.com/sourceforge.net/">http://www.kinimil.com/sourceforge.net/</a>
26:	InnoSetup	5.5.9	Windows	<a href="http://www.jrsoftware.org/iisinfo.php">http://www.jrsoftware.org/iisinfo.php</a>
27:	tk	4.1.0	Windows Linux Darwin	<a href="https://sourceForge.net/projects/incrto1/">https://sourceForge.net/projects/incrto1/</a>

## Episode 3: MAWT – Movie Automation With Tcl

- Introduced at EuroTcl 2018. Currently at version 0.1.0.
- High-level Tcl interface offering some of the video functionality of FFmpeg.
- Available at <http://www.mawt.tcl3d.org>
- Also available as part of BAWT-Tcl.



# Introduction - FFmpeg

poSoft

- A complete, cross-platform solution to record, convert and stream audio and video.
- It is programmed in the ISO C90 language with a few additional features from ISO C99.
- Available at <http://www.ffmpeg.org>



FFMPEG consists of several libraries:

**libavutil** contains functions for simplifying programming (ex. mathematics routines, core multimedia utilities, ...).

**libavcodec** contains decoders and encoders for audio/video codecs.

**libavformat** contains demuxers and muxers for multimedia container formats.

**libavdevice** contains input and output devices for access to many common multimedia input/output software frameworks.

**libavfilter** contains media filters.

**libswscale** performs highly optimized image scaling and color space/pixel format conversion operations.

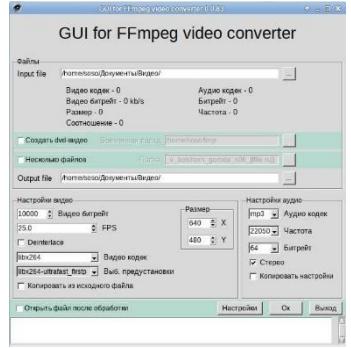
**libswresample** performs highly optimized audio resampling, rematrixing and sample format conversion operations.

# Introduction – Existing Tcl Video Packages

poSoft

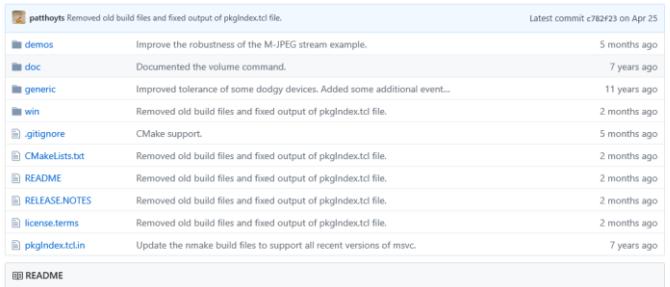
- **tkffmpeg** Graphical user interface for FFmpeg batch program.

<https://sourceforge.net/projects/tkffmpeg/>



- **tkvideo** DirectShow binding. Windows only.

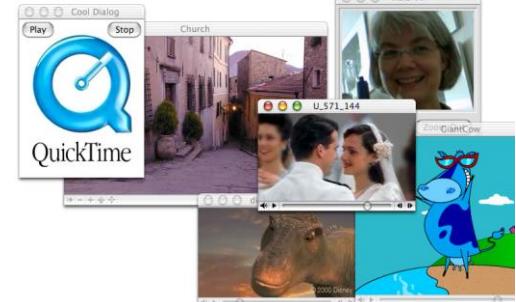
<http://www.patthoys.tk/tkvideo/>



- **QuicktimeTcl** Binding for Quicktime.

<http://quicktimetcl.sourceforge.net/>

QuickTime for Tcl/Tk on Mac OS X  
and **Windows**



# MAWT Architecture – Layer Overview

MAWT is realized as a 3 layer architecture.

mawtInterface.tcl

mawtQuery.tcl

mawtUtil.tcl

MAWT Interface (Tcl Level)

bytearray.i

tkphoto.i

vectors.i

FFmpeg Interface (SWIG Level)

ffmpegif.h

ffmpegifPrivate.h

ffmpegif.c

FFmpeg Interface (C Level)

libavutil

libavcodec

libavformat

libswscale

FFmpeg Libraries (C Level)

# MAWT Architecture – C Level

The FFmpeg C interface defines high level functions for handling video data

```
typedef struct videoStruct *VideoId;

VideoId VideoOpen  (const char *fileName);
void     VideoClose (VideoId videoId);
Bool     VideoGetNextImage (VideoId videoId, void *img, Bool flip);

VideoId VideoCreate(const char * fileName, int width, int height);
Bool     VideoFinish(VideoId videoId);
Bool     VideoWriteNextImage(VideoId videoId, void *img, Bool flip);

int      VideoStart (VideoId videoId, int width, int height);
void    VideoStop  (VideoId videoId);

int      VideoGetNumFrames (VideoId videoId);
double   VideoGetFramerate (VideoId videoId);
int      VideoGetWidth   (VideoId videoId);
int      VideoGetHeight  (VideoId videoId);

const char *GetErrorMessage (void);
const char *GetFFmpegVersion (void);
```

```
typedef struct videoStruct {
    struct SwsContext *swsCtx;
    AVFormatContext *formatCtx;
    AVCodecContext *codecCtx;
    AVOutputFormat * outFmt;
    AVFrame *rgbFrame;
    AVFrame *yuvFrame;
    AVStream * stream;
    AVPacket packet;
    int videoStream;
    int audioStream;
    char name[100];
    int numFrames;
    int started;
    int curFrame;
    int width, height;
    int outWidth, outHeight;
    unsigned int numTotalFrames;
    double frameRate;
    FILE *fp;
} VideoStruct;
```

ffmpeginf.h

ffmpeginfPrivate.h

ffmpeginf.c

FFmpeg Interface (C Level)

# MAWT Architecture – SWIG Level

The FFmpeg SWIG interface implements low level C functions for handling vector data and transferring vector data to/from byte arrays and photo images.

```
int VectorFromByteArray (
    Tcl_Interp *interp,
    const unsigned char *src,
    void *dest,
    int numBytes,
    int srcOff,
    int destOff
);

int VectorToByteArray (
    Tcl_Interp *interp,
    const void *src,
    unsigned char *dest,
    int numBytes,
    int srcOff,
    int destOff
);
```

```
int VectorFromPhoto (
    Tcl_Interp *interp,
    char *imgName,
    void *imgVec,
    int numChans,
    float scale,
    float offset
);

int VectorToPhoto (
    Tcl_Interp *interp,
    void *imgVec,
    const char *photoName,
    int width,
    int height,
    int numChans,
    int linesAreBottomUp);
```

```
%define %baseTypeVector(TYPE,NAME)
...
static TYPE *new_##NAME(int nelements)
{
    return (TYPE *)calloc(nelements,sizeof(TYPE));
}

static void delete_##NAME(TYPE *ary)
{
    free(ary);
}
...
%enddef

// Generate vector functions
// for the following types.
%baseTypeVector(unsigned char, UByte)
%baseTypeVector(char, Byte)
```

bytearray.i

tkphoto.i

vectors.i

FFmpeg Interface (SWIG Level)

# MAWT Architecture – Tcl Level

The MAWT Tcl interface defines the TclOO classes **Video** and **Vector**.

It also adds query and utility procedures.

```
oo::class create Video {
    variable mVideoId

    method Open { fileName } {
        set mVideoId [ffmpeginf::VideoOpen $fileName]
    }
    method GetNumFrames {} {
        return [ffmpeginf::VideoGetNumFrames $mVideoId]
    }
}

oo::class create Vector {
    constructor { numBytes } {
        set mVec [ffmpeginf::new_UByte $numBytes]
    }
    method FromPhoto
    method ToPhoto
}
```

```
proc GetFfmpegVersion {}
```

```
proc VideoExport {
    movieFile
    imgFilePattern
    args
}
```

```
proc VideoImport {
    imgFilePattern
    movieFile
    args
}
```

mawtInterface.tcl

mawtQuery.tcl

mawtUtil.tcl

MAWT Interface (Tcl Level)

## Dependencies to build MAWT:

- CMake
- SWIG
- FFmpeg
- Tcl/Tk

## Dependencies to run MAWT:

- Tcl/Tk
- FFmpeg
- *img::raw (optional)*

# MAWT Internals – Transfer Mode Put

- Create photo image, video and vector object.
- Attach photo image to label.
- Get next video frame.
- Create bytearray from vector data. **1 copy**
- Update photo image with **put** command using the bytearray data. **1 copy**



```
# Initialization
label .l
image create photo VideoFrame
.l configure -image VideoFrame

set videoObj [mawt Video new $movieFile]
set vectorObj [mawt Vector new $numBytes]

# Display loop
while { true } {
    $videoObj GetNextImage [$vectorObj Get]
    set imgBinVar [$vectorObj ToByteArray [expr {3 * $tx * $ty}]]
    VideoFrame put $imgBinVar -format \
        "raw -nomap 1 -width $tx -height $ty -nchan 3 -useheader 0"
    update
}
```

# MAWT Internals – Transfer Mode Create

- Create video and vector object.
- Get next video frame.
- Create bytearray from vector data.
- Create photo image with **image create** command using the bytearray data. **1 copy**
- Attach photo image to label. **1 copy**



```
ModeCreate.tcl (C:\poSoft\tcl3d-...rtraege\EuroTcl2018-Mawt) - GVIM1
Datei Editieren Werkzeuge Syntax Puffer Ansicht Hilfe
File Edit Tools Syntax Buffer View Help
label .1

set videoObj [mawt Video new $movieFile]
set vectorObj [mawt Vector new $numBytes]

# Display loop
while { true } {
    $videoObj GetNextImage [$vectorObj Get]
    set imgBinVar [$vectorObj ToByteArray [expr {3 * $tx * $ty}]]
    catch { image delete VideoFrame }
    image create photo VideoFrame -data $imgBinVar -format \
        "raw -nomap 1 -width $tx -height $ty -nchan 3 -useheader 0"
    .1 configure -image VideoFrame
    update
}
"ModeCreate.tcl" [Neu] 16L, 472C geschrieben 13,2 Alles
```

# MAWT Internals – Transfer Mode Mawt

- Create photo image, video and vector object.
- Attach photo image to label.
- Get next video frame.
- Update photo image with **ToPhoto** command of Vector class.

1 copy



```
ModeMawt.tcl (C:\poSoft\tcl3d-Sa...rtraege\EuroTcl2018-Mawt) - GVIM1
Datei Editieren Werkzeuge Syntax Puffer Ansicht Hilfe
File Edit Tools Syntax Buffer View Help
label .1
image create photo VideoFrame
.1 configure -image VideoFrame

set videoObj [mawt Video new $movieFile]
set vectorObj [mawt Vector new $numBytes]

# Display loop
while { true } {
    $videoObj GetNextImage [$vectorObj Get]
    $vectorObj ToPhoto VideoFrame $tx $ty
    update
}
~

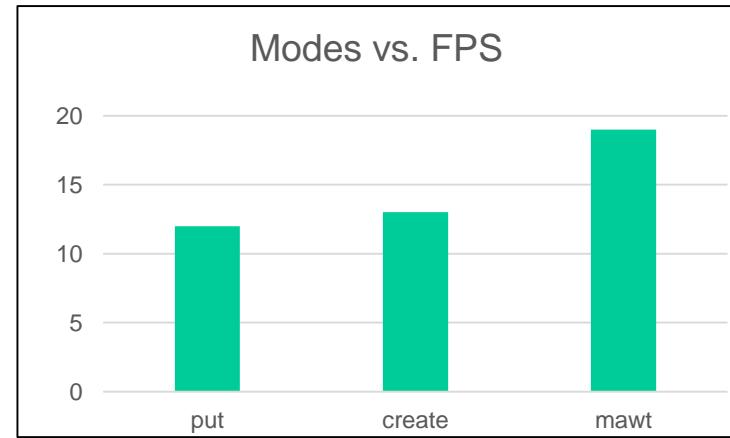
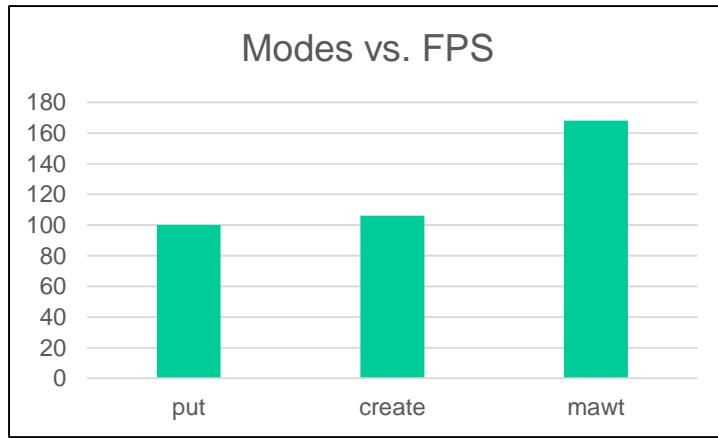
"ModeMawt.tcl" [Neu] 14L, 311C geschrieben      12,39      Alles
```

# MAWT Status – Performance Measurements

Performance measurements done with script `speedTest.tcl`

Comparison of transfer modes		
Mode	Size	FPS
put	640x464	100
create	640x464	106
mawt	640x464	168

Comparison of transfer modes		
Mode	Size	FPS
put	1920x1080	12
create	1920x1080	13
mawt	1920x1080	19

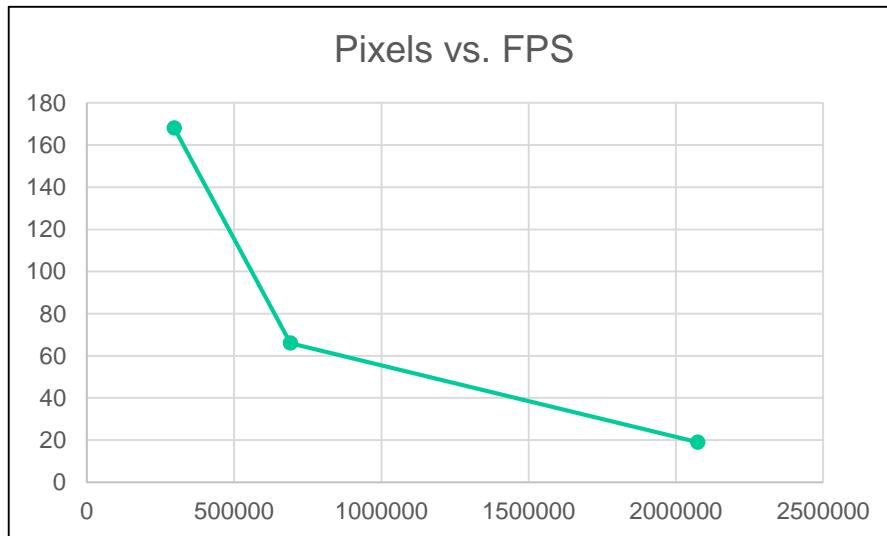


CPU: Core i7-4700HQ 2.4GHz

# MAWT Status – Performance Measurements

Performance measurements done with script `speedTest.tcl`

Comparison of video sizes			
Mode	Size	Pixels	Measurement
mawt	640x464	296,960	Faster than needed: 6.70 times (168 fps)
mawt	960x720	691,200	Faster than needed: 2.66 times (66 fps)
mawt	1920x1080	2,073,600	Faster than needed: 0.62 times (19 fps)



CPU: Core i7-4700HQ 2.4GHz

# MAWT Status – Demo and Test Programs

poSoft

## MAWT Demo: Simple movie player movieplayer.tcl

MAWT demo: A simple movie player

Movie list					
Filename	Width	Height	Duration	Frames	Frame Rate
Face2.avi	160	120	00:03.01	91	30
1920x1080.mp4	1920	1080	00:10.00	300	30
640x464.mp4	640	464	00:04.01	101	25
960x720.mp4	960	720	00:04.01	101	25
Numbers.mpg	256	256	00:00.00	0	25
TheKnack.mpg	240	180	01:01.24	1549	25
1984.flv	320	236	00:59.14	1430	24



1 of 1549

◀ ▶ ⟲ ⟳

MAWT demo: A simple movie player

Movie list					
Filename	Width	Height	Duration	Frames	Frame Rate
Face2.avi	160	120	00:03.01	91	30
1920x1080.mp4	1920	1080	00:10.00	300	30
640x464.mp4	640	464	00:04.01	101	25
960x720.mp4	960	720	00:04.01	101	25
Numbers.mpg	256	256	00:00.00	0	25
TheKnack.mpg	240	180	01:01.24	1549	25
1984.flv	320	236	00:59.14	1430	24



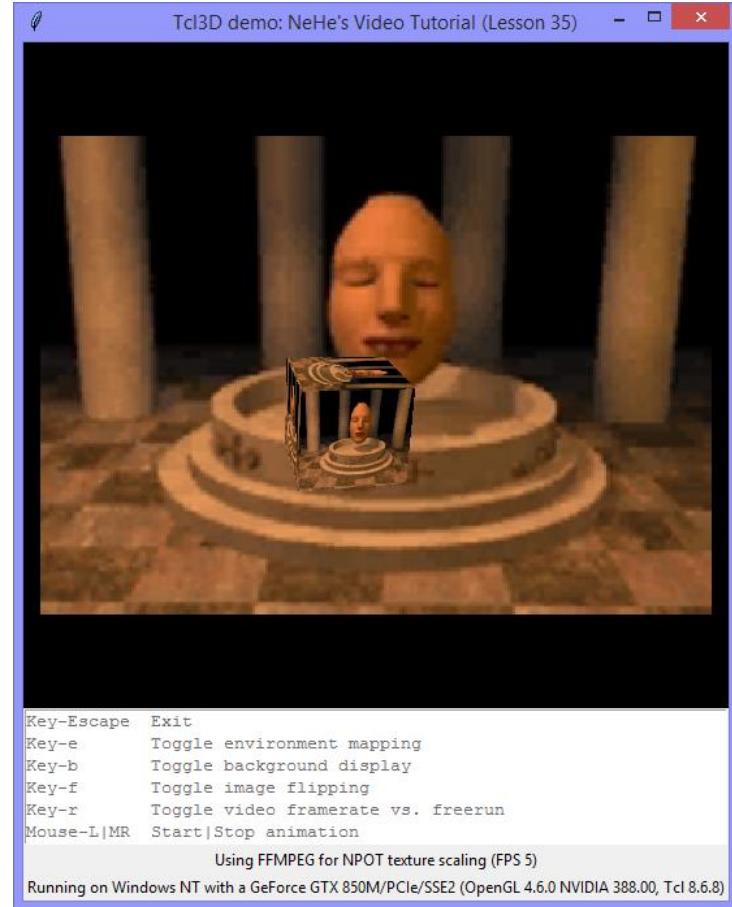
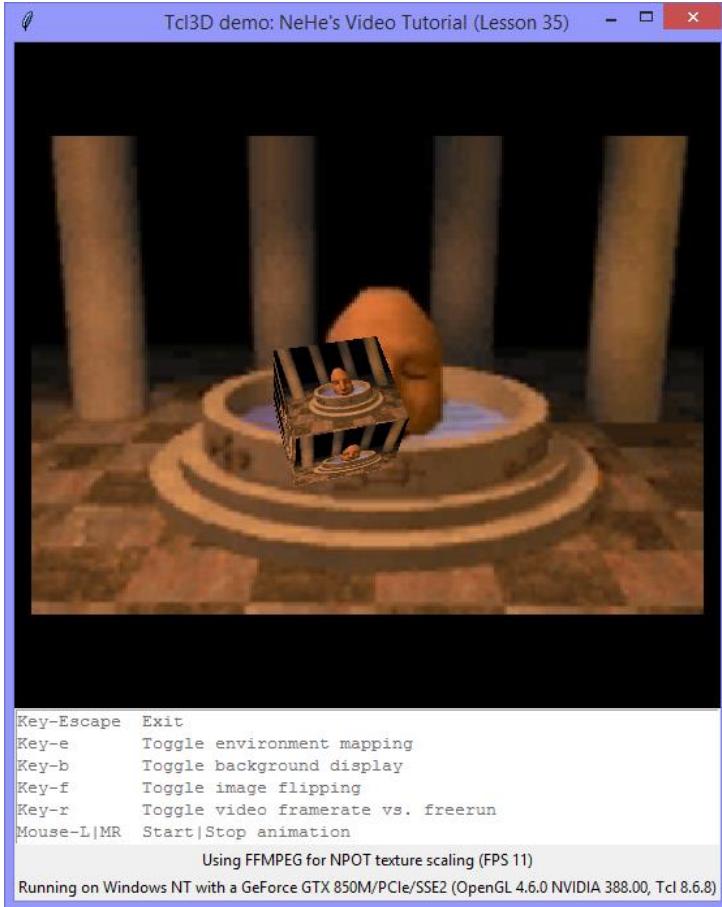
471 of 1549

◀ ▶ ⟲ ⟳

# MAWT Status – Demo and Test Programs

poSoft

## MAWT Demo: Tcl3D with animated textures NeHe-Lesson35.tcl



# MAWT Status – Demo and Test Programs

poSoft

## MAWT Demo: Test scripts for procs VideoImport and VideoExport

```
> tclsh VideoExport.tcl
```

```
FFmpeg version      : 3.4.2
Movie file         : Data/Face2.avi
Image file pattern: TestOut/Face.%03d.png
Number of images   : 51
Time needed        : 0.16 seconds
```

```
> tclsh VideoImport.tcl
```

```
FFmpeg version      : 3.4.2
Image file pattern: TestOut/Face.%03d.png
Movie file         : TestOut/Face2.mpg
Number of images   : 51
Time needed        : 0.10 seconds
```

# Summary

poSoft

## Current status

- ✓ Architecture and build structure defined.
- ✓ Video display with Tcl/Tk and FFmpeg is possible.
- ✓ Exporting images from a video is possible.
- ✓ Generating a video from images is possible.

## Future work

- Improve display performance.
  - Implement zero-copy transfer mode.
  - Improve FFmpeg compilation in BAWT.
- Improve support of video formats.
- Improve FFmpeg interface.
- *Add audio support.*

## MAWT Resources

MAWT Sources ( Homepage of MAWT )	<a href="http://www.mawt.tcl3d.org">http://www.mawt.tcl3d.org</a>
MAWT Binaries ( Part of BAWT-Tcl )	<a href="http://www.bawt.tcl3d.org">http://www.bawt.tcl3d.org</a>