

# GNNViz User Guide

GNNViz was developed as part of project 05-4-1-12, “Innovative, 3-D, interactive, and immersive techniques for visualizing, querying, and understanding regional maps of forest vegetation, fuels, and fire risk,” funded by the Joint Fire Science Program. For more information and contacts, visit the project website: <http://fsl.orst.edu/lemma/gnnviz>.

## Introduction

GNNViz is a Windows PC application that allows users to explore a virtual representation of large-scale terrain and geospatial datasets, including detailed maps of vegetation and fuels created with the Gradient Nearest Neighbor (GNN) method. Building on computer game concepts, GNNViz creates a 3D, interactive environment where the user can move around and explore data and information. In addition to viewing dataset images draped over the terrain, GNNViz can also display trees on the terrain surface, as controlled by density maps from the GNN predictive model.

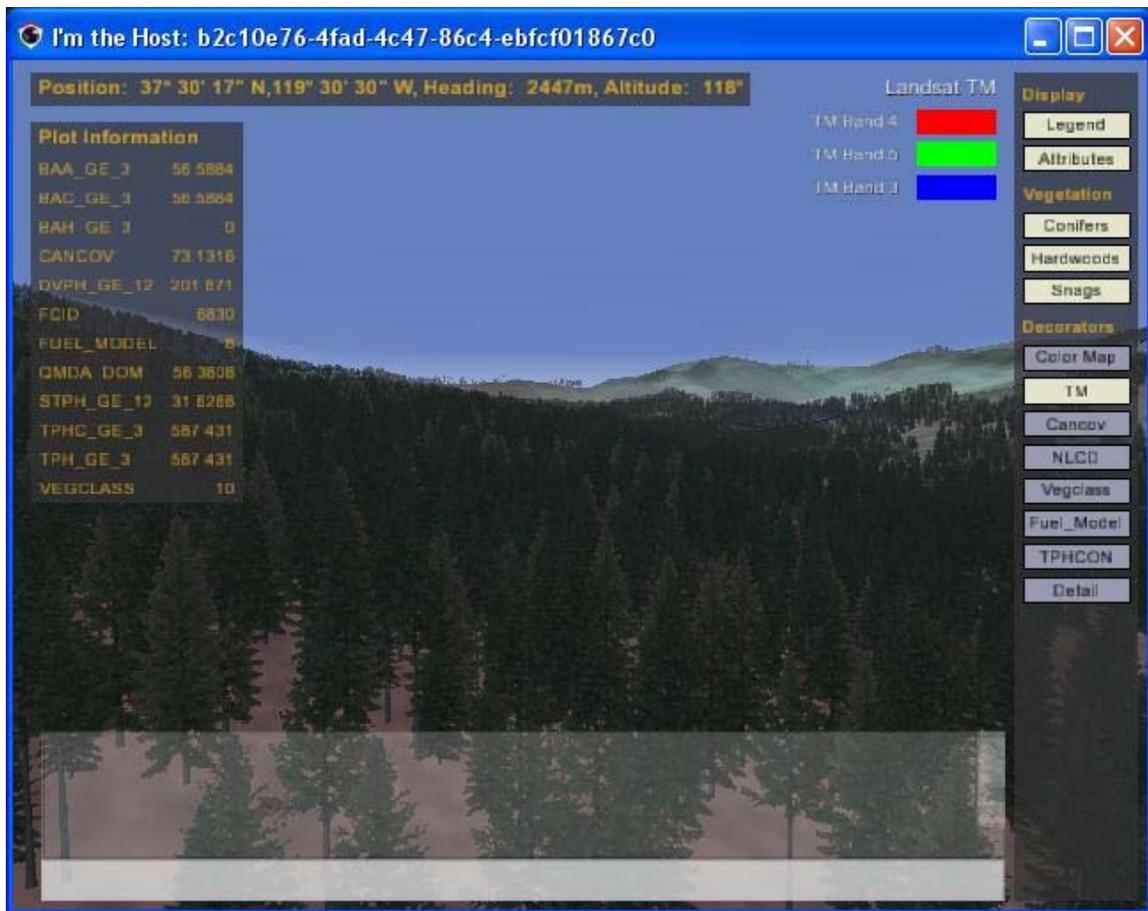


Figure 1: A screenshot of the GNNViz application

Fig. 1 shows a typical GNNViz screen, with trees displayed over a LANDSAT TM image which has been draped over the terrain surface. Selected plot data from the GNN predictive model are being displayed for the viewer's current location.

GNNViz is both a single- and multi-user application. Single users may explore a virtual space alone, or they may choose to either host or join a server where multiple users can explore and interact in the same shared space.

## **System Requirements**

---

GNNViz was developed and tested using medium- to high-performance PC hardware running the Windows XP operating system. The basic functionality of the application should work as expected on the majority of hardware, provided the hardware has OpenGL, audio, and networking devices. Higher-end hardware will give much better performance and result in smoother screen rendering of the 3D environment.

GNNViz was only tested on a Windows PC environment, though theoretically it should compile and run on a Linux machine, since the underlying game engine used (Delta3D) supports multiple platforms. For more information on Delta3D and building Delta3D applications on other platforms, see the Delta3D website at <http://www.delta3d.org/>.

## **Using GNNViz**

---

GNNViz is usually started via the command-line or from a windows shortcut, as a command-line argument is required to reference a Mission XML File that tells GNNViz what regional area and datasets are to be displayed.

To start GNNViz from the command-line...

1. Open up a command-line shell in Windows (Start / Run... / cmd.exe).
2. Change the current working directory to the Delta3d/gnnviz/bin directory.
3. Start the gnnviz.exe program, using the --mission command-line argument to specify the name of the mission XML file to use (Fig. 2).
4. The default installation of GNNViz comes with three predefined mission files: mission\_california.xml, mission\_oregon.xml and mission\_washington.xml.

When GNNViz is started, it will open up a new window for the application, and you will see a variety of informational text displayed in the original command-line window. You may also see warnings and even error messages if some kind of problem occurs such as a file not found, etc. Your initial view and position in GNNViz is controlled by the mission XML file you specified when starting the program. Fig. 3 shows the startup screen for California.

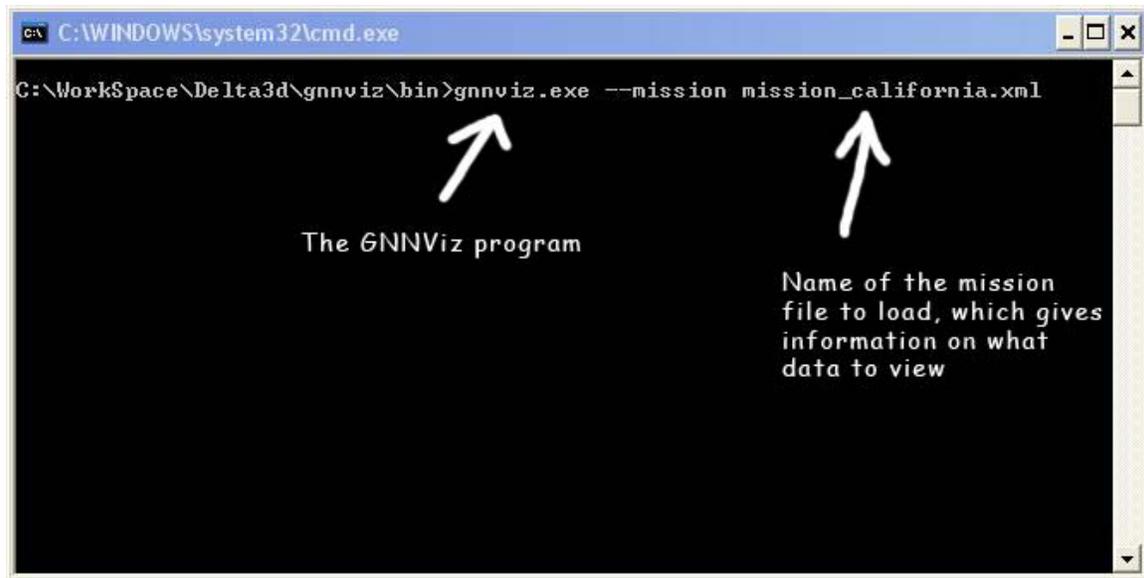


Figure 2: Starting GNNViz with the California mission XML file

Below is a screenshot of the initial view with the California mission file.

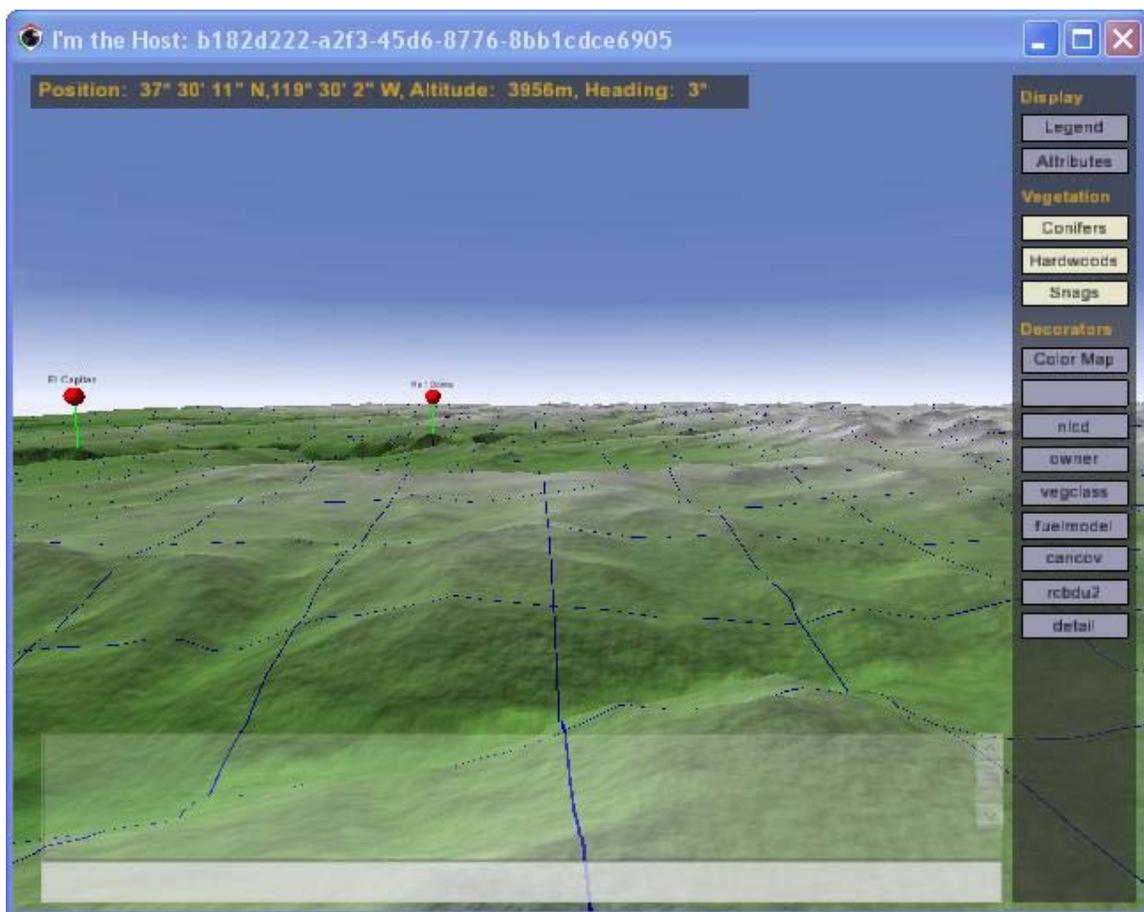


Figure 3: Initial GNNViz display for the California dataset

By default, GNNViz displays a basic color map draped over the terrain. This is a simple coloration of the terrain, with the colors used based on the elevation. There is also a simple grid overlaid on the terrain surface. This grid marks out areas of one-minute spacing with blue lines, and every even degree is marked out with red lines. This grid can help you get a sense of scale for how big of an area you are looking at and how fast you are traveling when moving.

Also shown on the display, at the top of the screen, is your current position, altitude, and heading. Along the right-hand side of the screen is a set of buttons that can be clicked to control what is displayed by the application. Along the bottom of the screen is a console where various status and other messages are displayed, and where text may be entered for chat and to give certain commands.

## Looking and Moving Around in GNNViz

### Basic Navigation

If you have some previous experience playing PC games, you'll probably not have much difficulty using GNNViz. It uses the conventional game "WASD" keys to move (Fig. 4), and the mouse for looking around.

If you're not familiar with PC games, just imagine that in GNNViz you are in a helicopter that can hover and fly to any position. The first thing you need to learn is how to change where you're looking. Click the left mouse button with the cursor in the middle of the screen, and then drag it from side to side and up and down. You should see your view change as you do this. The closer you click to the edges of the GNNViz window, the faster and farther your view will change.

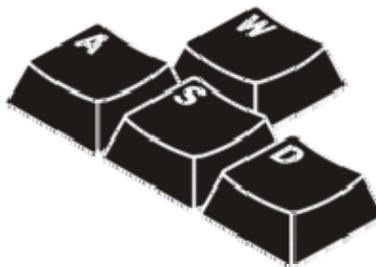


Figure 4: The WASD key layout for movement in GNNViz

Now that you know how to look around, let's cover basic movement. GNNViz uses the W, A, S and D keys (Fig. 4) to control movement. They have a very similar layout to the default four arrow keys present on most PC keyboards, and their actions mimic the four arrow keys at the same positions. Pressing the "W" key will move you forward, "S" will

move you backwards, and “A” and “D” will move you left and right respectively. The actual direction of movement is based on the direction you’re looking, so if you look towards the ground and move forward (pressing “W”), you will fly down towards the ground, and moving backwards (Pressing “S”) will make you fly away from the ground. Don’t worry about flying all the way down to the ground as you won't crash.

## **Moving Faster**

At times the default movement speed of GNNViz may be too slow. As you are moving, you can also press and hold down the “F” key to increase your speed. As long as the “F” key is down, you will move significantly faster than the default speed. There are other shortcuts to movement, described under “Chat and Special Commands” below.

## **Ground Mode**

In ground mode, your view is clamped to the terrain’s surface as if you’re walking or moving along the ground. Unlike fast mode, which is only on when a key is pressed, ground mode is something you can toggle on and off. To toggle into ground mode, press the “G” key. If you’re up in the air when you turn it on, your view will move instantly to the ground.

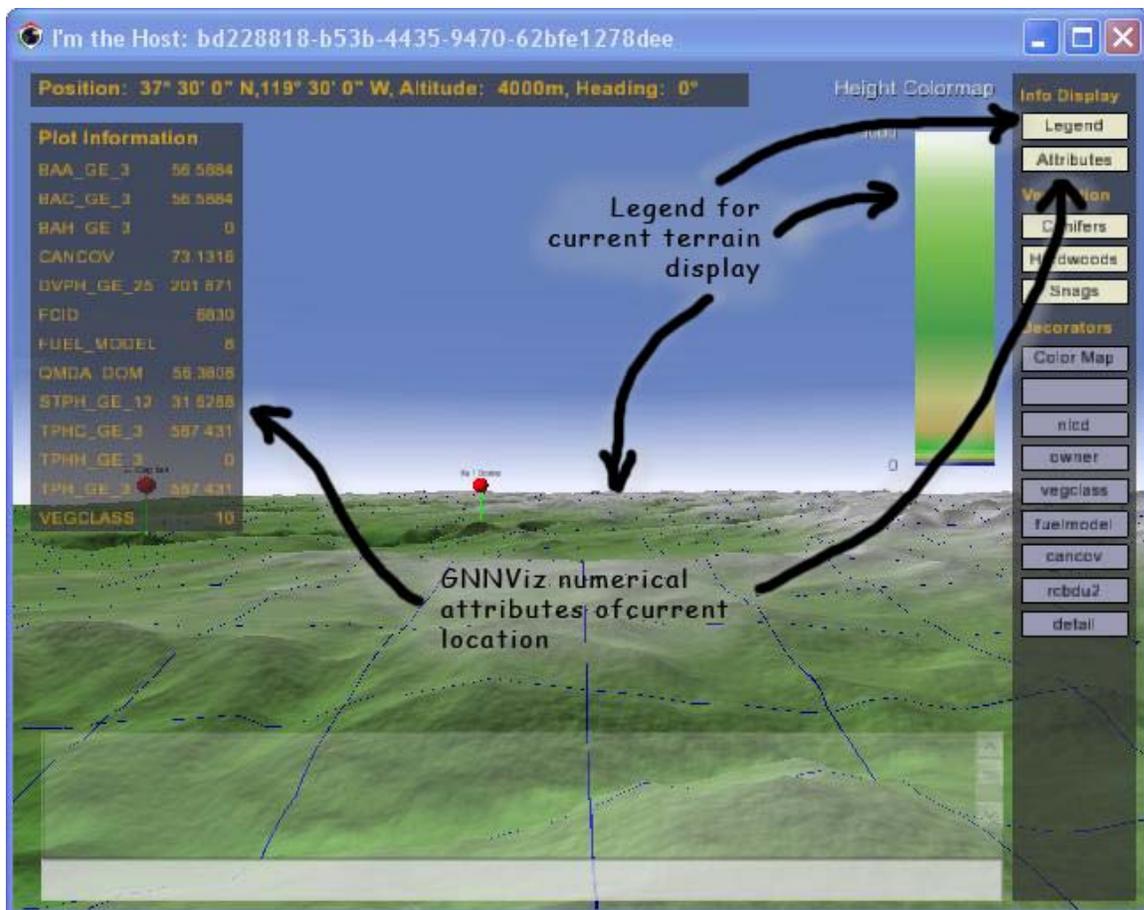
To help keep the camera from inadvertently going under the terrain surface in complex areas, the camera’s height in ground mode has been set to be 4 meters off the terrain. Keep this in mind as you move around, and don’t mistake it for a view as if from a person’s eye, which would be around 1.5 to 2 meters off the ground. Note that fast mode does not work while in ground mode and that the movement speed while in ground mode is significantly slower.

## **Changing the Display**

Now that you’re familiar with moving around in GNNViz, let’s learn how to change what data is displayed on the terrain’s surface as well as other informational displays.

All display options are controlled with a set of grouped buttons along the right-hand side of the application window (Fig. 5). Buttons that are white are considered selected or “on”, buttons in gray are considered unselected or “off”.

The top two buttons, in the “Info Display” group, control display of a data legend and attributes at the current position. By default, both are off. Go ahead and click the “Legend” button first, and see how clicking it turns on and off the display of a map legend for whatever data layer is currently drawn on the terrain’s surface. The “Attribute” display turns on and off the display of numerical data values from the GNN predictive model for your current location (left-hand side of display, Fig. 5). See Appendix 4: Data Dictionary of the GNNViz final report, available from our website (<http://fsl.orst.edu/lemma/gnnviz>) for more information on the vegetation and fuels attributes.



**Figure 5: The GNNViz Legend and Attribute displays**

The next group, “Vegetation,” has three buttons that control the display of the different tree and vegetation types on the terrain’s surface. By default, trees are not initially rendered on the terrain, so you can ignore these buttons for now. Their use is covered in more detail in the “Displaying Trees” section below.

Lastly is the group of “Decorators” buttons to control what decorator is to be drawn on the terrain’s surface. Only one decorator at a time may be displayed on the terrain’s surface, and they vary in type from LANDSAT TM images to attributes like fuel models (Fig. 6) or tree density from the GNN model.

The last decorator on the list (“Detail”) is an algorithmic ground texture designed to give a more realistic visual simulation of the Earth’s surface when viewed from higher altitudes and from the ground. It uses a simple algorithm based on canopy cover to control the type and blend of different textures that are applied to the surface. The textures vary with the distance between the viewer and the surface.

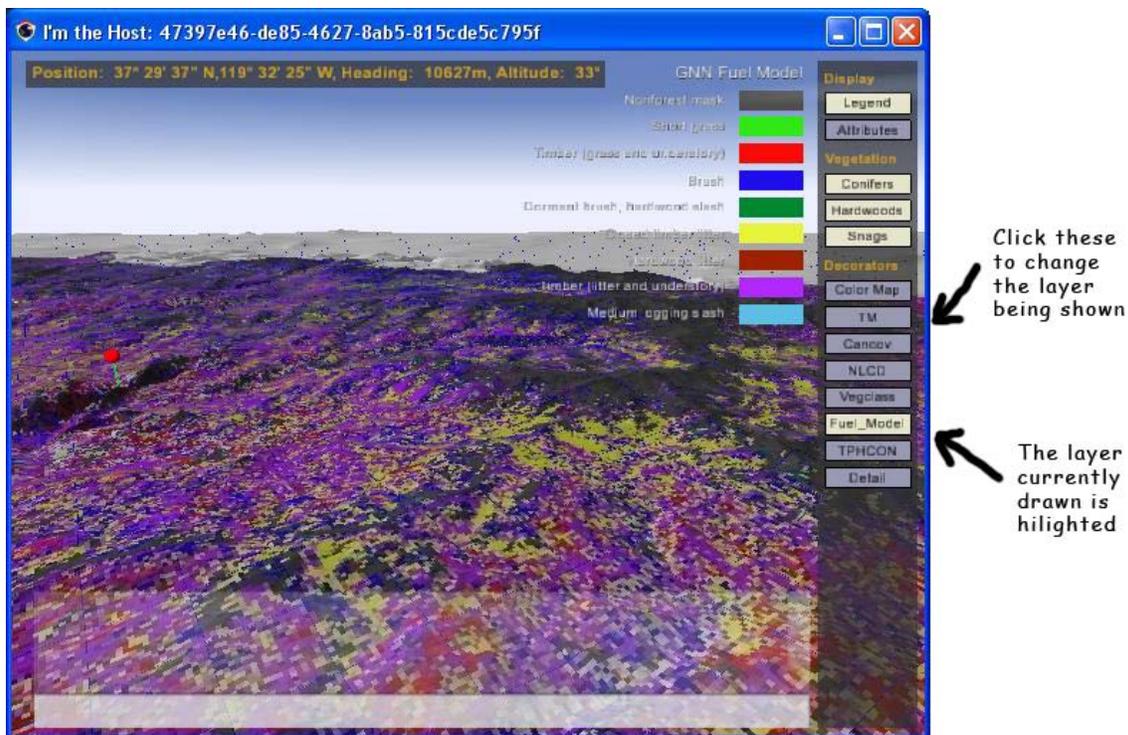


Figure 6: Displaying the Fuel Model data overlaid on the terrain surface

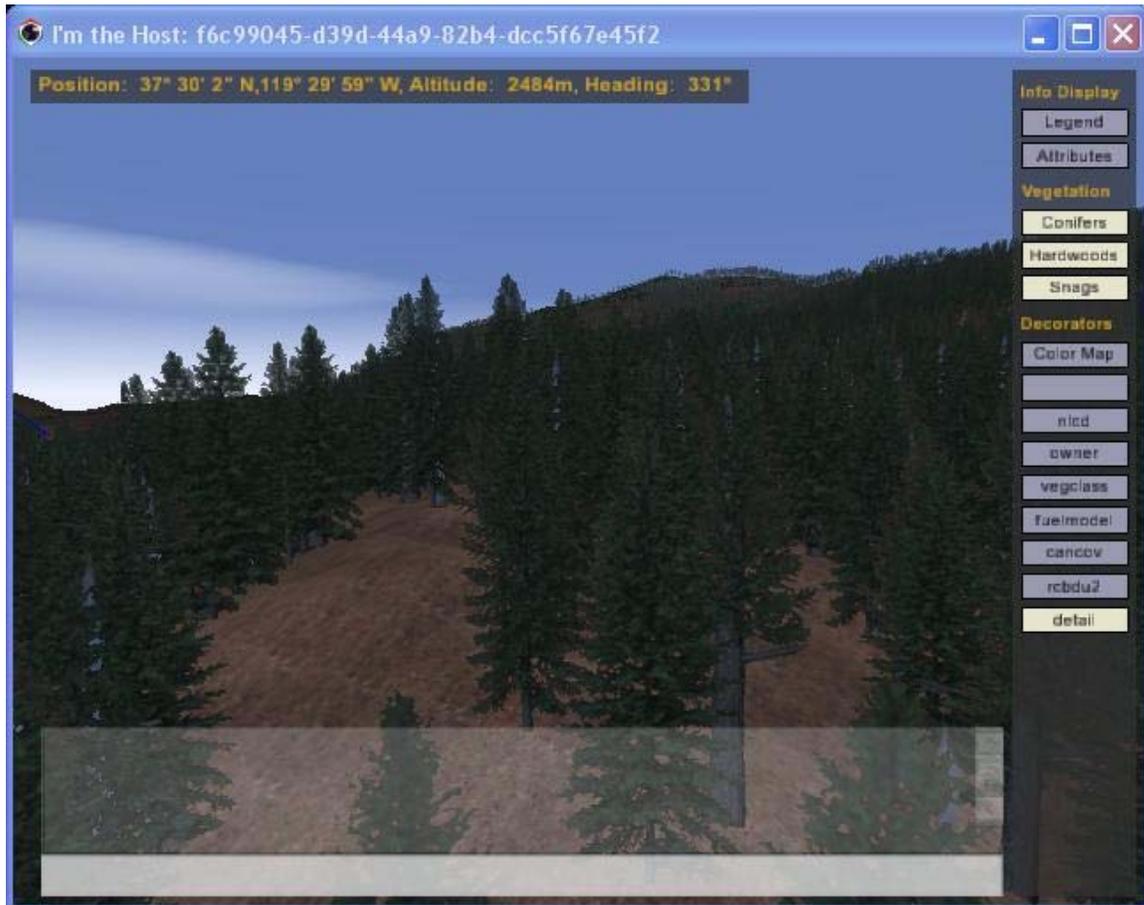
## Displaying Vegetation

Besides drawing terrain with different decorators, GNNViz is also able to draw trees and vegetation on the terrain surface based on data from the GNN predictive vegetation model (Fig. 7). This initial version of GNNViz categorizes vegetation as conifers, hardwoods and snags, using a flat, 2D image to represent the different vegetation types. Although at first the 2D representation may seem simple, it is a surprisingly effective and efficient way to display trees in a virtual environment.

It's important to note that GNNViz does not automatically generate and draw trees on the terrain surface. Generating the tree locations is not that time-consuming, but it does take a sufficient amount of time such that GNNViz can't generate them as you move about across the terrain. GNNViz also does not generate and draw trees over the entire terrain area visible, but rather limits drawing of trees to a selected geographical area that is approximately two miles across.

Why all the limits on when and where trees are placed and drawn? Because the sheer volume of individual items that must be calculated and drawn by GNNViz can get to be so large that the program is unable to run smoothly. When overloaded, the frame rate (rate at which the program is able to redraw the screen) can fall so low that the program becomes unresponsive and difficult to use.

To actually generate and display trees, you will need to press the “B” key on your keyboard. The positions of trees will be generated in an area centered on the point directly below your point of view, covering an area approximately two miles across. Note that if you are high up in the air, you may not actually see the trees, and you’ll need to fly down closer to the ground or press “G” to enter ground mode for a better view.



**Figure 7: A representative screenshot of GNNViz showing trees displayed on the terrain surface**

Once trees are generated, you can move around and view them as you wish, and use the buttons on the right hand-side of the program window to control what vegetation types are displayed and which decorator is drawn on the terrain surface.

Controlling the type of vegetation displayed can help you see some detail in denser areas. For example, in Figs. 8 and 9 the same area is shown first with all trees rendered (Fig. 8), and then with conifers hidden so only hardwoods and snags are displayed (Fig. 9). Note that with conifers hidden, it is easier to see the collection of snags in front of the current position, as well as the more distant groups of hardwoods.

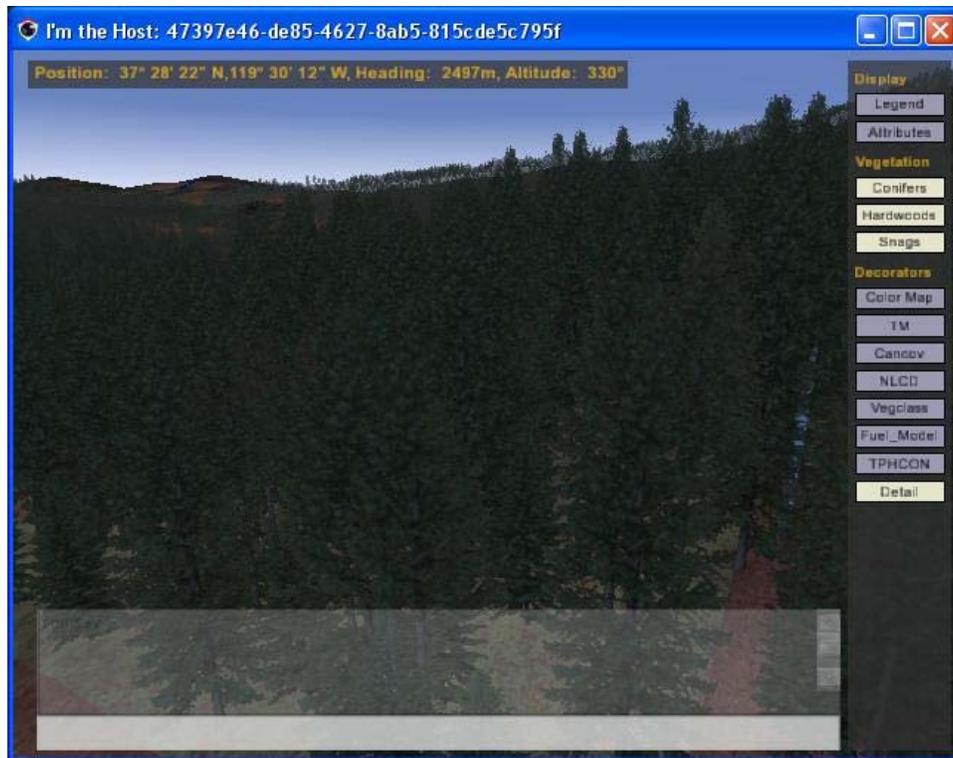


Figure 8: GNNViz display with all trees rendered

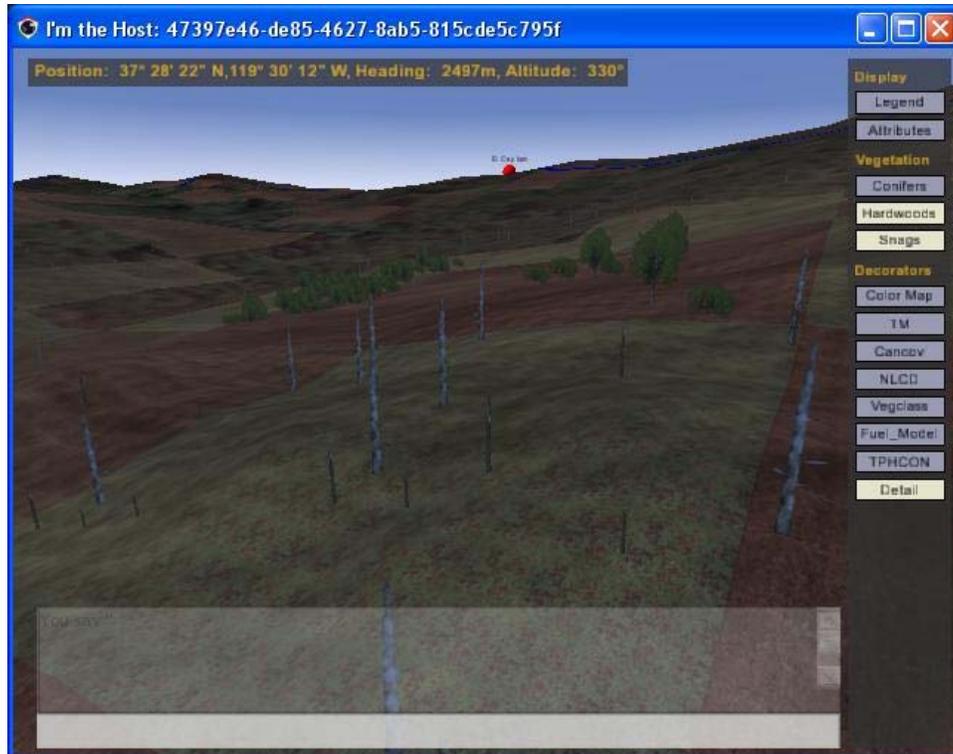


Figure 9: The same area but with conifers hidden

## Program Commands and Options

---

### Command-line Options

GNNViz supports several command-line options which allow you to set the program's behavior when started (Table 1). All are optional. However, it's necessary to specify a mission file with the `--mission` command-line option.

Command Option	Description
<code>--mission &lt;mission file&gt;</code>	Loads the given mission file and sets up GNNViz to use the configuration information therein to specify data values and options to be displayed.  Mission files are located in the <code>gnnviz\data</code> directory.
<code>--treeradius &lt;radius&gt;</code>	Instructs GNNViz to generate trees in an area that is 2x the given value across, with the dimensions being in degrees.  For example the command-line option, “ <code>--treeradius 0.1</code> ” would instruct GNNViz to generate trees in an area 0.2 degrees across.
<code>--host &lt;host address&gt;</code>	Instructs GNNViz to connect as a client to a GNNViz server at the given host address.

Table 1: GNNViz command-line options

### Keys

A number of basic commands and actions can be activated by simply pressing keys on the keyboard in GNNViz (Table 2). Most have been previously mentioned in describing movement and generating trees, but there are several in Table 2 that haven't yet been described.

Key	Description
<b>W</b>	Move forward relative to the current direction faced
<b>A</b>	Move left relative to the current direction faced
<b>S</b>	Move backwards relative to the current direction faced
<b>D</b>	Move right relative to the current direction faced
<b>F</b>	While held down, move 10x faster than normal
<b>G</b>	Toggle ground-lock model, which keeps the point of view pinned to a fixed height (4 m) above the ground
<b>B</b>	Generate tree billboards centered on the current user location.
<b>Enter</b>	Begin text entry mode for chat or commands (described below)

<b>Space</b>	Toggle display mode of GNNViz between normal and wireframe mode. This serves no real purpose other than letting the user see how the application has rendered the terrain as a series of polygons.
<b>Equal</b>	Toggles display of the 1-minute grid on the terrain surface. Note that you must change the display decorator to make the change take effect (this is a bug)

Table 2: GNNViz keys

## Console Commands

Besides the keys in Table 2, certain commands may be entered in the GNNViz console (Table 3). The console is at the bottom of the GNNViz window, and should not be confused with the command-line window that the program was initially started from.

To enter a console command, press the “Enter” key and begin typing. Press “Enter” again to submit the text. GNNViz interprets the text you enter in one of two ways: as a command to be interpreted or as a chat message for other users if in multi-player mode. Commands are identified by starting with a “/” character. For example typing, “/quit” in the console and pressing Enter will tell GNNViz to shut down and quit the application. Some commands have no arguments, but others take additional text following the command as arguments. Below (Table 3) is a list of the different commands and the actions they trigger.

<b>Command</b>	<b>Description</b>
<b>/jumpto &lt;latitude&gt; &lt;longitude&gt;</b>	Jumps the user’s point-of-view from the current location up into the air, and then back down to the new latitude and longitude coordinates given. Note that coordinates must be in decimal latitude and longitude, with negative values for South latitude and East longitude.
<b>/moveto &lt;latitude&gt; &lt;longitude&gt;</b>	Moves the user’s point-of-view from the current location in a direct line to the new coordinates.
<b>/goto &lt;latitude&gt; &lt;longitude&gt;</b>	Moves the user’s point-of-view directly to the new coordinates without any interim motion display.
<b>/speed &lt;speed&gt;</b>	Sets the movement speed of the viewer. Fast mode is 10x this speed, and when in ground mode, movement is 1/10 of this speed. The speed value is a floating point number, and represents the movement speed in meters per second.
<b>/name &lt;name&gt;</b>	Sets the user’s name that will be shown to others also connected in a multi-user environment. By

	default, the user's name is a unique string of letters and numbers.
<b>/who</b>	If the user is connected to a server or is hosting a multi-user server, this command lists the names of all other users who are connected to the server. The names displayed can be changed using the <code>/name</code> command described above.
<b>/quit</b>	Ends the GNNViz program.

Table 3: Console commands available in GNNViz

## Running GNNViz with Multiple Users

---

As mentioned previously, it is possible to run GNNViz as a multi-user application, where more than one person can move about and interact in the virtual environment. There is no centralized server that GNNViz users need to connect to, but rather any instance of GNNViz may be a server for others to connect to.

### Starting a GNNViz Server

To start a GNNViz server, simply start GNNViz as normal – no special command-line arguments or options are required. GNNViz is set to use port 4444 for all network communications between clients and a server, so be sure your PC firewall or network configuration isn't blocking access to this port by other machines.

In order for others to connect to your server, you will need to give them the IP address of your server so they can use it to connect in. If you don't know your server's IP address, open up a command-line window and run the “ipconfig” program. Fig. 10 shows an example of how ipconfig shows you your IP address.

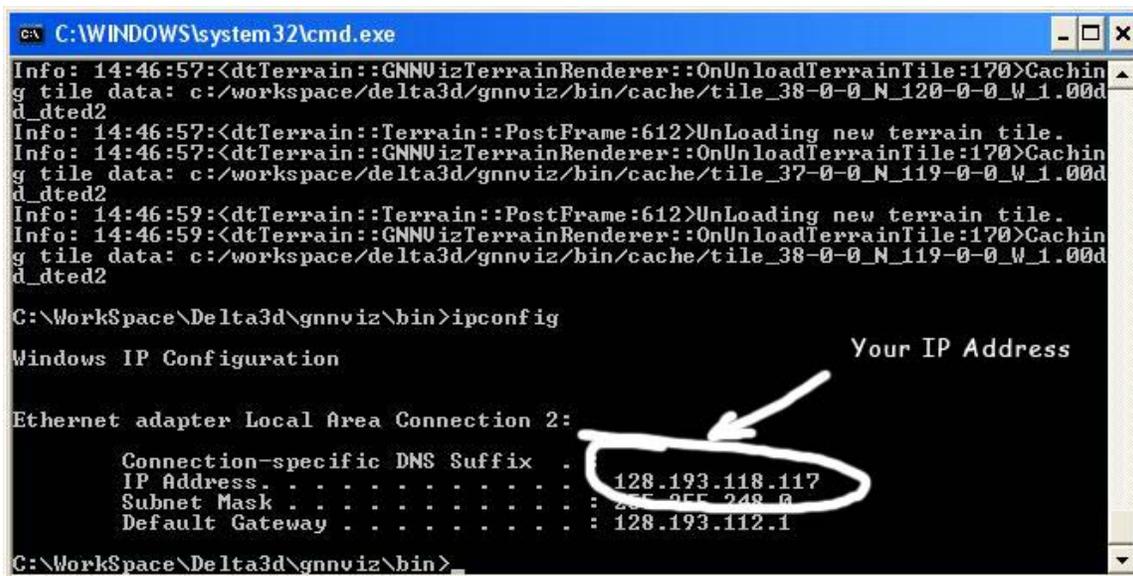


Figure 10: Determining your IP address

### Connecting to a Server

To connect to a GNNViz server, you will need to know its IP address. Start GNNViz as you normally would, specifying the mission file to use with the --mission command-line option. You will also need to use the --server command-line option to tell GNNViz the address of the server to connect to. Below (Fig. 11) is an example of starting up GNNViz and connecting in to the server at IP address 192.193.118.200.

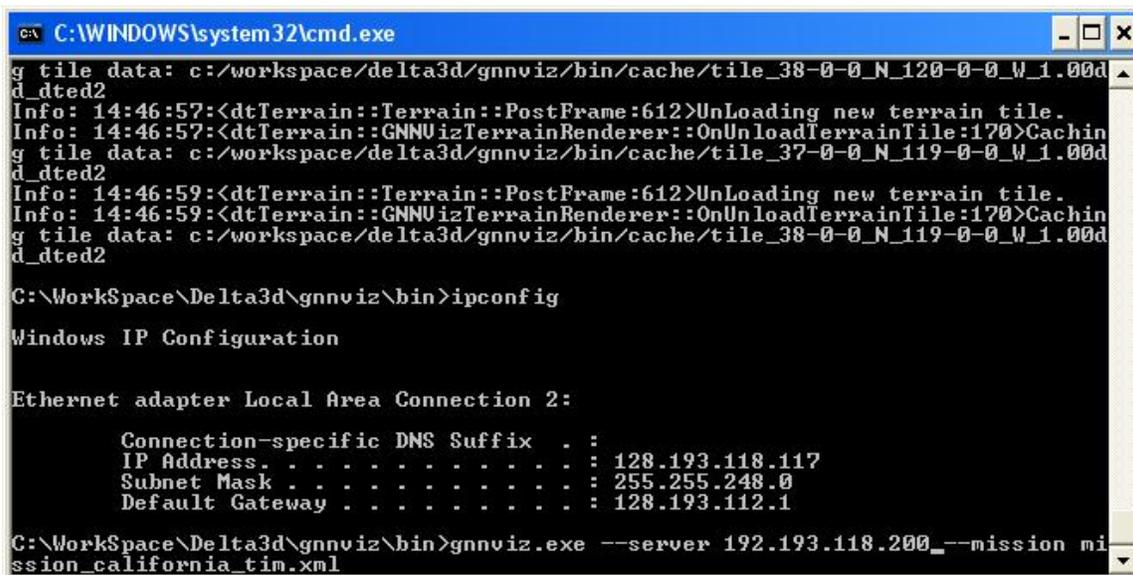


Figure 11: Specifying a server address for multi-player use

Once connected to the server, you can see and chat with other users on the server. Although you can't see yourself, you can see other players and they can see you. Each player is represented by a helicopter model, which will hover and move about based on where the controlling player is looking or moving (Fig. 12).



**Figure 12: Looking at another user in GNNViz**

Note that because the size of the landscape in GNNViz is immense, it can be rather difficult to find and see other players.

## Chatting with Other Users

GNNViz has a simple chat system built in that lets you send text messages to all users currently on the server. To send a chat message, press the “Enter” key to start text entry, type out your message, and hit “Enter” again to send it. As long as your text doesn’t start with a “/” character (reserved for commands), your text message will be repeated on the screens of all other connected users.

You probably should set your name with the “/name” command as described earlier under “Advanced Commands and Interactions.” The name you set will be used to label your text messages so others will know who has sent the message.

## **Building the GNNViz Cache**

---

To make management and access of data faster and more streamlined, GNNViz uses a cache system where data and images are written to disk as separate tiles of data covering 1-by-1-degree squares. As you move around in the GNNViz virtual landscape, these tiles will be loaded into memory as they become visible. Tiles that are no longer visible will be removed from the program's memory.

Although GNNViz can generate and build the cache files, it is a slow and memory-intensive process that can greatly impact the performance of GNNViz. To simplify generating the cache files, GNNViz comes with a companion program named "buildcache.exe". It is located in the same directory as the gnnviz.exe program, and uses the same --mission command-line option to specify the mission file to be read for processing. Generating the cache should only take a minute or so and will greatly speed up use of GNNViz.

## **Modifying and Building the GNNViz Executable**

---

GNNViz was developed using the Microsoft Visual C++ compiler and is based on Delta3D version 1.4. Although GNNViz is delivered as a compiled executable program, it is possible to build it from the source code provided on the DVD or downloaded from our website. Along with the GNNViz source code you will find a Microsoft Visual C++ solution file that can be used to build the main gnnviz.exe and buildcache.exe programs.

Note that during development of GNNViz, version 1.5 of Delta3D was released. Rather than move to this version, the developers chose to stick with version 1.4, as 1.5 did not seem to offer any particular advantage or improvements applicable to GNNViz.

To download the GNNViz source code, visit the project website <http://www.fsl.orst.edu/lemma/gnnviz>. For information on building and using Delta3D, see the Delta3D website at <http://www.delta3d.org>.

If you do make any modifications, bug fixes or additions to GNNViz, you are encouraged to send them to the developers for possible inclusion. Contact either [janet.ohmann@oregonstate.edu](mailto:janet.ohmann@oregonstate.edu) or [matt.gregory@oregonstate.edu](mailto:matt.gregory@oregonstate.edu).

## Frequently Asked Questions

---

Below is a collection of FAQs that may help you with using GNNViz.

### **Why are trees only drawn over a small area on the terrain surface?**

Basically, to keep the programming running fairly smoothly. If too many objects are on the screen at one time, the program's performance degrades so that movement and any interaction becomes very jerky and difficult. Realize that for a typical forested area, there may be many tens of thousands of individual trees on display.

### **Why doesn't GNNViz draw the actual number of trees present according to the GNN prediction model?**

Like the question above, the answer lies in keeping the number of objects the game engine has to draw down to a reasonable number so that the program is not overloaded.

### **Why are the trees drawn as flat images rather than as 3D models?**

Two reasons. First, it is far easier to create 2D images of trees with Photoshop or other image programs than to create full 3D models of trees. Second, drawing of billboards (as the flat images are called) is often more efficient than rendering 3D models. It's actually a very visually effective way to render trees, and one can't exactly tell they're just flat billboards except when viewed up close.

### **Can I create my own billboard images for use with GNNViz?**

Yes! The images are simply PNG files with a transparent background. All billboards are rendered to a default height of 10 meters multiplied by the scaling factor specified for the billboard in the mission XML file. A scale of 2.5, for example, translates into the given image being displayed as a 25 meter tall object.

A good commercial source of vegetation billboard images is XFrog. See <http://www.xfrogdownloads.com/greenwebNew/products/Version2/press.htm>

### **Why are there long delays at times when I change to a new decorator layer I haven't previously looked at?**

When a decorator is displayed for the first time, GNNViz has to first tile the data – breaking it up into a series of fixed 1-by-1-degree images and saving them to a cache directory for use in displaying the data. Depending on the size of the initial source image and its original projection, this conversion may take up to several minutes. It's actually advisable to run the buildcache.exe program before running GNNViz to pre-build the entire cache file before running GNNViz.

### **Why does GNNViz pause at times as I'm flying across the landscape?**

GNNViz only keeps the currently visible terrain in memory to reduce the amount of work required to render the terrain. The data is divided up into a series of 1-by-1-degree tiles, and GNNViz loads and unloads these tiles as needed. When you fly to a point where a new tile becomes visible, GNNViz has to unload tiles that have gone out of range and then load up tiles that have become visible.

### **Help! All I see on my screen is blue sky!**

It's fairly easy to change your view such that you are looking up at the sky, which can be a bit confusing as you have no points of reference. To re-orient yourself, click on the middle of the GNNViz window and move the cursor downwards.

### **Why are the decorator images drawn all "pixelly" on the terrain surface?**

The pixelation of the rendered decorators reflects the underlying, 30-meter resolution of the original data from the GNN model. A smoothed and blended image might perhaps look better, but would not be an accurate reflection of the actual underlying data.

### **Why don't I see any trees on the terrain surface?**

First, you need to press the "B" key to command GNNViz to generate trees centered on your current location. Next, make sure that the individual vegetation groups (conifer, hardwood, and snag) are turned on in the right-hand set of display control buttons.

### **I started a server and nobody seems to be able to connect to it.**

First, check that you have given the other users the right IP address and that they're actually connecting to your server's machine. Next, make sure that neither your machine, the network, nor the other users' machines are blocking port 4444. Firewalls and network configurations and policies may block access to this non-standard port number. If you aren't sure if it's being blocked, contact your system administrator staff for assistance.

### **I have joined a server and I can't see anyone else!**

Don't forget that the size of the players (represented by helicopters) is very small as compared to the size of the terrain being drawn. You may find you're within several miles of another player, but are unable to see them. You might try getting the other player's coordinates and use the /goto command to move to their location, or agree to both /goto a specific coordinate.

### **Is there a way to send a message to just a single person in a server?**

No, your chat messages can only be broadcast to all users on the server.

### **Can I pre-build my cache files to save time when running GNNViz?**

Yes. See “Building the GNNViz Cache” in this users guide.

### **Is it possible to change the keys used for movement and other commands?**

The key bindings are hardwired into the GNNViz application and can't be changed without modifying the source code.

### **I have modified my mission file, and now GNNViz won't run!**

The format of the mission XML file is very specific and must match a certain set format. If you've introduced an error in your XML, you'll find that the program won't run and will probably quit with an error message. As such it is always a good idea to save a copy of your mission files before editing them.

To tell if there is a problem with your XML format, look at the text output from the command-line window where you started GNNViz and see if there are any error messages referring to a “Xerces error”. Xerces is the name of the XML parsing library used by GNNViz.

For example, trying to load a mission file with an error in the second line displays the error message...

```
Error: 02:20:41:<dtUtil::XercesErrorHandler::error:32>Xerces
error occurred while parsing at line, 2
```

### **Is it possible to change the size of the area where vegetation is drawn and generated?**

Yes. There's a command-line option for GNNViz that can be used to override the size of the default area in which vegetation models are generated on pressing the “B” key. The command-line option is `--treeradius` followed by the size of the half dimension of the bounds in degrees. For example using the option `--treeradius 0.1` will generate vegetation in an area that is 0.2 degrees across.

The default value is one minute, or 0.01666. Note that increasing this value to large sizes may greatly decrease the performance of the GNNViz program and cause excessive CPU usage. Conversely, decreasing the value will increase the program's performance, which may be useful for those on slower or older systems not capable of running GNNViz effectively.

### **How do I take screenshots or capture movies of GNNViz?**

Try the free program called MWSnap, available at <http://www.mirekw.com/winfreeware/mwsnap.html>.

**Why are some areas of the terrain decorator white and showing no data?**

These are areas outside of the GNN project area and no data is available for these areas.

**I have flown to the edge of the project area and can't seem to move any further – it's like there is an invisible wall in front of me.**

GNNViz does not allow you to fly beyond the project boundaries specified in the mission XML file.