MÄK Stealth User's Guide Addendum A

This is the first addendum to the *MÄK Stealth User's Guide*, STL-3.3-1-981008. This addendum supports MÄK Stealth Release 3.4. This addendum covers the following subjects:

Document ID: STL-3.4-1A-990512

System Requirements

This section describes platform support and system requirements for the Stealth. For the most up-to-date information about system requirements, see the Product Versions page on the MÄK web site at: http://www.mak.com/military/compatibility.html

Stealth for SGIs

The Stealth is available for systems running Irix6.2 or later, with OpenGL and Performer 2.2.

The Stealth for SGIs 3.4 was built under IRIX6.2 against IRIS Performer 2.2 using the N32 ABI. The Stealth executable is statically linked with Performer, so you don't need to have Performer installed on your system in order to run. The smaller *Stealth.dynamic* executable is dynamically linked with Performer, and with the OpenFlight Performer loader.

If you are using *Stealth.dynamic*, you need performer_eoe version 2.2 or a compatible version installed. You must also have a version of the MultiGen loader DSO (*libpfflt_ogl.so*) that is compatible with the *pfflt.h* header file that comes with Performer 2.2.

The loader DSO that comes with Performer 2.2 is a version 14.2 loader. In order to load files of more recent OpenFlight versions using the dynamically linked Stealth, you need to obtain a later version of the loader DSO (*libpfflt_ogl.so*) from MultiGen's FTP site (ftp.multigen.com/pub), and install it in */usr/lib32/libpfdb*, or some other location contained in your LD_LIBRARYN32_PATH environment variable.

Stealth for PCs

The Stealth for PCs requires the following:

- A Pentium-class PC (or higher) with a standard VGA or SVGA video board and CD-ROM drive.
- Windows NT 4.0 or Windows 95/98.
- 14 Mb of disk space for program files, and 65 Mb for optional data files.
- A 3D video board based on the Voodoo (3Dfx) chipset, Direct3D, or a video board with OpenGL drivers. The 3D board may have been included as part of your Stealth for PCs package.
- A second monitor enhances ease-of-use with boards based on the Voodoo chipset and is highly recommended, but is not required.

Backward Compatibility

Stealth 3.4 changes the configuration parameters for electromagnetic emissions (sensor volumes). If you have changed the default configuration for sensor volumes (SGI only), you must update your configuration file with the new parameters to continue using this feature. For complete details, see "Configuring Electromagnetic Emissions (SGI only)" on page 19.

Network Compatibility

HLA only

The Stealth 3.4 is built against VR-Link 3.4. This change makes the Stealth compliant with:

- RPR-FOM 0.5, 0.7, and 0.8
- MAK RTI 1.3.1
- DMSO RTI 1.3v4, 1.3v5, 1.3v6.

If you run the HLA Stealth 3.4 with version 0.5 of the RPR FOM, it is network compatible with applications that are using VR-Link 3.3. It is not network compatible with applications that use VR-Link 3.2 or earlier and it is not compatible with applications using VR-Link 3.3 if the Stealth is using RPR FOM 0.7 or 0.8.

DIS only

The DIS Stealth 3.4 is network compatible with previous releases. The DIS protocol has not changed.

Changes to FLEXlm Setup and Configuration

With this release of the MÄK Stealth, we have simplified the process for configuring the FLEXlm license management. It is no longer necessary to put a license file on each client machine, and you no longer have to merge license files when you purchase new products or additional licenses.

If you do not have a prior installation of MÄK Technologies products, follow the instructions in "Setting Up and Using the FLEXlm License Manager" on page 6. If you already have MÄK products installed, your upgrade procedure depends on the version of FLEXlm that you are running. Table 1 lists MÄK products and how to proceed.

Table 1: Product versions and license management

Product	FLEXlm Version	Procedure to Follow
VR-Link 3.0, 3.1 Stealth for SGI 3.1, 3.1.1 Stealth for PCs 3.2.1 Logger 3.1	5.x	Follow installation procedure in original product documentation. Merge license files as described in "Merging New License Files for Products Using FLEXlm 5.x" on page 5.
VR-Link 3.2 or later Stealth for SGI 3.2 or later Stealth for PCs 3.3 or later Logger 3.2 or later Plan View Display 1.x	6.0	Follow the instructions in "Updating Your Existing License Management Configuration" on page 4.

Updating Your Existing License Management Configuration

If you have FLEXlm license management configured for products you purchased prior to the current release, you must do the following to update your license management configuration:

1. On the license server machine, rename the *license.dat* file to *mak.lic* (or any name ending in *.lic*). (If you are unsure about the distinction between the license server machine and client machines, see "The FLEXlm Client-Server Architecture" on page 6.)
2. Open the license file in a text editor.
3. Remove the port number from the SERVER line. For example, if the first line of the license file is:

   ```
   SERVER oak 0800359de785 4567
   ```

   the port number is 4567. Delete it from the file.
4. Remove the LM_LICENSE_FILE environment variable from the server machine and all client machines unless another vendor requires it.

5. Delete the *license.dat* file from all client machines.
6. Set the MAKLMGRD_LICENSE_FILE environment variable on all client machines (including the license server machine if you run MÄK Technologies products on it).

   The syntax for the environment variable is: *@Server_name*. For example, if the server machine is oak, set the environment variable to `@oak`.

Merging New License Files for Products Using FLEXlm 5.x

If you are using FLEXlm 5.x, you cannot use the new method of license management. If you receive a new license file from MÄK, you need to merge it with the old file.

1. Your new license file will look something like the following:

```
SERVER mollusc 006097752f93
DAEMON maklmgrd .
PACKAGE dir maklmgrd 1999.177 60D00041FA200D8A795D \
      COMPONENTS=dir1
INCREMENT dir maklmgrd 1999.177 1-jan-0 1 0BB480511615C175B8CF \
      PLATFORMS=i86_n
```

   Your existing license.dat file is similar to the following:

```
SERVER mollusc 006097752f93 4567
DAEMON maklmgrd "c:\Program Files\MÄK Technologies\stealth\flexlm\
      maklmgrd"
PACKAGE makproduct maklmgrd 1999.177 40C020D12D609AB04D55 \
      COMPONENTS="hprod1 dprod1 prod1"
INCREMENT makproduct maklmgrd 1999.177 1-jan-0 1 AB4450916DD1944435D2 \
      PLATFORMS=i86_n
```

Notes

- The server name and host id in the two license files should be identical. The new file will not have a port number.
- The line continuation marks (\) in the examples are for clarity of presentation. They may not be present in the files you receive.

2. Open up the two files in a text editor. Append everything except the Server and Daemon lines (the Server lines will be identical except for the port) from the new *license.dat* file to the end of the old *license.dat* file. For example:

```
SERVER mollusc 006097752f93 4567
DAEMON maklmgrd "c:\Program Files\MÄK Technologies\stealth\flexlm\
      maklmgrd"
PACKAGE makproduct maklmgrd 1999.177 40C020D12D609AB04D55 \
      COMPONENTS="hprod1 dprod1 prod1"
INCREMENT makproduct maklmgrd 1999.177 1-jan-0 1 AB4450916DD1944435D2 \
      PLATFORMS=i86_n
PACKAGE dir maklmgrd 1999.177 60D00041FA200D8A795D \
      COMPONENTS=dir1
INCREMENT dir maklmgrd 1999.177 1-jan-0 1 0BB480511615C175B8CF \
      PLATFORMS=i86_n
```

3. Put the updated *license.dat* file in the *flexlm* directory on the license server and on all machines on which you have the Stealth installed.

Setting Up and Using the FLEXlm License Manager

The Stealth, like other MÄK Technologies products, uses the FLEXlm license manager. Before you can use the Stealth, you must obtain a valid license file and configure the license server and client machines. (For information about FLEXlm, view the FAQ page at www.globetrotter.com/flmfaq.htm.) Contact the MÄK Technologies sales department for information about special licensing agreements.

The FLEXlm Client-Server Architecture

This section explains the FLEXlm architecture to provide a context for the setup instructions that follow. FLEXlm uses a client-server architecture. The FLEXlm executable, *maklmgrd* runs on one computer, known as the license server. This machine services every machine (including possibly itself) on which you want to run a MÄK Technologies product. The machines on which the MÄK Technologies products are running are called clients.

The license server has a license file (supplied by MÄK Technologies). This file identifies the MÄK Technologies products for which you have licenses. On each client machine, the MAKLMGRD_LICENSE_FILE environment variable (which you must set) points to the server.

Licenses "float", so you can install product software on more machines than you have licenses for, but at any given time, you can run only as many instances of a product as you have licensed.

! The license key in the license file is tied to the host id of the server machine. Therefore, if you decide to change the server, you need to get a new license file. Then, you must update the MAKLMGRD_LICENSE_FILE environment variable on every client. This is not a trivial process, and is not recommended.

Figure 1 illustrates the client-server architecture for FLEXlm and MÄK products.

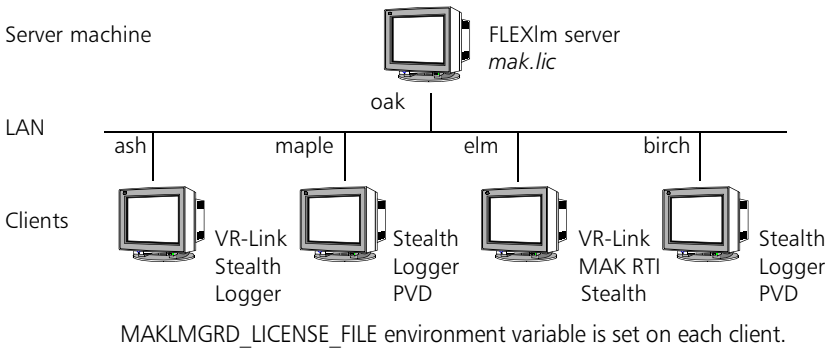

Figure 1. FLEXlm example architecture

Installing and Configuring the License Manager

The general procedure for installing and configuring license management is as follows (each step is explained in the rest of this section):

1. Install license manager software.
2. Request a license file.
3. Put the license file in the *flexlm* directory on the license server.
4. Set the MAKLMGRD_LICENSE_FILE environment variable on all client machines.

If you have MÄK Technologies products installed and you purchase additional licenses or products, you must get a license file for the new products or licenses. For this procedure, see "Adding Additional License Files" on page 10.

Installing the License Manager Software

As part of the installation of all MÄK products, the FLEXlm software is installed in the *flexlm* directory under the root product directory.

1. Install your MÄK products on the machine(s) on which you want to run them.
2. If this is a networked system, and you did not install MÄK software on the license server, copy the *flexlm* directory from a client machine to the license server machine.

Requesting A License File

License files are keyed to the host id of the license server. You need to supply MÄK with some information so that we can generate a license file for you.

To get a license file:

1. On the license server, in a command window, change to the *flexlm* directory.
2. Execute *lmhostid.* This program generates a unique number that MÄK uses to generate your license activation codes. Write down the number.
3. Go to http://www.mak.com/support/keyintro.htm and complete the license key request form. MÄK will e-mail you a file named *mak.lic*.

To get a license file, you need to know the name of the license server machine. To get the name of your machine follow the procedure for your operating system:

UNIX

At a command prompt, type `hostname`.

Windows 95/98/NT

1. On the Start menu, choose Settings → Control Panel.
2. Open the Network applet.
3. Select the Identification tab. The machine name is listed in the Computer Name field.

Putting the License File in the *flexlm* Directory

After you receive the license file, put it in the *flexlm* directory on the license server.

! If you already have a *mak.lic* file in the *flexlm* directory, do not overwrite the file. See the instructions in "Adding Additional License Files" on page 10.

Setting the MAKLMGRD_LICENSE_FILE Environment Variable

The MAKLMGRD_LICENSE_FILE environment variable identifies the server machine so that the FLEXlm software on a client can check out a license when you run a MÄK Technologies product. You must set MAKLMGRD_LICENSE_FILE on every client machine.

---

i If you are running MÄK Technologies products on the license server machine, it is also a client, so you must set MAKLMGRD_LICENSE_FILE.

The syntax for the environment variable is: *@Server_name*. For example, if the server machine is oak, set the environment variable to `@oak`.

The following sections explain how to set environment variables on the different platforms that MÄK Technologies products run on.

Windows 95/98

On Windows 95/98, you set environment variables by adding or editing lines in the *Autoexec.bat* file. To set the MAKLMGRD_LICENSE_FILE variable, add a line similar to the following:

```
set MAKLMGRD_LICENSE_FILE=@oak
```

! Do not put spaces around the equal (=) sign.

Windows NT

On Windows NT, you set environment variables in the Environment tab of the Properties dialog box for the machine:

1. Right-click the My Computer icon on the Desktop.
2. On the pop-up menu, select Properties. The System Properties dialog box opens (Figure 2).
3. Select the Environment tab.
4. In the Variable text field, type MAKLMGRD_LICENSE_FILE.
5. In the Value field, type the value, for example:

```
@oak
```

Figure 2. Setting an environment variable in Windows NT

UNIX

On UNIX, you set environment variables in your *.cshrc* (or equivalent startup file). Set the variable similarly to the following example:

```
setenv MAKLMGRD_LICENSE_FILE @oak
```

Adding Additional License Files

If you buy additional MÄK products, or additional licenses for products you already have, you must get an additional license file.

> To request an additional license file, follow the same procedure as for your first license file. See "Requesting A License File" on page 8.

To add a new license file to your license server:

1. When you get the new license file, rename it from *mak.lic* to something like *mak2.lic*, or some other name ending in *.lic*, so that it has a different name from the license file(s) currently in the *flexlm* directory on the license server.
2. Put the renamed file in the *flexlm* directory on the license server, with the other license files.

New Features and Product Changes

This release of the MÄK Stealth has the following new features:

- Support for the VR-Link FOM Mapper.
- Multichannel support for PCs.
- Sky color on PCs is configurable through MTL variables.
- The format for latitude, longitude, and altitude units in the 2D overlay is determined by the settings chosen in the GUI Options dialog box.
- Support for electromagnetic emissions in HLA and new configuration parameters.
- Support for clip textures (SGI only).
- Statistics option for the 2D overlay (PC only).
- Support for Direct 3D graphics cards (PC only).
- New default installation directory structure.

Selecting A FOM Mapper

Stealth 3.4 has built-in support for versions 0.5, 0.7, and 0.8 of the RPR FOM. It also supports VR-Link's ability to support alternative FOMs through the FOM Mapper.

You can specify which FOM Mapping code to use with the `-f` startup option, in one of two ways:

```
stealth -f { vlRPR05 | vlRPR07 | vlRPR08 }
stealth -f library:init_data
```

If the argument to the `-f` option is one of the vlRPR0x keywords, the Stealth uses VR-Link's built-in RPR FOM mapper, configured for RPR FOM version 0.5, 0.7 or 0.8 as specified by the keyword.

If the argument to the `-f` option is not one of the three keywords, it is interpreted as the name of a shared library (DSO or DLL) containing code for an alternate FOM Mapper that you would like the Stealth to use, optionally followed by any initialization data the library requires. The library name and initialization data must be separated by a colon.

Examples

```
stealth -f vrlRPR05
```

This command instructs the Stealth to use the built-in RPR FOM Mapper, configured for RPR FOM version 0.5.

```
stealth -f myFomMapLib:ABC
```

This command instructs the stealth to use the FOM Mapper contained in a shared library called *myFomMapLib.so* (or *myFomMap.dll*). The string "ABC" is passed to the FOM Mapper creation function in the shared library as an initialization parameter.

Specifying A FED File that is Appropriate for the FOM Mapper

The `-f` option only tells the application which FOM Mapping code to use. It does not ensure that you are using the right FED file for the FOM Mapper. Make sure that you are using a FED file that is consistent with the FOM Mapper you have chosen.

By default the Stealth uses a FED file called *VR-Link.fed*, but you can specify an alternate FED file using the `-x` option. The `-x` option actually indicates the name of the federation execution name, but we append *.fed*, and use that as the FED file name as well.

The Stealth comes with three FED files:

- *VR-Link.fed*, based on RPR FOM 0.8
- *VR-Link07.fed*, based on RPR FOM 0.7
- *VR-Link05.fed*, based on RPR FOM 0.5

The following example instructs the application to use the RPR FOM 0.5 FOM Mapping code (`-f vlRPR05`), and also instructs it to use the *VR-Link05.fed* file (`-x VR-Link05`).

```
stealth -f vrlRPR05 -x VR-Link05
```

As an alternative to using the `-x` option, you could rename the *VR-Link05.fed* file to *VR-Link.fed*.

Specifying the FOM Version in A Configuration File

You can use the following configuration parameters instead of the `-f` option to specify which FOM Mapper you would like to use:

```
(setq rprFomVersion version_number)
(setq fomMapperInitData "data")
(setq fomMapperLib "libName")
```

Set the *version_number* for **rprFomVersion** to 0.8, 0.7 or 0.5.

fomMapperLib specifies the name of a FOM Mapper shared library to use. **fomMapperInitData** specifies a text string that you would like to pass as initialization data to the FOM mapper creation function contained in the library.

Just as with the `-f` option, you specify the appropriate FED file for the FOM version you are using. Specify a FED file with the `-x` command line option or the **execName** configuration parameter.

Displaying Multiple Views on PCs

Previous releases of the Stealth supported a configuration in which the Stealth controls are displayed on one monitor and the graphics view on a second. The Stealth for PCs now supports display of graphics on multiple monitors using the following possible configurations:

- One PC, running one Stealth, displaying graphics views on multiple monitors
- Multiple PCs, each running a Stealth, displaying different views that are controlled from one master PC
- Any combination of the above options.

Although you can mix and match the two configuration options, we will discuss them separately for the sake of convenience.

Notes

- The process for using multiple monitors with the Stealth for PCs is completely different from that for Stealth for SGI.
- Configuring multi-channel support in PCs requires the use of several new MTL parameters and a new environment variable. The new variable and parameters are introduced in the context of their use.

Figure 3 illustrates how three monitors could provide three side-by-side views into a Stealth. These views could be provided by either of the configuration options discussed above. Figure 4 illustrates how monitors could be arranged so that some of the views are rotated around the z axis of the Stealth's eyepoint.

Setting the FX_GLIDE_NO_SPLASH Environment Variable

Regardless of how you configure PCs and monitors, you must set the FX_GLIDE_NO_SPLASH environment variable to 1. If you are not sure how to set environment variables on your PC, see "Setting the MAKLMGRD_LICENSE_FILE Environment Variable" on page 9.

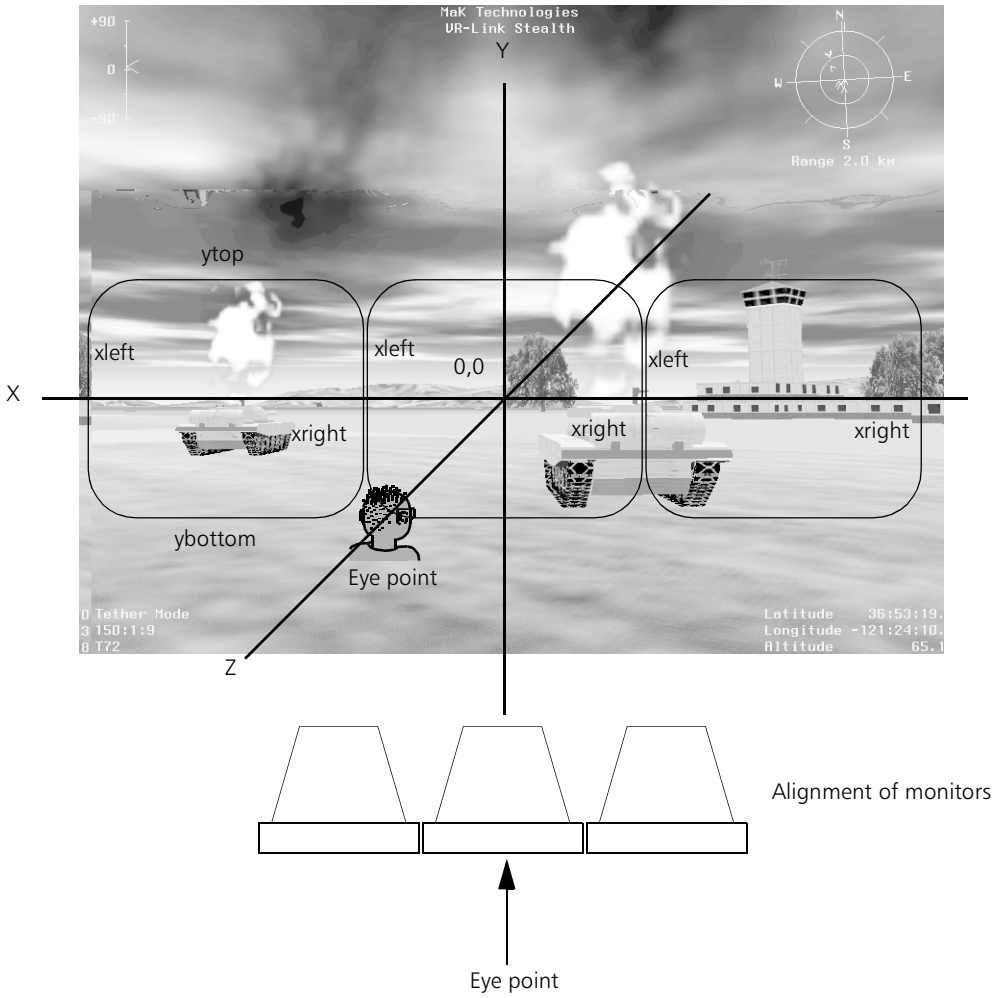

Figure 3. Three monitors, same orientation

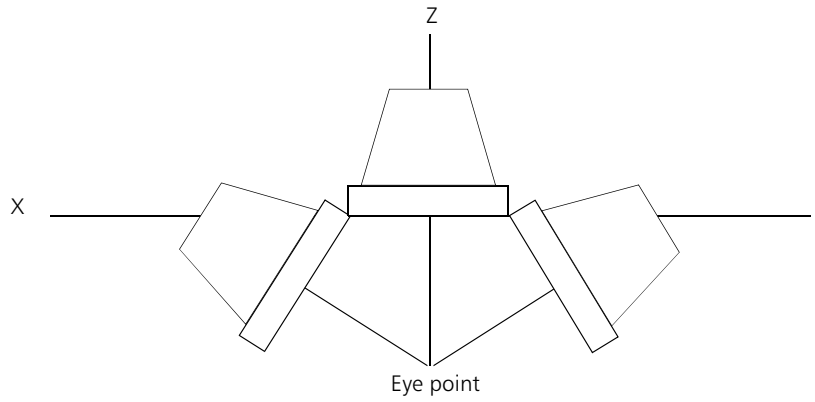

Figure 4. Three monitors, different orientations

Viewing the Stealth on One PC with Multiple Monitors

Running multiple monitors from one PC is essentially an extension of the dual-monitor mode. You can run a monitor for each graphics board you install in the PC and can have as many graphics boards as you have available slots for installation. The boards must use the 3Dfx chip set and must be the same manufacturer and model.

The advantage of this configuration (versus multiple PCs) is relatively low cost. The primary disadvantage is that as you add graphics boards, performance can be affected.

Configuring Multiple Monitors Attached to One PC

Configuration information for multiple monitors is contained in *channelConfig.mtl*. (This is the default configuration file. You can name the file anything you want.) You identify the configuration file with the following line in the *params.mtl* file:

```
(setq channelConfigFilename "channelConfig.mtl")
```

In the *channelConfig.mtl* file, you specify the views for each monitor using the create channel command, in the following format:

```
(create channel
    xres    yres
    xleft ybottom ytop xright
    heading pitch roll
)
```

where:

- xres is the horizontal resolution of the channel, in pixels.
- yres is the vertical resolution of the channel, in pixels.
- xleft is the position on the x-axis of the left edge of the view.
- ybottom is the bottom edge of the view along the y-axis.
- ytop is the top edge of the view along the y-axis.
- xright is the position on the x-axis of the right edge of the view.
- heading, pitch, and roll is the orientation of the view relative to the eye point, in degrees.

Put one create channel command in the configuration file for each monitor you want to configure.

Sample Configuration File for Identically Oriented Monitors

Figure 3 shows three monitors, side-by-side, all with the same orientation. The configuration file for these monitors would look like the following:

```
;; Channel configuration MTL file
;; Arguments: xres yres left bottom right top heading pitch roll

(create-channel
   1024  768
  -1.4029 -0.3249 -0.5907 0.3249
   0.0   0.0   0.0
)

(create-channel
   1024  768
  -0.4061 -0.3249 0.4061 0.3249
   0.0   0.0   0.0
)

(create-channel
   1024  768
   0.5907 -0.3249 1.4029 0.3249
   0.0   0.0   0.0
)
```

Although the monitors in Figure 3 are organized horizontally, they could just as easily be vertical or diagonal or some other relationship. Figure 4 shows three monitors, with the two side monitors set off at an angle. Again, there could be six, eight, or more monitors arranged in a circle or some other format, if your hardware supports such a configuration.

Sample Configuration for Angled Monitors

The following code shows how to configure monitors similarly to those in Figure 4:

```
;; Channel configuration MTL file
;; Arguments: xres yres left bottom right top heading pitch roll

(create-channel
   1024  768
  -0.4061 -0.3249 0.4061 0.3249
   -44.2   0.0   0.0
)

(create-channel
   1024  768
  -0.4061 -0.3249 0.4061 0.3249
   0.0   0.0   0.0
)

(create-channel
   1024  768
  -0.4061 -0.3249 0.4061 0.3249
    44.2   0.0   0.0
)
```

Calculating the X and Y Values

Calculate the coordinates for the view edges (xleft, xright, ytop, ybottom) with the eyepoint positioned one (1) unit from the screen. The coordinate value is the length along the x or y axis from 0 to the edge. You can calculate this value by finding the tangent of the angle formed by 0, the eyepoint, and the view edge.

For example, in Figure 5, with the eyepoint 1 unit from 0 on the x axis, the angle for the xleft coordinate on monitor 1 is 60^{o}. The xleft coordinate is -tan(60).

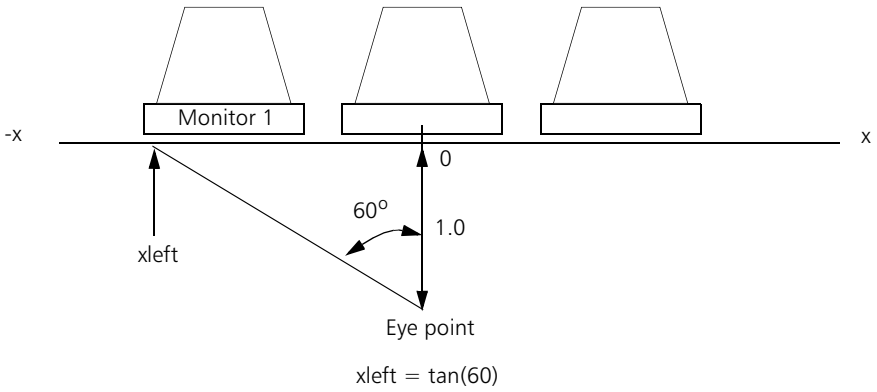

Figure 5. Calculating a view edge coordinate

Configuring the Heading, Pitch, and Roll Values

If your monitors are arranged perpendicularly to the x and y axis of the eye point, your heading, pitch, and roll values are all 0. If you rotate the monitors around the z axis, you need to set the heading, pitch, and roll accordingly. See "Sample Configuration for Angled Monitors" on page 16 for an example of setting heading, pitch, and roll.

Displaying Multiple Views on Multiple PCs

To run multiple views on separate PCs, you need the following hardware configuration:

- One PC from which to run the Stealth controls
- As many PCs as desired for Stealth views, each with one or more 3D graphics boards (they do not have to be the same in each PC)
- A Stealth license for each PC in the configuration

In this configuration, there is one "master" PC and multiple "slaves". The master PC controls the slaves by sending View Control messages to a unique ID. You need to configure the ID that the master uses to send the messages to the slaves, and configure the slaves to listen for messages using that ID. You also configure the views on the slaves.

Configuring the Master PC

To configure the master PC, set the **broadcastVcInteractions** and **broadcastVcViewerId** parameters in *params.mtl*. **broadcastVcViewerId** takes different arguments for DIS and HLA.

DIS only

```
(setq broadcastVcInteractions 1)
(setq broadcastVcViewerId "site:host:entity")
```

For DIS **broadcastVcViewerId** takes the site:host:entity triplet, for example:

```
(setq broadcastVcViewerId "5:6:7")
```

HLA only

```
(setq broadcastVcInteractions 1)
(setq broadcastVcViewerId "id_name")
```

For HLA, **broadcastVcViewerId** takes a name of your choice to identify the ID to which the master PC is broadcasting, for example:

```
(setq broadcastVcViewerId "hla_stealth")
```

The default value for **broadcastVcInteractions** is zero (0), which means that it doesn't send View Control messages. The default for **broadcastVcViewerId** is "", that is, no unique ID.

You can use any ID that is appropriate for your protocol (DIS or HLA).

In the Stealth Options dialog box for the master PC, set the Stealth View Control option to Ignore (you don't want it listening to itself), or set the MTL parameter **processViewControl** to 0 (zero).

Configuring the Slave PCs

1. In the *params.mtl* file for each slave PC, set the value to which the master PC will broadcast.

   For HLA, use the parameter **stealthHlaName**, for example:

   ```
   (setq stealthHlaName "hla_stealth")
   ```

   For DIS, specify the site:host:entity values, for example:

   ```
   (setqb stealthSite 5)
   (setqb stealthHost 6)
   (setqb stealthEntity 7)
   ```

2. In the Stealth Options dialog box on each slave PC, set the View Control option to Process Mine, or set the MTL parameter **processViewControl** to 1 (one).

3. On each slave PC, add a create channel command for each graphics board to the *channelConfig.mtl* file.

Running Multiple Views on Multiple PCs

To run multiple views on multiple PCs:

1. Configure the PCs as described in the previous sections.
2. Start the Stealth on each PC.
3. Use the Stealth controls on the master PC. The views on the slave PCs will respond as if they were one integrated viewer.

Configuring Sky Color on PCs

To configure sky color on the Stealth for PCs, use the same parameters as those documented for Stealth for SGIs. The parameters are skyColor[0], skyColor[1], and skyColor[2].

Displaying Latitude, Longitude, and Altitude on the 2D Overlay

The format for latitude, longitude, and altitude units in the 2D overlay is determined by the settings chosen in the GUI Options dialog box. In previous versions of the Stealth you could not change the units displayed in the 2D overlay.

Configuring Electromagnetic Emissions (SGI only)

The Stealth now supports electromagnetic emissions in HLA. The parameters in the params.mtl file have changed. If you have been using electromagnetic emissions for DIS exercises, and you changed any of the default settings, you will need to reconfigure them with the new parameters.

The electromagnetic emissions section of the params.mtl configuration file is as follows:

```
;; Electromagnetic emissions
      (cond
         ((equal hla 1) (setq emEmissionsTimeoutInterval -1.0))
         (t (setq emEmissionsTimeoutInterval  100.0))
      )

      (setq emEmissionsDisplayMode       1)
      (setq emEmissionsDefaultFarColor   0x7fff00ff)
      (setq emEmissionsDefaultNearColor  0x7fffffff)
      (setq emEmissionsHighFrequency     1000000000.0)
      (setq emEmissionsLowFrequency      800000000.0)
      (setq emEmissionsHighPrFrequency   3000.0)
      (setq emEmissionsLowPrFrequency    100.0)
      (setq emEmissionsRadiusScaleFactor 100.0)
      (setq emEmissionsRadiusExponent    0.100000)
      (setq emEmissionsShowLines         1)
      (setq emEmissionsColorIndicates    2)
      (setq emEmissionsSegmentAngle      10.0)
```

Table 2 compares the old parameters to the new ones. Changes that you should note are:

- The section in the params.mtl file is called Electromagnetic Emissions instead of Sensor Volumes.
- The parameters are prefaced by `setq`, not `setqb`.
- Near and far colors are specified by one parameter each, rather than four. They use a hexadecimal number. Colors are specified in the same format as the overlay text color, described on page 6-6 of the *MÄK Stealth User's Guide*.
- The time-out value is off by default in HLA.
- The discussion of sensor volumes in the *MÄK Stealth User's Guide* uses DIS terminology. Where DIS terms are used, the equivalent HLA action takes place if you use electromagnetic emissions in HLA. For example, where the manual talks about sending an EM-Emission PDU, in HLA a state update message is sent.

Table 2: Electromagnetic emission parameters (Sheet 1 of 2)

Sensor Volume Parameters	Electromagnetic Emission Parameters
MySensorVolume::dfltFColor[0] 1. MySensorVolume::dfltFColor[1] 0.0 MySensorVolume::dfltFColor[2] 1.0 MySensorVolume::dfltFColor[3] 0.500000	setq emEmissionsDefaultFarColor 0x7fff00ff
MySensorVolume::dfltNColor[0] 1.0 MySensorVolume::dfltNColor[1] 1.0 MySensorVolume::dfltNColor[2] 1.0 MySensorVolume::dfltNColor[3] 0.500000	setq emEmissionsDefaultNearColor 0x7fffffff
MySensorVolume::hiFreq	setq emEmissionsHighFrequency
MySensorVolume::hiPRFreq	setq emEmissionsHighPrFrequency
MySensorVolume::initColorIndicates	setq emEmissionsColorIndicates
MySensorVolume::initMaxAngle	setq emEmissionsSegmentAngle
MySensorVolume::initRadiusExp	setq emEmissionsRadiusExponent
MySensorVolume::initRadiusSF	setq emEmissionsRadiusScaleFactor
MySensorVolume::initShowLines	setq emEmissionsShowLines
MySensorVolume::lifetime 100.0	(cond ((equal hla 1) (setq emEmissionsTimeoutInterval -1.0)) (t (setq emEmissionsTimeoutInterval 100.0)))

Table 2: Electromagnetic emission parameters (Sheet 2 of 2)

Sensor Volume Parameters	Electromagnetic Emission Parameters
MySensorVolume::loFreq	setq emEmissionsLowFrequency
MySensorVolume::loPRFreq	setq emEmissionsLowPrFrequency
stealthDisplayEmPdu	setq emEmissionsDisplayMode

Clip Textures (SGI only)

Clip textures allow you to use texture images that are too large to hold in memory. As the view changes, the appropriate parts of the database are loaded into memory. Support for clip textures is transparent. There is no configuration required.

Displaying Statistics in the 2D Overlay (PCs only)

The Overlay drop-down menu on the Stealth Options dialog box has a new option: Stats. If you select this option, the 2D overlay displays the frame rate and the number of polygons being displayed.

Support for Direct 3D

The Stealth for PCs now support the Direct3D graphics. When you install the Stealth, you must specify the type of graphics card that you have. The Stealth supports the following types of graphics cards:

- Glide
- OpenGL
- Direct3D.

New Default Directory Structure

The PC version of the Stealth has a new default directory structure. Instead of *C:\Program Files\MAK Technologies*, the Stealth is installed in *C:\Program Files\MAK*.

Documentation Updates

On page 7-9, the line that reads:

```
"Stealth -G 35:41:57.6,121:24:42.5,1250"
```

should read:

```
"Stealth -G 35.3321,121.6451,1250"
```

Specifying Cloud Position in Geocentric Mode

On page 6-30, the Stealth User's Guide describes how to configure cloud position using UTM coordinates (DtCigFrameKind = 0). In geocentric mode (DtCigFrameKind = 1), the cloud position values are interpreted as the offset in meters from the origin of the first terrain patch (provided in the terrain patch's *.info* file - see section 11.1.3). cloudOffsetX and cloudOffsetY refer to the offset in meters to the East and North, respectively. This is analogous to the UTM settings.

Fixed Bugs

The following problems in the Stealth for PCs have been fixed in this release:

- The Stealth Control Dialog and Info on Attached Entity Dialog Box reported incorrect position in the database coordinates text box. Position was all zeroes.
- Under certain circumstances, users could not switch out of the GVS display when it first comes up.
- Broadcast mode did not work.
- Geocentric mode did not work.

The following problems on both PCs and SGI have been fixed in this release:

- Configuration of cloud position in geocentric mode did not work.

Known Problems

- In the printed version of the *Stealth User's Guide*, some index references to Chapter 3 are off by one or two pages. If you cannot find the information you are looking for on the referenced page, scan the page or two before the referenced page. The index references in the PDF version of the manual are correct.
- If you create your own FED file to use with the Stealth for PCs, the file must be in UNIX format. That is, the lines must end with a line feed, not a carriage return.