

# **ShowObj Documentation**

Andreas Heumann

**COLLABORATORS**

	<i>TITLE :</i> ShowObj Documentation		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Andreas Heumann	April 12, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>ShowObj Documentation</b>	<b>1</b>
1.1	ShowObj Documentation . . . . .	1
1.2	Introduction . . . . .	2
1.3	Requirements . . . . .	2
1.4	Installation . . . . .	3
1.5	Inputs . . . . .	3
1.6	Usage . . . . .	3
1.7	Legal Stuff . . . . .	5
1.8	MUI . . . . .	6
1.9	Credits . . . . .	6
1.10	Author . . . . .	7
1.11	History . . . . .	7
1.12	Future . . . . .	8
1.13	RayStorm . . . . .	8

---

# Chapter 1

## ShowObj Documentation

### 1.1 ShowObj Documentation

23 ↔  
 June ↔  
 ↔  
 1995 ↔

```

                S h o w O b j
v2.2           An Imagine TDDD-object and LightWave LWOB-object ←
                displayer.
                Copyright 1994,95 by Andreas Heumann.
                All Rights reserved.
    
```

```

-----
                Introduction
                | | | | |
                Requirements
                ----- |---| | | |
                Installation
                | | | | | | | -| |
-----
                Inputs

                Usage
                -----
                Legal Stuff
                | | | | |
                Credits
                | | |--- |
                Author
                | | | | |
                History
                -----
                Future

                RayStorm
    
```

## 1.2 Introduction

Introduction

=====

Showobj takes a file in Imagine's TDDD (3D data description) or in LightWave's LWOB format and displays it's contents in a window on a public screen, or if no is given on Workbench. If no filename is given, the program will ask for it with a filerequester.

Showobj can display the objects in three different modes:

- boundingbox mode
- wireframe mode
- dither solid mode
- gray solid mode
- color solid mode

Four views modes can be selected:

- front view
- right view
- top view
- perspective view

In perspective view the object can be rotated with the mouse while pressing the left mouse button, and zoomed while pressing the right button.

In front, right and top view it can be moved with the mouse while pressing the left button.

With the cursor keys the object is moved in front, right and top view, and rotated in perspective view.

It's possible to track the camera in perspective mode to the objects, and get information about them. Objects tracked to are drawn in wireframe mode in a different color.

Dither, color and gray solid mode are realized with a Z-Buffer algorithm which will need a large amount of memory in high resolution modes (655kByte with 640x256, 1,8Mb with 800x600 and 3Mb with 1024x768).

If there's not enough memory available the window is divided into stripes, but then it takes a bit longer to draw the object.

The fastest solid mode is dithered solid mode, and especially on screens with only a little colors it's the best mode.

## 1.3 Requirements

Requirements

=====

- (1) ShowObj needs at least OS 3.0, nevertheless it should run on any machine with any processor.
- (2) ShowObj was written using  
MUI  
. So you need

muimaster.library V2.0+ or later to run ShowObj.

Tested with:

A2000 4Mb 68030/882 ECS  
A2000 9Mb 60040, Merlin Gfx Board

## 1.4 Installation

Installation  
=====

Installing ShowObj is really easy, you only have to copy the file 'showobj' or 'showobj\_020' if you have a 68020+68881/2 processor to a directory where you can find it ('c:' is recommended).

## 1.5 Inputs

Inputs  
=====

TDDD: Can read and display:

- normal Objects saved in detail editor
- perfect spheres
- grounds

Can't read and display:

- Lamps
- Others than the first state of a state-object
- Others than the first state of a cycle-object
- Paths
- Objects saved in forms or spline editor

LWOB: Can't display polygons with more than four points (rectangles) in shaded mode.

## 1.6 Usage

Usage  
=====

Workbench usage  
-----

Simply click on the showobj icon. If another icon is selected ShowObj attempts to load it.

ShowObj understands following Tooltypes:

PUBSCREEN=<name>  
    opens showobj window on pubscreen with name <name>  
VERBOSE=<TRUE or FALSE>

---

verbose mode, displays information about file  
 TDDD: names of Objects; amount of objects, faces, edges and points  
 LWOB: names of Surfaces; amount of surfaces, faces, edges and points

#### Shell usage

-----

\*\*\*\*\* IMPORTANT \*\*\*\*\*  
 If you want to display objects with many parts (more than 20) you need a larger stack than the default stack of 4096 Bytes!  
 I had a object with more than 200 parts which needed a stack of 20000 bytes. So if ShowObj has problems with large objects, use a larger Stack!  
 \*\*\*\*\* IMPORTANT \*\*\*\*\*

showobj [verbose] [pubscreen <name>] filename

#### With:

verbose verbose mode, displays information about file  
 TDDD: names of Objects; amount of objects, faces, edges and points  
 LWOB: names of Surfaces; amount of surfaces, faces, edges and points

pubscreen opens showobj window on given pubscreen

filename name of the TDDD- or LWOB-file which is displayed, if no name is given a filerequester appears.

#### Tool window

-----

Load	load a new object
About	informations about ShowObj
Browser	open browser window
Quit	quit ShowObj

#### Displaymode:

Wireframe	wireframe mode
Dither Solid	dither solid Z-Buffer mode
Gray Solid	gray solid Z-Buffer mode
Color Solid	color solid Z-Buffer mode
BoundingBox	boundingbox mode

#### View:

Front	view from front
Right	view from right
Top	view from top
Perspective	perspective view

#### Browser window

-----

Object names are displayed in the listview on top of the browser window and can be selected there. The three following gadgets use a selected object.

Track            track to selected object  
Info            show information about selected object  
Remove         remove selected object and all objects which are grouped to  
                 it  
Close          close browser window

Display window  
-----

This keys can be used in the display window:

ESC - Quits ShowObj  
w - Switches wireframe mode on  
b - Switches bounding box on  
d - Switches dither solid mode on  
g - Switches gray solid mode on  
c - Switches color solid mode on  
f - Switches to front view  
r - Switches to right view  
t - Switches to top view  
p - Switches to perspective view  
i - Zoomes in  
o - Zoomes out  
Cursorkeys to move in front, right and top view and to rotate in  
perspective view.

## 1.7 Legal Stuff

Legal Stuff

=====

### DISCLAIMER

THERE IS NO WARRANTY FOR THIS PROGRAM TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHERE OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDER AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

COPYRIGHT

---



ShowObj 2.1 Copyright 1994,95 by Andreas Heumann.  
All Rights Reserved. It is released under the concept of 'Cardware'.  
So if you like it please send me a postcard of your city. It has never  
been, nor will it ever be in the 'Public Domain'.

The archive of the RayStorm Demo may only be distributed in unmodified  
form. No files may be added, changed or removed. You may not charge for  
this archive, other than the cost of the media and duplication fees.  
Distribution is allowed in all forms, such as BB systems, floppy or  
compact disks, and ftp sites.

MUI

## 1.8 MUI

This application uses

MUI - MagicUserInterface

(c) Copyright 1993/94 by Stefan Stuntz

MUI is a system to generate and maintain graphical user interfaces. With  
the aid of a preferences program, the user of an application has the  
ability to customize the outfit according to his personal taste.

MUI is distributed as shareware. To obtain a complete package containing  
lots of examples and more information about registration please look for  
a file called "muiXXusr.lha" (XX means the latest version number) on  
your local bulletin boards or on public domain disks.

If you want to register directly, feel free to send

DM 30.- or US\$ 20.-

to

Stefan Stuntz  
Eduard-Spranger-Straße 7  
80935 München  
GERMANY

## 1.9 Credits

Credits  
=====

---

Special thanks goes to:

Massimo Tofani  
Alessio Cappelli  
Ole E. Petterson  
for their postcards and suggestions and  
Nigel Critten  
for the description of the LightWave format.

Also thanks to Mike Hesser for help and motivation and to Stephan Dorenkamp for testing.

## 1.10 Author

Author  
=====

For bug reports, comments, suggestions ... you can contact me at the following address.

S-mail: Andreas Heumann  
Heilmeyersteige 105  
89075 Ulm  
Germany

E-mail: heumann@hugo.rz.fh-ulm.de

PS: I've found out that most people who are using ShowObj don't send me a postcard. I don't know, perhaps they have no time or money. I think it's not expensive to buy a postcard, put a stamp on it and send it to me. Or will it be better to release the next version as shareware?

PPS: It's not a bug, if ShowObj crashes with complex objects, try to use a large stack (20000 bytes).

## 1.11 History

History  
=====

07.06.1994 first public release (1.0)  
12.09.1994 added asl filerequester (1.01)  
03.10.1994 added solid mode (1.02)  
09.11.1994 added perspective mode (1.03)  
02.12.1994 added z-buffer (YEAH!!) (1.1)  
15.12.1994 z-buffer now works with stripes (1.11)  
16.12.1994 added perfect sphere and ground (with SHP2- and SHAP-Chunk)  
18.12.1994 some minor changes, cleanup (2.0)  
26.12.1994 fixed bug in perspective mode (problems with large objects)  
moving objects with mouse in FRONT-,RIGHT- and TOP-view is now possible  
31.12.1994 added LightWave support (2.1)  
02.01.1995 solid mode is now more than twice as fast

---

little speedup in perspective wireframe and boundingbox mode (10%)  
 02.01.1995 object is now automatically placed in the middle of the screen  
 05.01.1995 little bugfix; test if object outside was wrong  
 11.01.1995 added rectangles for lightwave objects  
 object is now automatically zoomed to fit best on screen  
 30.01.1995 bugfix in viewerstrans -> see there  
 31.01.1995 bugfixes  
 - wrong coordinatesystem for Lightwave-objects  
 - wrong automatic zooming in perspective  
 - added lineclipping (I hoped OS does this for me :-( )  
 25.02.1995 compiled with Maxon C++ 3.0 (should be faster now)  
 06.03.1995 added support of tooltypes and workbenchusage (2.11)  
 26.05.1995 added MUI tool window, cleaned source (2.12)  
 12.06.1995 added browser (2.13)  
 22.06.1995 added color and grey solid mode (2.2)

## 1.12 Future

Planned for the future

=====

- use {"RayStorm" link RayStorm} to render scene
- ARexx Port
- save picture as IFF-file

## 1.13 RayStorm

R A Y S T O R M

=====

RayStorm is a fast raytracing-package, which can be controlled with a ARexx-port. RayStorm is almost compatible to Imagine, only faster.

RayStorm has been written to be as fast as possible, and use as less memory as possible. Thus we have implemented a octree algorithm, and optimized all calculations as much as we could.

Features

-----

- Fast. About 20% faster than Imagine.
- ARexx-port. RayStorm can be used by all programs with the ARexx-port.
- Imagine compatible. RayStorm is designed to be almost compatible to Imagine. It can load Imagine objects and use Imagine textures.
- Octree algorithm used for rendering.
- Antialiasing possibility (adaptive supersampling).
- Image can be saved as 24Bit IFF-ILBM file.

- Three builtin object types: sphere, plane and triangle.
  - Three light types: ambient, point and spot.
-