

2 Installation

This section contains build instructions for the OVIS release. You will be required to obtain and install supporting software that is not part of OVIS in a manner appropriate to your platform. Necessary supporting software is listed below. System settings for running OVIS are also given.

2.1 Hardware Requirements

These requirements apply to computers that will run the OVIS Baron GUI.

1. A Core 2 Duo or newer CPU and 2 GB or more RAM are recommended.
2. An NVIDIA graphics card, preferably a GeForce 8800 or newer or a Quadro FX (not NVS), is recommended.

2.2 Supporting Software

1. If possible, use a recommended operating system: Fedora 9 or newer, or Red Hat Enterprise Linux 5 or newer. Mac OS X can also be used.
2. Obtain and install a C compiler, a C++ compiler, and OpenGL version 1.2 or newer. These are probably already installed on your system.

Regarding OpenGL: Video cards with NVIDIA chipsets for 3-D rendering combined with the NVIDIA-provided, closed-source OpenGL drivers are all we test and support, given the lack of features and/or stability other solutions currently provide. The `akmod-nvidia` package from `rpmfusion.net` is an alternative to manually installing NVIDIA drivers on Fedora and other RPM-based Linux distributions. Installing the `xorg-x11-drv-nvidia-libs` and `xorg-x11-drv-nvidia-devel` packages provides the remaining elements for OpenGL support.

3. Obtain and install database software – either MySQL or PostgreSQL will work. As of this writing the current RPMs for Fedora will work.

MySQL:

- (a) Obtain and install MySQL at least version 5.0.77 (install client, server, devel, and shared)
- (b) Obtain and install MySQL-python

PostgreSQL:

- (a) Obtain and install PostgreSQL at least version 8.2.9
- (b) Set the following in your environment:

```
setenv PGSQL_HOME /usr/local/pgsql
set path=($PGSQL_HOME/bin \ $path)
```

4. Obtain and install Qt using the latest stable version of 4.6.x. As of this writing the current RPM for Fedora will work (Qt version 4.6.x). You will need the qt4-devel RPM and the RPM for the appropriate database plugin as well.
5. Obtain and install CMake at least version 2.8.1 from <http://www.cmake.org/>
6. Obtain and install Boost at least version 1.42.0 from <http://www.boost.org/>
7. You must provide a multicast domain name service (mDNS) service discovery (SD) daemon. This may be either Bonjour or Avahi. If you plan to use Avahi,
 - Obtain and install libdaemon.
 - Obtain and install dbus.
 - Obtain and install Avahi at least version 0.6.22. This may require you to get intltool for the install. On 64-bit systems, you will need version 0.6.23 or newer. Whatever version of Avahi you use, you must build the avahi-qt4 library if you plan to build the OVIS Baron GUI.
 - Build Avahi and install it as a system service on all nodes which will run any OVIS software. Configure Avahi as follows:

```
setenv PKG_CONFIG_PATH /usr/local/lib/pkgconfig:$QTDIR/lib/pkgconfig
./configure --disable-python-dbus --disable-mono --disable-python \
--disable-gtk --disable-qt3
```

- If you use the RPM's install avahi-devel as well.

If you plan to use Bonjour,

- Obtain and install mDNSResponder on all nodes which will run any OVIS software.

2.3 OVIS Install

1. Obtain the OVIS source from ovis.ca.sandia.gov.
2. `mkdir /path/to/OVISBuildDir`
3. `cd /path/to/OVISBuildDir`
4. `ccmake /path/to/OVISSrcDir`
5. Set the following variables within ccmake to the values shown. (Type 'c' to configure and then 't' to see these options; ignore initial warnings.)

```
CMAKE_BUILD_TYPE: Debug
```

`OPENGL_INCLUDE_DIR`: Location of NVIDIA headers (e.g., `/usr/lib64/nvidia`)

`OPENGL_gl_LIBRARY`: Location of NVIDIA OpenGL library (e.g., `/usr/lib64/nvidia/libGL.so`)

6. Type 'c' to configure again until you have a 'g' to generate option.
 7. Type 'g' to generate.
 8. make
 9. make install
- # optional. Running as root may be needed

There is an additional parameter, `OVIS_SQL_FP_RTOL`, relating to the analyses, that can be set in `cmake`. It is described further in Section 7.

The VTK source is included in the OVIS source, and VTK will get built as part of the OVIS build.

You are now done building OVIS 3.2!

Note that OVIS will place the plain text configuration file `${HOME}/.config/Sandia/ovis.conf` on your system. This will contain Baron state information (as described in §8) and database usernames and passwords should you choose to explicitly save them. (The `ServerConnection` window of Figure 17, described in §8 and §9.1, contains the control on whether this information is saved or not.)

2.4 MySQL Settings

1. If MySQL is not currently running (however, it should be if you are running on a system with Fedora 8 or greater installed):

```
/sbin/chkconfig --level 345 mysqld on # run MySQL daemon at boot
/sbin/service mysqld start # run MySQL daemon now
```

2. You should create a database user named `ovis`. This user will need full administrative privileges on the local machine and the ability to alter tables and insert records from machines where data will be collected. The administrative privileges are required to load a shared library (`libovis-mysql.so`) that contains functions used to signal the OVIS Shepherd process when rows in certain tables are inserted or modified. For connections from network interfaces that face outside the cluster, you may require a password for the `ovis` user. Remote connections from external networks will still require permission to insert records into the database in order to request analyses of collected data. For example, if you will be using OVIS to store data into a database called `OVIS_Cluster`:

```

mysql -u root -p
mysql> GRANT ALL PRIVILEGES ON OVIS_Cluster.* TO 'ovis'@'localhost';
mysql> GRANT ALL PRIVILEGES ON OVIS_Cluster.* TO 'ovis'@'localhostsipaddress';
      #similarly for all addresses (including virbr0)
mysql> GRANT INSERT,DELETE,UPDATE,EXECUTE,SELECT ON OVIS_Cluster.* TO
      'ovis'@'someremotehost' IDENTIFIED BY 'somepassword';
mysql> flush privileges;

```

The remotehosts should include that of the nodes running the Sheep §5 that will insert into a given database, and that of all other Shepherds that may be participating in an analysis §7.2.

Note that you should configure mysqld so that no password is required to access the database from the private (administrative) network of your cluster or the local host where mysqld and the OVIS Shepherd process will run. This way, Sheep and Shepherd processes do not need to be configured with any database passwords. You can require a password for the machine that is running the Baron.

3. There is a user defined function that must be loaded into the MySQL database. This will go into the `mysql.func` table. It needs only to be added once – not on a per OVIS_Cluster basis. If you use the database effector described in §4.2, this will occur with the `-t 4` flag. If you load a mysqldump (such as the example one for the Glory database that comes with the release), ideally the function should be loaded as part of this process, however in practice the authors have seen that this may be dependent on the MySQL version. It is thus recommended that you run the database effector at this point. For MySQL, the database effector will run as the user `mysql` and will require that `/libovis-mysql.so`, built during the install be accessible by that user; directories that are so accessible are specified in, possibly, `/etc/ld.so.conf.d/mysql-x86_64.conf`, which may contain the entry `/usr/lib/mysql/plugin`.

- Either

- add the path to `libovis-mysql.so` to `mysql-x86_64.conf`
- `/sbin/ldconfig -v`
to update the paths

or

- add a link for, or copy, `libovis-mysql.so` to an existing accessible directory, such as `/usr/lib/mysql/plugin`
- also, if applicable, use
`chcon`
to set the security context to that of other files that exist there.

- after you run the database effector, check that the functions are indeed present in the `mysql.func` table.

4. Note: Some users have reported that VTK's MySQL interface cannot find `mysql.sock` despite its location being specified in the `/etc/my.cnf` configuration file. If this occurs, do the following:

```
cd /tmp
ln -s /var/lib/mysql/mysql.sock mysql.sock
```

2.5 PostgreSQL Settings

If you wish to use a PostgreSQL database to hold OVIS information, you will need to configure it appropriately.

1. You will need to edit the configuration file `/var/lib/pgsql/data/postgresql.conf` and change the `listen_addresses` setting to allow incoming connections from remote machines (both Sheep inserting measurements of cluster behavior and users connecting with the Baron to perform analysis). Unless you have a reason to specifically avoid a particular network interface, we suggest listening on all interfaces. We also recommend turning off informational messages printed by clients.

```
listen_addresses = '*' # listen on all network interfaces
client_min_messages = warning # print only warnings+errors
```

2. In addition to requesting that the daemon listen on network interfaces, you must specify how authentication should occur in `/var/lib/pgsql/data/pg_hba.conf`. For local connections or from network interfaces on the administrative network of a cluster, you should require no password so that Sheep and Shepherd processes may connect. For connections from network interfaces that face outside the cluster, you may require a password for the ovis user. As an example, consider the following lines:

```
# TYPE DATABASE USER CIDR-ADDRESS METHOD
## Connections on the local machine
local ovis ovis trust
host ovis ovis 127.0.0.1/32 trust
local OVIS_Cluster ovis trust
host OVIS_Cluster ovis 127.0.0.1/32 trust
## Connections on private cluster admin network
host OVIS_Cluster ovis 192.168.1.254/24 trust
## Connections from remote sites. Requires password
host OVIS_Cluster ovis 74.125.19.19/24 ident
```

3. If the PostgreSQL postmaster daemon was running and you changed any of the configuration files above, you should restart it:

```
/sbin/service postgresql restart
```

Otherwise, if the daemon was not currently running, set it to run on reboot and then start it manually:

```
/sbin/chkconfig --level 345 postgresql on # run daemon at boot
/sbin/service postgresql start # run PostgreSQL daemon now
```

4. You should create a database user named `ovis`. This user will need full administrative privileges on the local machine and the ability to alter tables and insert records from machines where data will be collected. The administrative privileges are required to load a shared library (`libovis-psql.so`) that contains functions used to signal the OVIS Shepherd process when rows in certain tables are inserted or modified. For example, if you will be using OVIS to store data into a database called `OVIS_Cluster`:

```
createuser -s -d -l -P ovis # You will be prompted for a password.
createdb -p -U ovis ovis # Enter the password for ovis.
createdb -p -U ovis OVIS_Cluster
```

5. There is a user defined function that must be loaded into the PostgreSQL database. It needs only to be added once – not on a per `OVIS_Cluster` basis. If you use the database effector described in §4.2, this will occur with the `-t 4` flag.

2.6 Additional General System Settings

Finally, many Linux distributions will need some system settings changed, links created, and daemons turned on or off.

1. While it is possible to run the Shepherd process on systems with SELinux enabled, it is beyond the scope of this document to cover all of the configuration issues required. You may wish to configure your Shepherd nodes to run in permissive rather than enforcing mode.
2. Place the following lines in your iptables configuration file (`/etc/sysconfig/iptables` on most systems):

```
# Allow mDNS (also known as Avahi, Zeroconf, Bonjour)
-A INPUT -m state --state NEW -m udp -p udp --dport 5353 \
    -d 224.0.0.251 -j ACCEPT
# OVIS
-A INPUT -m state --state NEW -m udp -p udp --dport 49154 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 53170 -j ACCEPT
```

You may also need to add entries to allow PostgreSQL (port 5432) or MySQL (port 3306) connections, depending on which distribution of Linux you use and which database you prefer.

3. Set symbolic links for libraries.

- If you have root and if you have done `make install` (note in the below that `/path/to/ovisInstallDir` is `/usr/local`):

```
cd /usr/lib64 # if you are on a 64 bit machine, or /usr/lib if you are not
ln -s /path/to/ovisInstallDir/lib64/libovis-mysql.so \
    /usr/lib/mysql/plugin/libovis-mysql.so
# your mysql-plugin directory or similiar, as described earlier
ln -s /path/to/ovisInstallDir/lib64/vtk-5.7/libvtksys.so.5.7 \
    libvtksys.so.5.7
ln -s /path/to/ovisInstallDir/lib64/libovis-mysql.so libovis-mysql.so
ln -s /path/to/ovisInstallDir/lib64/libvtkCommon.so.5.7 \
    libvtkCommon.so.5.7
ln -s /path/to/ovisInstallDir/lib64/libvtkFiltering.so.5.7 \
    libvtkFiltering.so.5.7
ln -s /path/to/ovisInstallDir/lib64/vtk-5.7/libvtksys.so.5.7 \
    libvtksys.so.5.7
```

- Or if you have not done `make install`:
 - do the above *for the mysql-plugin directory only*, replacing `/path/to/ovisInstallDir` with `/path/to/ovisBuildDir`
 - you may also have to add, or copy, `libvtkCommon.so.5.7`, `libvtkFiltering.so.5.7`, and `libvtksys.so.5.7` to the `mysql-plugins` directory as well.
 - put `/path/to/ovisBuildDir` in your `LD_LIBRARY_PATH`.