VMD Installation Guide

A. Dalke, P. Grayson, J. Gullingsrud, W. Humphrey, B. Isralewitz, S. Izrailev, D. Norris, J. Stone, J. Ulrich

Version 1.7.1

December 22, 2001

Theoretical Biophysics Group[1]
University of Illinois and Beckman Institute
405 N. Mathews
Urbana, IL 61801
VMD WWW home page: `http://www.ks.uiuc.edu/Research/vmd/`

Description

This document describes how to install one of the precompiled releases of VMD and contains instructions on how to compile VMD from the source code release.

[1] http://www.ks.uiuc.edu/

1 Credits and Program Reference

The authors request that any published work or images created using VMD include the following reference:

Humphrey, W., Dalke, A. and Schulten, K., "VMD - Visual Molecular Dynamics" *J. Molec. Graphics* **1996**, *14.1*, 33-38.

VMD has been developed by the Theoretical Biophysics group at the University of Illinois and the Beckman Institute. The main authors of VMD are A. Dalke, P. Grayson, J. Gullingsrud, W. Humphrey, B. Isralewitz, S. Izrailev, D. Norris, J. Stone, J. Ulrich. This work is supported by grants from the National Institutes of Health (grant number PHS 5 P41 RR05969-04), the National Science Foundation (grant number BIR-9423827 EQ), and the Roy J. Carver Charitable Trust.

2 Copyright and Disclaimer

The terms for using, copying, modifying, and distributing VMD are specified by the VMD License. The license agreement is distributed with VMD in the file LICENSE. If for any reason you do not have this file in your distribution, it can be downloaded from:
`http://www.ks.uiuc.edu/Research/vmd/current/LICENSE.html`
Some of the code and executables used by VMD have their own usage restrictions:

- STRIDE
  STRIDE, the program used for secondary structure calculation, is free to both academic and commercial sites provided that STRIDE will not be a part of a package sold for money. The use of STRIDE in commercial packages is not allowed without a prior written commercial license agreement. See `http://www.embl-heidelberg.de/argos/stride/stride_info.html`

- SURF
  The source code for SURF is copyrighted by the original author, Amitabh Varshney, and the University of North Carolina at Chapel Hill. Permission to use, copy, modify, and distribute this software and its documentation for educational, research, and non-profit purposes is hereby granted, provided this notice, all the source files, and the name(s) of the original author(s) appear in all such copies.
  BECAUSE THE CODE IS PROVIDED FREE OF CHARGE, IT IS PROVIDED "AS IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.
  This software was developed and is made available for public use with the support of the National Institutes of Health, National Center for Research Resources under grant RR02170.
  See `ftp://ftp.cs.unc.edu/pub/projects/GRIP/SURF/surf.tar.Z`

- url_get
  The Perl script url_get, was written by Jack Lund at the University of Texas as Austin. There appear to be no restrictions on its use.

- Python
  Python is made available subject to the terms and conditions in CNRI's License Agreement. This Agreement together with Python may be located on the Internet using the following unique, persistent identifier (known as a handle): 1895.22/1012. This Agreement may also be obtained from a proxy server on the Internet using the following URL: `http://hdl.handle.net/1895.22/1012`

- PCRE
  The Perl Compatible Regular Expressions (PCRE) library used in VMD was written by Philip Hazel and is Copyright (c) 1997-1999 University of Cambridge.
  Permission is granted to anyone to use this software for any purpose on any computer system, and to redistribute it freely, subject to the following restrictions:
  1. This software is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
  2. The origin of this software must not be misrepresented, either by explicit claim or by omission.
  3. Altered versions must be plainly marked as such, and must not be misrepresented as being the original software.
  4. If PCRE is embedded in any software that is released under the GNU General Purpose Licence (GPL), then the terms of that licence shall supersede any condition above with which it is incompatible.

- Tachyon
  The Tachyon ray tracing system distributed with VMD was written by John E. Stone and is Copyright (c) 1994-2001 by John E. Stone. Please see the current distribution of Tachyon for redistribution and or licensing information outside of VMD use. Permission is granted to use Tachyon freely along with VMD.

3 Registering VMD

VMD is made available free of charge for all interested end-users of the software (but please see the Copyright and Disclaimer notices). Redistribution of the software to third parties may require a special license, please check the current VMD license agreement for details. We would like to request that you register with us that you are using VMD. This is so that we can maintain some idea of the number of users of the program and so that we know who to contact about program updates, bug fixes, etc. Registration is now part of our software download procedure, so once you've filled out the forms on the VMD download area you are finished.

4 Acknowledgments

The authors would particularly like to thank those individuals who have contributed improvements to VMD in the form of new features or entire replacement codes for old features. Special thanks go to Andrew Dalke, Paul Grayson, and Charles Schwieters for their VMD contributions. The entire VMD user community now benefits from your contributions.

The authors would like to thank the members of the Theoretical Biophysics group, past and present, who have helped tremendously in making suggestions, pushing for new features, and trying out bug-ridden code. Thanks go to Alexander Balaeff, Daniel Barsky, Tom Bishop, Ivo Hofacker, Xiche Hu, Barry Israeliwitz, Dorina Kosztin, Ilya Logunov, Jim Phillips, Ari Shinozaki, Svilen Tzonev, Willy Wriggers, Dong Xu, Feng Zhou. Thanks also to all of you who have tried out the program.

Many external libraries and packages are used in VMD, and the program would not be possible without them. The authors wish to thank Jon Leech for the code to compute the uniform point distributions; John Ousterhout and the other authors of the Tcl and Tk packages; the authors of the VRPN library from the University of North Carolina; Amitabh Varshney, author of SURF, also from UNC; Dmitrij Frishman at EMBL for developing STRIDE; Jack Lund for the url_get perl script; and Ethan Merrit from the University of Washington for developing the algorithm for drawing ribbons.

We also received invaluable assistance from people who got the source code and sent in patches and explicit bug reports. The VMD developers would like to thank Axel Berg, Andrew Dalke, Rick Kufrin, Joe Landman, Clare Macrae, Lukasz Salwinski, Stephen Searle, Charles Schwieters, Michael Tiemann, Raymond de Vries, and Simon Warfield for their bug fixes and correspondence.

5 Obtaining VMD Source and Binary Distributions

The VMD source code and binary distributions can be obtained after registering at the VMD web page. Download the appropriate distribution file with your web browser. For Windows binary distributions are self extracting, so once the distribution file is downloaded, proceed to the installation directions below.

For source distributions and Unix binary distributions, uncompress and untar the file. This will produce a subdirectory named `vmd-1.7.1`. Unless otherwise specified, all references to VMD code will be from this subdirectory, so `cd` there.

6 Installing a Pre-Compiled Version of VMD

To install the pre-compiled Windows version of VMD, simply run the self-extracting executable, and it will start the VMD Windows installer program, which includes built-in help.

To install the pre-compiled Unix version of VMD, then only three steps remain to be done after you uncompress and untar the distribution.

- Edit the `configure` script. If necessary, change the following values:

```
   $install_bin_dir

This is the location of the startup script 'vmd'.  It should
be located in the path of users interested in running VMD.

   $install_library_dir

This is the location of all other VMD files.  This includes
the binary and helper scripts.  It should not be in the path.
```

- Next generate the Makefile based on these configuration variables. This is done by running `./configure` .
- After configuration is complete, `cd` to the `src` directory and type `make install`. This will put the code in the two directories listed above. After this, you just type `vmd` to begin, provided that `vmd` is in your path.

7 Customizing VMD Startup

The Unix version of VMD reads in several data files (if they exist) when it starts up. These files control the initial appearance and behavior of VMD at the start, and may be customized to suit each users particular tastes. Default versions of these files are placed in the `INSTALLLIBDIR` directory (usually `/usr/local/lib/vmd`). While each user may specify to use different versions of these files, unless this is done the commands and values in the default files are used. In this way, an administrator may customize the default behavior of VMD for all users, while allowing each user the option to change the default behavior however they choose. This chapter describes each of these data files.

Several configurable parameters may also be set in a number of ways, including by command-line options or by environment variables. The order of precedence of these methods is as follows (highest precedence to lowest):

1. Command-line options (see the Users Guide).
2. Environment variable settings (see the Users Guide).
3. Built-in defaults, as specified by compilation configurable parameters. These are used only if no other values are specified by the other methods mentioned in this list.

8 The `.vmdrc` and `vmd.rc` files

After initialization is complete, VMD reads the *startup* file. This file contains text commands for VMD to execute, just as if they had been entered at the VMD text console command prompt. The file can contain any number of commands, including blank lines and comment lines (which begin with the `#` character). If an error is encountered while reading this file, the command in error is skipped and processing of the file continues.

The base filename for this startup file is `.vmdrc` by default on Unix systems and `vmd.rc` on Windows; this is determined by the configuration parameter `STARTUPFILENAME`. VMD searches for this file in a number of locations, and reads in the *first* version of the file it finds. The order of searching for the file is:

1. `./STARTUPFILENAME`
2. `$HOME/STARTUPFILENAME`
3. `INSTALLLIBDIR/STARTUPFILENAME`

See the Users Guide for a list of all VMD text commands.

9 The `.vmdsensors` file

If VMD is compiled with the VRPN option, the file `.vmdsensors` is used to specify how to access external spatial tracking devices. This file will be read when VMD is told to initialize a specific external device. The Tracker library will look for the file in the following directories in the order given, and will use the first one found:

1. `$HOME/.vmdsensors`
2. `INSTALLLIBDIR/.vmdsensors`

This file contains extensive comments on how to configure the file properly. There is no configuration parameter to change the name of the file. If the VRPN option is omitted when compiling VMD, this file is not necessary.

10 What to Do If It Doesn't Work

If you are running a VMD binary which has been built with a native OpenGL implementation (i.e. not Mesa), you should make sure that you have the vendor provided OpenGL runtime libraries and the X server extensions correctly installed. SGI systems normally have the OpenGL runtime support installed on them. Sun, HP, and IBM systems often do not come with OpenGL support by default. If you don't have the OpenGL runtime libraries for these systems, they can be downloaded for free from the Sun, HP, and IBM web sites respectively. Each of the vendor's OpenGL implementations generally include "install check" programs which verify the correct installation and operation of the OpenGL libraries.

- Sun's OpenGL WWW site is at `http://www.sun.com/solaris/opengl/`
- HP's OpenGL WWW site is at `http://www.hp.com/unixwork/products/opengl.html`
- IBM's OpenGL WWW site is at `http://www.austin.ibm.com/software/OpenGL/`

We suggest that you check that you are doing everything correctly, and if it still doesn't work, report the problem by e-mail to `vmd@ks.uiuc.edu`. We'll try to help you.

11 Compiling Your Own Version of VMD

If for some reason you want to recompile VMD, then you will need to read the rest of this document. Most users will want to use the binary distributions we provide since they have been thorougly tested prior to release. It may be necessary for you to compile your own version of VMD in cases where we do not provide a binary for your platform, or when the provided binaries will not run correctly with a particular version of your operating system.

12 Working with the Configure Script

You must configure the compile time options you wish to use. These are set with the `./configure` script which produces Makefiles for the `src` and `$ARCH` directories. After doing the configuration you will do the make. The configuration can be set in two ways:

- On the command line, as in:

  ```
  ./configure IRIX6 OPENGL TK FLTK
  ```

  This will save the options to the file `./configure.options` so the next time you want to regenerate the Makefiles, you need but do

  ```
  ./configure
  ```

- Alternatively, you may edit the file `./configure.options` yourself. You may wish to keep your settings in several configure.options files if you intend to compile VMD on multiple platforms. This is a convenience which can save time when you want to recompile VMD more than once.

  For an SGI version one would do something like:

```
cp ./configure.options.IRIX5.opengl ./configure.options
./configure
```

Several configure.options files are included in the VMD distribution.

After you've set the options in the file, run `./configure` to propagate the new definitions to the Makefiles! The full list of compile-time options set by the configure script are:

- Which OS are you running? VMD can be compiled under several operating systems. The configure time options for these OSs are:

```
  Option:    Tested with:
       AIX4 -- IBM AIX 4.x
     HPUX10 -- HPUX 10.20
     HPUX11 -- HPUX 11.0
      IRIX6 -- SGI IRIX 6.5.4
   IRIX6_64 -- SGI IRIX 6.5.4 (64-bit)
      LINUX -- Linux
 LINUXALPHA -- Linux Alpha
   LINUXPPC -- Linux PowerPC
     MACOSX -- MacOS X with XFree86
   SOLARIS2 -- Solaris 8 Sparc
SOLARIS2_64 -- Solaris 8 Sparc (64-bit)
 SOLARISX86 -- Solaris 7 x86
      TRU64 -- Tru64 Unix 4.0 (64-bit)
      WIN32 -- Windows
```

- Which type of display graphics will you use? There are four options which will be carefully explained below:

```
    Option:    Graphics Support:
     OPENGL -- OpenGL graphics library (overall best choice)
       MESA -- Free, Portable, Open Source, OpenGL workalike
 NOGRAPHICS -- Graphics support disabled
```

  OPENGL – VMD compiles with native OpenGL for DEC, HP, IBM, Sun, and SGI. OpenGL offers the best rendering performance on new systems, and is best supported graphics option for VMD at this time. Contact your vendor or support service to get OpenGL for your system.

  MESA – Mesa is a free OpenGL compatible library which can be used on systems which have no native OpenGL implementation. Mesa normally performs all of its rendering in software, and may run significantly slower than a native OpenGL implementation. On systems using XFree86 4.0 or later, the new DRI interface can provide hardware-accelerated Mesa, which will speed up VMD significantly. With few exceptions, a native OpenGL implementation will provide the best rendering performance. We have compiled VMD with MESA 3.1 on all of the tested OSs. The Mesa library is available for free via ftp from `ftp.mesa3d.org` in the `mesa` directory. Mesa is also mirrored on sunsite in the directory `pub/packages/development/graphics/mesa`. See the web address http://www.mesa3d.org/ for more information, and the README in ./lib/Mesa for information about how to get Mesa working with VMD.

- VMD can be compiled to run in the CAVE. If you don't know what the CAVE is, see http://evlweb.eecs.uic.edu/EVL/VR/ . To compile in the CAVE you will need to get the appropriate CAVE library. VMD now only supports OpenGL, so you need to use libcave_ogl.a.

  To get the CAVE library, please contact Pyramid Systems, Inc. The Pyramid WWW page is at http://www.pyramidsystems.com/. We cannot distribute the CAVE libraries, but the standard SGI binary distribution version is compiled with the CAVE library. To use that option, start VMD with the run-time option -dispdev cave.

  The configuration options are:

  ```
  Option:    Graphics Support:
     CAVE -- include run-time support for CAVE display (OpenGL).
  ```

- Which kind of GUI do you want to use? (The GUI is nearly independent from the graphics display method.) They are:

  ```
  Option:    GUI Support:
     FLTK -- the standard FLTK GUI
       TK -- Support for user-written GUI elements in scripts etc.
  ```

  You should compile with FLTK, in which case you need to get the source or binaries from http://www.fltk.org/.

  VMD was compiled against Tcl/Tk version 8.0.4. You must have the 8.0 or better versions of both packages. The latest versions can be found at:

  ```
  Tcl/Tk 8.0.4 is at http://www.scriptics.com/
  ```

  Because Tcl and Tk are very useful tools, we make the assumption that these will be installed on your system, so there is no ./lib/tcl directory. To configure VMD to use the appropriate Tcl/Tk directory, you may have to edit ./configure and change a few variables. The current settings look in /usr/local/[include,lib] unless the corresponding environment variable is set. The variables are:

  From the configure script:

  ```
  ################ Tcl / Tk
  # location of TCL library and include file.
  # If left blank, standard system  directories will be searched.
  $stock_tcl_include_dir=$ENV{"TCL_INCLUDE_DIR"} || "/usr/local/include";
  $stock_tcl_library_dir=$ENV{"TCL_LIBRARY_DIR"} || "/usr/local/lib";

  # location of Tk (for TK option)
  $stock_tk_include_dir=$ENV{"TK_INCLUDE_DIR"} || "/usr/local/include";
  $stock_tk_library_dir=$ENV{"TK_LIBRARY_DIR"} || "/usr/local/lib";
  ```

Either change the default values or override them with the environment variables.

- VMD uses Amitabh Varshney's SURF program to compute the molecular surface. This is actually an external program that VMD calls; it is not compiled into VMD. You will need to compile SURF yourself to use option. The configuration options are:

```
SURF -- add the external call to SURF (uses DrawMolItemSurf.\{Ch\}
        and adds the option to the graphics pop-up list)
```

  There is no reason not to compile SURF. It is actually distributed (with permission) as part of the VMD distribution. To compile surf, go to `./lib/surf` and follow the instructions in `README.VMD` (the `README` file if from surf itself). The end result will be a program named `surf_$ARCH` in that directory. Leave it there.

  For the main SURF distribution, see
  `ftp://ftp.cs.unc.edu/pub/projects/GRIP/SURF/surf.tar.Z`

- In addition to the CAVE library, VMD can use the VRPN library to get information about various 3DOF and 6DOF spatial trackers and other input devices. Our standard VMD builds now enable VRPN by default on most platforms. The configuration options for using VRPN input devices are:

```
VRPN -- Use the VRPN library to support local and remote trackers
        and other input devices.
```

  For more information about VRPN see
  `http://www.cs.unc.edu/Research/vrpn`

- VMD can connect to NAMD to perform interactive simulations; the protocols related to this connection are referred to as Interactive Molecular Dynamics (IMD). The configuration option for IMD is enabled by default:

```
IMD -- use IMD for doing remote simulations
```

- There are several compilation flags for VMD. They are:

```
DEBUG or NODEBUG
```

  When turned on, these print all "msgdebug" messages. There are a lot of them and they really slow things down. (BTW, you can tell who wrote which sections of code by who uses msgDebug :) Also, set the -g flag when compiling. By default, VMD is compiled with NODEBUG. If you only want to add the -g flag, edit the configure script so the compiler is defined as "CC -g" (or "g++ -g").

```
SILENT or NOSILENT
```

  The default is SILENT, in which case you don't see the details of the compilation as it suppresses the make command echo. All this does is add or not add the line ".SILENT" near the top of the produced Makefile.

13 Building Libraries for VMD

Now, a detour. There are few libraries you will have to build. These are part of the standard VMD distribution but not part of VMD proper. They are:

13.1 STRIDE

STRIDE is used by VMD to compute the secondary structure given the protein 3D coordinates. The appropriate STRIDE binary is included in the VMD binary distribution. To compile it yourself, see the web site at
`http://www.embl-heidelberg.de/stride/stride_info.html` for information on how to get the source and see `./lib/stride` for information on how to use it with VMD.

Change line 43 of `stride.h` from

```
#define MAX_AT_IN_RES            50
```

to

```
#define MAX_AT_IN_RES            75
```

because there are many structures with non-standard residues containing more than 50 atoms.

Change line 96 of `stride.c` from

```
return(SUCCESS);
```

to

```
return(0);
```

since a program should return 0 if everything ended correctly.

13.2 Babel

Babel is used to convert different file formats automatically into PDB files(s) before reading them into VMD. Babel is a freely available general purpose molecular file conversion program. See the web site `http://www.eyesopen.com/babel.html` for more information about it. This program must be available on the path somewhere or you can set the BABEL_BIN environment to point directly to the binary.

13.3 Renderers

VMD supports many types of rendering formats. You may want to get the packages associated with them. They are (among others) Raster3D, Radiance, POV-Ray, and Tachyon. Information about these and other programs can be found at:
`http://www/Research/vmd/allversions/supported_output.html`.

13.4 xmgr

The Labels form allows you to call an external program to graph a selected geometry value over time. The default program, which we suggest using, is xmgr, available from
`ftp://plasma-gate.weizmann.ac.il/pub/xmgr4/`

14 Working with the VMD Source Code

Okay, so you've set the compile time options and run `./configure`, right? The next step would be to check out the latest versions of the files from RCS or CVS, but odds are you don't have your copy of VMD under revision control. In the future we intend to provide an anonymous CVS service to allow VMD users to track developments in the VMD source tree.

Note that the configure command also makes the subdirectory `$ARCH`.

Next you'll need to make the dependency list INSIDE THE SRC DIRECTORY. This is done with

```
  cd ./src
  make depend
  make
```

However, the current implementation needs `gcc` to get the list of include files needed by the different source files. The default Makedata.depend files for the different platforms is available in the appropriate binary distribution. After the dependencies are generated, run `make`, and the executable and object files will be deposited in the `$ARCH` directory

```
  cd ../$ARCH
(for example
  ../LINUX
  ../IRIX5
etc.)
```

This directory was made when you ran the `configure` script.