

VR-Vantage

Users Guide

VR-Vantage Stealth
VR-Vantage PVD
VR-Vantage IG

VT MAK
A company of VT Systems



VR-Vantage

Users Guide

VR-Vantage Stealth
VR-Vantage PVD
VR-Vantage IG

Version 2.3

VT MAK
A company of VT Systems

Copyright © 2018 VT MAK
All rights Reserved. Printed in the United States.

Under copyright laws, no part of this document may be copied or reproduced in any form without prior written consent of VT MAK.

VR-Exchange™, VR-TheWorld™, VR-Vantage™, DI-Guy™, and DI-Guy Scenario™ are trademarks of VT MAK. MÄK Technologies®, VR-Forces®, RTIspy®, B-HAVE®, and VR-Link® are registered trademarks of VT MAK.

GL Studio® is a registered trademark of The DiSTI® Corporation.

Portions of this software utilize SpeedTree® technology (©2008 Interactive Data Visualization, Inc.). SpeedTree is a registered trademark of Interactive Data Visualization, Inc. All rights reserved.

SilverLining™ is a trademark of Sundog Software.

Terrain Profiles are based in part on the work of the Qwt project (http://qwt.sourceforge.net).

3ds Max® is a registered trademark of Autodesk, Inc.

All other trademarks are owned by their respective companies.

For third-party license information, please see "Third Party Licenses," on page xxx.

VT MAK
150 Cambridge Park Drive, 3rd Floor
Cambridge, MA 02140 USA

Voice: 617-876-8085
Fax: 617-876-9208

info@mak.com

www.mak.com

Revision VRV-2.3-1-180226



Contents

Preface

How the Manual Is Organized xxi
Documentation Set xxiv
MAK Products xxv
How to Contact Us xxviii
Document Conventions xxix
Mouse Button Naming Conventions xxx
Third Party Licenses xxx
Boost License xxxii
libXML and libICONV xxxii
Freefont OpenType Font Set xxxii
NVIDIA xxxiii
Third-Party Licenses for VR-Vantage Applications xxxiii

Section I Installation, Startup, and Concepts

Chapter 1. Introduction to VR-Vantage

1.1. VR-Vantage Features 1-2
1.1.1. Support for HLA, DIS, and CIGI 1-3
1.1.2. Realistic Display of Vehicles and Terrain 1-4
1.1.3. Multiple Attach Modes 1-4
1.1.4. Accurate Vehicle Positioning 1-4
1.1.5. eXaggerated Reality (XR) 1-5
1.1.6. Plan View Display 1-5
1.1.7. Terrain Agility and Composability 1-5
1.1.8. Dynamic Ocean 1-6
1.1.9. Lighting Effects 1-6
1.1.10. Streaming Video 1-6
1.1.11. Remote Control 1-6
1.1.12. DI-Guy 1-7
1.1.13. Flexible, Intuitive Graphical User Interface 1-7

1.2. Third-Party Software and Content 1-8
1.2.1. SilverLining 1-8
1.2.2. Triton Ocean SDK 1-9
1.2.3. GL Studio 1-9
1.2.4. SpeedTree 1-10
1.2.5. 3D Models, Terrain, and Graphical Content 1-10
1.2.6. OpenSceneGraph 1-11
1.2.7. osgEarth 1-11

Chapter 2. Installing VR-Vantage

2.1. Installing VR-Vantage 2-2
2.1.1. Installing VR-Vantage on Windows 2-2
2.1.2. Installing VR-Vantage on a Linux System 2-3
2.1.3. Uninstalling VR-Vantage on Windows 2-3
2.2. The VR-Vantage Directory Structure 2-4
2.3. Installing and Setting Up the MAK License Manager 2-5
2.3.1. Specifying the License Server 2-6
2.4. Installing an RTI 2-8
2.4.1. Installing the MAK RTI 2-9
2.5. Localizing the Graphical User Interface 2-10
2.5.1. Translating Other Interface Files 2-13
2.5.2. Applying the Language Files 2-13

Chapter 3. Running VR-Vantage Applications

3.1. Starting VR-Vantage Applications 3-2
3.1.1. Starting VR-Vantage on Windows 3-2
3.1.2. Starting VR-Vantage on Linux or the Windows Command Line 3-4
3.1.3. Command-Line Options 3-5
3.1.4. Shutting Down VR-Vantage 3-9
3.2. Introduction to the VR-Vantage Graphical User Interface 3-9
3.3. Opening a Terrain 3-10
3.3.1. Closing a Terrain 3-10
3.3.2. Reloading a Terrain 3-11
3.4. Connecting to a Simulation Network 3-11
3.4.1. Setting Up Auto Connect 3-13
3.4.2. Disconnecting from the Simulation Network 3-14
3.4.3. Using the Simulation Connections Toolbar 3-14
3.5. Managing VR-Vantage Settings 3-15
3.5.1. Synchronizing Settings Among VR-Vantage Installations 3-17
3.5.2. Global Settings and Observer-Specific Settings 3-17

Chapter 4. Managing Network Connections

4.1. Managing Simulation Connections 4-2
4.1.1. Displaying Connections 4-2
4.1.2. Adding a Simulation Connection 4-3
4.1.3. Renaming a Simulation Connection 4-4
4.1.4. Deleting a Simulation Connection 4-4

4.2. Configuring Connection Settings 4-4
4.2.1. HLA Connection Settings 4-5
4.2.2. DIS Connection Settings 4-6
4.2.3. CIGI Connection Settings 4-8

Chapter 5. Optimizing Performance

5.1. Factors that Affect Performance 5-2
5.2. Displaying Performance Statistics 5-3
5.2.1. Displaying the VR-Vantage Function Profiler 5-5
5.2.2. Displaying OSG Statistics 5-5
5.3. Preloading Models 5-6
5.3.1. Command-Line Options for Preloading Models 5-7
5.4. Clearing the Model Instancing Cache 5-7
5.5. Configuring VSync 5-8
5.6. Configuring Graphics Quality 5-8
5.6.1. Alpha to Coverage 5-9
5.7. Using Incremental Compiling with Streamed Data 5-11
5.8. Coloring Draw Calls 5-11
5.9. DI-Guy Performance Settings 5-12
5.10. Using the Developer Debug Overlay 5-13

Section II Display Engine, Video Streaming, Oculus Rift

Chapter 6. Windows, Channels, and Display Engines

6.1. The VR-Vantage Display Engine 6-3
6.1.1. How VR-Vantage Responds to Display Engine Connections 6-4
6.1.2. The Display Engine's Connection States 6-4
6.1.3. Windows and Channels 6-4
6.1.4. Window Types 6-4
6.2. Managing Display Engine Configurations 6-5
6.2.1. Adding a Window 6-5
6.2.2. Adding a Channel to a Window 6-7
6.2.3. Removing a Window 6-8
6.2.4. Removing a Channel 6-8
6.2.5. Saving a Display Engine Configuration 6-8
6.2.6. Loading a Display Engine Configuration 6-9
6.3. Changing a Window's Attributes 6-11
6.4. Changing a Channel's Attributes 6-12
6.4.1. Setting the Clipping Planes 6-14
6.4.2. Specifying the Projection Resize Policy Attribute 6-16
6.4.3. Changing a Channel's Frustum (Field of View) 6-19
6.4.4. Changing the Viewport 6-20
6.4.5. Configuring Water Visibility 6-21
6.4.6. Channel Keywords 6-22
6.5. Starting a VR-Vantage Display Engine 6-22
6.6. Connecting to a Display Engine 6-23
6.6.1. Disconnecting from a Display Engine 6-24
6.7. Configuring Multichannel Displays 6-24

6.7.1. Changing the Camera's Position and Orientation Offset 6-26
6.7.2. Creating a Multichannel Configuration 6-27
6.7.3. Loading a Multichannel Configuration 6-27
6.8. Stereoscopic Displays 6-28
6.8.1. Configuring Anaglyphic Stereo 6-29
6.8.2. Configuring Polarized Stereo 6-30
6.9. Taking Screen Captures 6-31
6.9.1. Specifying the Location of Screen Captures 6-31
6.10. Display Issues on Linux 6-32

Chapter 7. Display Engine Tutorials

7.1. Saving and Loading Display Engine Configurations 7-2
7.1.1. Connect a VR-Vantage Display Engine 7-2
7.1.2. Save Display Engine Configurations 7-5
7.1.3. Load the Display Engine Configurations in the GUI 7-7
7.1.4. Load Display Engine Configurations from the Command Line 7-8

Chapter 8. Using Streaming Video

8.1. Streaming Video 8-2
8.2. Starting a Video Stream 8-3
8.2.1. Setting Up Auto Start 8-4
8.3. Configuring Video Streams 8-4
8.3.1. Configuring GStreamer H.264 VLC Out 8-7
8.3.2. Configuring the Number of Capture Buffers 8-8
8.4. Viewing Streaming Video 8-9
8.4.1. Viewing Streaming Video Using VLC 8-10
8.4.2. Streaming Video to a Web Browser 8-13
8.5. Recording the VR-Vantage Display 8-14

Chapter 9. Using Head Mounted Displays for Virtual Reality

9.1. Introduction to Virtual Reality 9-2
9.1.1. Requirements 9-2
9.1.2. Resetting the Tracking Sensor 9-2
9.2. Running VR-Vantage with an HMD 9-3
9.2.1. Enabling and Disabling Virtual Reality Rendering 9-4
9.3. Measuring Performance 9-5
9.4. Configuration Options for Virtual Reality 9-6

Chapter 10. Using Multiple Projectors with VR-Vantage

10.1. Using Multiple Projectors 10-2
10.2. Hardware Requirements 10-2
10.3. Installing the Scalable Display Manager 10-2
10.3.1. Configuring the Master Computer 10-3
10.4. Running VR-Vantage with the Scalable Plug-in 10-10

Section III Scenes and Environmental Effects

Chapter 11. Scenes and Environmental Effects

11.1. Working with Scenes 11-2
11.1.1. Creating a Scene 11-2
11.1.2. Saving a Scene 11-2
11.1.3. Opening a Scene 11-2
11.2. Setting the Date and Time of Day 11-3
11.2.1. Setting the Time Scale 11-4
11.3. Specifying the Environment Conditions 11-5
11.3.1. Setting the Default Environment Condition 11-7
11.3.2. Adding an Environment Condition 11-7
11.3.3. Editing an Environment Condition 11-8
11.3.4. Deleting an Environment Condition 11-8
11.4. Configuring the Weather 11-9
11.4.1. Setting the Wind Speed and Direction 11-9
11.4.2. Setting Visibility (Fog) 11-10
11.4.3. Specifying Precipitation Type and Intensity 11-11
11.4.4. Specifying Cloud Cover 11-11
11.5. Configuring Marine Conditions 11-12
11.5.1. Enabling Marine Effects 11-13
11.5.2. Configuring Marine Conditions 11-14
11.5.3. Enabling Tidal Stream Wakes 11-16
11.5.4. Setting Ocean Quality 11-17
11.5.5. Configuring Breaking Waves 11-17
11.6. Displaying Water Droplets and Splash Effects 11-18
11.6.1. Enabling Sea Spray 11-20
11.6.2. Setting the Size of Water Droplets 11-21
11.6.3. Enabling High Quality Rotor Wash 11-21
11.7. Displaying Remote Environment Settings 11-22
11.8. Disabling Lighting and Clouds 11-22

Chapter 12. Using Sensors

12.1. Using Sensors 12-2
12.2. Enabling Sensors 12-3
12.3. Configuring Sensors 12-4
12.4. Adding New Sensors 12-6

Chapter 13. Lighting Effects

13.1. Displaying Lighting Effects 13-2
13.2. Enabling and Disabling Lighting 13-2
13.2.1. Configuring Ambient and Diffuse Lighting 13-3
13.2.2. Specifying the Emissive Lighting Scale 13-4
13.3. Displaying Dynamic Lighting 13-5
13.3.1. Daylight Control of Lights 13-6
13.4. Displaying Ocean Planar Reflections 13-7
13.4.1. Configuring Ocean Planar Reflections 13-8
13.5. Lens Flare 13-9

13.6. Displaying Crepuscular (Sun) Rays 13-11
13.7. Configuring the Skybox Cube Map 13-13
13.8. Flashlight Mode 13-13
13.9. Using High Dynamic Range Lighting 13-14
13.10. Displaying Shadows (3D Only) 13-16
13.10.1. Configuring Shadows 13-18

Section IV The Observer

Chapter 14. The Observer and Observer Modes

14.1. Switching Between the 2D and 3D Projections 14-2
14.2. Observer Modes 14-2
14.3. Changing the Observer Mode 14-3
14.4. Creating and Editing Observer Modes 14-4
14.4.1. Creating an Observer Mode 14-4
14.4.2. Deleting an Observer Mode 14-5
14.4.3. Editing an Observer Mode 14-5
14.4.4. Renaming an Observer Mode 14-5
14.5. Adding an Observer 14-6
14.5.1. Adding an Observer on the Observer Settings Page 14-6
14.5.2. Adding an Observer in the Display Engine Configuration Editor 14-7
14.6. Removing an Observer 14-9
14.7. Communicating Information About the Observer 14-9
14.7.1. Making the Observer Visible to Other Applications ... 14-10
14.7.2. Viewing and Attaching to Other Observers 14-11

Chapter 15. Moving the Observer

15.1. Moving the Observer 15-3
15.1.1. Using the Compass to View the Observer's Heading 15-4
15.1.2. Editing the Compass's Attributes 15-5
15.2. Observer Movement Functions and Keyboard Mappings 15-6
15.2.1. 3D Observer Movement Coordinate Systems 15-7
15.2.2. 2D Observer Coordinate System 15-7
15.2.3. The Attached Context and Observer Movement (3D Only) 15-8
15.3. Moving the Observer from the Keyboard in the 3D Projection 15-10
15.3.1. Navigation Options 15-12
15.3.2. Orbiting 15-13
15.3.3. Moving the Camera 15-13
15.4. Moving the Observer Using the Mouse (3D Projection) 15-14
15.4.1. Dragging the Terrain 15-14
15.4.2. Changing the Observer's Orientation with the Mouse 15-16
15.4.3. Dragging the View 15-16
15.4.4. Orbiting the Terrain with the Mouse 15-16
15.4.5. Teleporting the Observer 15-17
15.5. Moving the Observer in the 2D Projection 15-17

15.5.1. Moving the Observer from the Keyboard in the 2D Projection 15-18
15.5.2. Moving the Observer Using the Mouse in the 2D Projection 15-19
15.6. Zooming the Observer 15-19
15.6.1. Zooming the Observer in the 2D Projection 15-19
15.6.2. Zooming the Observer in the 3D Projection 15-20
15.6.3. Setting the Scale Factor for Paged LODs 15-21
15.6.4. Setting the Magnification LOD Angle for Paged LODs 15-22
15.7. Simulating Camera Jitter 15-23
15.8. Forcing Orientation North (Plan View Mode Only) 15-24
15.9. Enabling and Disabling View Constraints (3D Only) 15-25
15.10. Controlling Navigation Speed 15-25
15.10.1. Enabling or Disabling Speed Scaling 15-26
15.10.2. Changing the Observer's Movement Speed 15-27
15.11. Saving and Recalling Views 15-28
15.11.1. Recalling a View 15-29
15.11.2. Deleting Views 15-30
15.11.3. Loading Views Automatically When You Load a Terrain 15-30
15.11.4. Saving Views to a File 15-31
15.11.5. Loading Views 15-32
15.12. Controlling the Observer from Other Applications 15-32
15.12.1. Enabling the Processing of View Control Messages .. 15-32
15.12.2. Overriding the Global View Control Settings 15-33

Chapter 16. Attaching the Observer to Objects

16.1. Attaching the Observer to a Simulation Object or Prop 16-2
16.1.1. Attaching the Observer to Another Observer 16-3
16.1.2. Attaching to an Articulated Part 16-4
16.1.3. Attaching the Observer in Mimic Track or Tether Track Mode 16-4
16.1.4. Attaching the Observer in Mimic Location Track or Tether Location Track Mode 16-4
16.1.5. Filtering the Object Attachment List 16-5
16.1.6. Detaching from a Simulation Object 16-5
16.1.7. Hiding the Attached Model (3D Projection Only) 16-5
16.1.8. Attaching to Props 16-5
16.1.9. Improving Performance when Attaching to Ground Objects 16-6
16.2. Attach Modes in the 3D Projection 16-6
16.2.1. Absolute Mode 16-7
16.2.2. Follow Mode 16-7
16.2.3. Mimic Mode 16-8
16.2.4. Mimic Track Mode 16-9
16.2.5. Mimic Location Track Mode 16-9
16.2.6. Space Follow Mode 16-10
16.2.7. Tether Mode 16-11
16.2.8. Tether Track Mode 16-12

16.2.9. Tether Location Track Mode 16-12
16.2.10. Track Mode 16-13
16.3. Attach Modes in the 2D Projection 16-14
16.4. Attaching the Observer at Startup 16-15
16.4.1. Observer Attachment Tutorial 16-16
16.4.2. Binding Multiple Observers Tutorial 16-17

Chapter 17. Inset Views, Sensor FOV, and HUDs

17.1. Inset Views 17-2
17.2. Displaying Cockpit Displays 17-3
17.3. Displaying Instrument Panels 17-6
17.3.1. Configuring Entities to Use Instrument Panels 17-7
17.4. Displaying Sensor Views 17-8
17.4.1. Displaying a Sensor Field of View Cone 17-9

Chapter 18. Exaggerating Models and Terrain

18.1. Exaggerating the Scale of 3D Entity Models 18-2
18.2. Exaggerating the Terrain Scale 18-5
18.2.1. Enabling Terrain Scaling 18-6
18.2.2. Scaling the Terrain 18-7
18.2.3. Track Histories and Terrain Scaling 18-8

Section V Viewing Simulation Objects

Chapter 19. Viewing Simulation Objects

19.1. Selecting Simulation Objects 19-2
19.2. Displaying Entity Labels 19-3
19.2.1. Displaying Text Labels for 3D Models 19-5
19.2.2. Turning Entity Labels On and Off 19-6
19.2.3. Pinning 3D Text Labels to the Window 19-6
19.2.4. Displaying Labels for 2D Icons 19-7
19.2.5. Configuring Shadowing and Background Color for 2D Labels 19-9
19.3. Changing Force Colors 19-9
19.4. Displaying HAT Lines for Entities (3D Only) 19-10
19.5. Changing the Model Set 19-11
19.6. Simulation Object Icons 19-11
19.6.1. Changing the Size of 2D Icons 19-12
19.6.2. Rotating 2D Icons to a Simulation Object's Heading .. 19-13
19.7. Viewing Occluded 2D Icons 19-14
19.8. Hiding 3D Models 19-15
19.9. Display of Embarked Entities 19-15
19.10. Viewing Object Counts 19-15
19.11. Displaying Models for Remote Observers 19-16
19.12. Filtering Simulation Objects Using Interest Management . 19-16
19.12.1. Enabling Interest Management 19-17
19.12.2. Configuring Interest Management 19-17

Chapter 20. Displaying Units

20.1. Introduction to Units 20-2
20.2. Expanding and Collapsing Units 20-2
20.3. Displaying Ghosted Icons 20-5
20.4. Displaying Unit Icons and Bounding Volumes 20-8
20.4.1. Specifying the Color Scheme for Unit Bounding Volumes 20-10

Section VI Visual and Audio Effects

Chapter 21. Viewing Entity Effects

21.1. Using Ground Clamping 21-2
21.1.1. Configuring Ground Clamping 21-3
21.2. Trajectory Smoothing 21-4
21.2.1. Configuring Trajectory Smoothing 21-5
21.3. Displaying Trailing and Decal Effects (3D Only) 21-6
21.3.1. Specifying Models for Trailing Effects and Decal Effects 21-8
21.4. Displaying Track Histories 21-8
21.4.1. Configuring Track Histories 21-9
21.5. Viewing Fire and Detonation Effects 21-11
21.6. Viewing Fire and Detonation Lines 21-12
21.7. Displaying Tactical Smoke 21-13
21.8. Displaying Electromagnetic Emissions 21-14
21.9. Displaying Radio Communication Lines 21-15
21.9.1. Enabling Radio Communications Graphics 21-16
21.9.2. Configuring the Lifetime and Display Mode 21-17
21.9.3. Configuring the Color of Radio Communication Lines 21-18
21.10. Using the Depth of Field Effect 21-20
21.10.1. Configuring Depth of Field 21-22
21.11. Displaying Wakes and Spray Effects 21-23
21.12. Enabling Buoyancy for Surface Entities 21-24
21.13. Displaying Laser Designators 21-26

Chapter 22. Intervisibility Objects

22.1. Intervisibility (Line-of-Sight) 22-2
22.2. Types of Intervisibility 22-3
22.2.1. Point-to-Point Intervisibility 22-3
22.2.2. Intervisibility Fan 22-4
22.2.3. Entity Intervisibility 22-4
22.3. Intervisibility Objects can be Transient or Permanent 22-5
22.3.1. Creating a Transient Intervisibility Object 22-6
22.3.2. Creating a Permanent Intervisibility Object 22-7
22.4. Editing an Intervisibility Object 22-8
22.5. Configuring Intervisibility Lines and Fans 22-9
22.6. Pinning an Intervisibility Object to a Simulation Object 22-10
22.7. Showing and Hiding Intervisibility Lines 22-10

22.8. Displaying Intervisibility Line Information 22-11
22.9. Deleting Intervisibility Objects 22-12

Chapter 23. Sound Effects

23.1. Introduction to Sound Effects 23-2
23.2. Enabling Sound Effects 23-2
23.2.1. Setting the Sound Volume 23-3
23.2.2. Specifying the Range in which Sounds are Heard 23-3
23.3. Associating Sound Files with Entities and Events 23-4
23.3.1. Mapping a Sound File to an Entity 23-5
23.3.2. Mapping Sound Files to Fire and Detonation Events .. 23-8

Section VII Terrain

Chapter 24. Changing How the Terrain is Displayed

24.1. Displaying Feature Data (2D) 24-2
24.1.1. Displaying a List of Vector Features 24-4
24.1.2. Changing the Colors Used for Feature Data 24-5
24.2. Displaying Grid Lines 24-6
24.2.1. Configuring Grid Lines 24-7
24.3. Configuring Visible Surfaces (GDB Soil Types) 24-8
24.4. Coloring Terrain Based on Elevation 24-9
24.4.1. Configuring Elevation Color Settings 24-10
24.5. Displaying Contour Lines 24-11
24.5.1. Configuring Contour Lines 24-12
24.6. Displaying Models and Terrain in Wireframe Mode 24-13

Chapter 25. Composing Terrains

25.1. Terrain Agility and Composability 25-2
25.1.1. Supported File Formats 25-2
25.1.2. Feature Layers and Props 25-4
25.2. Creating a Composed Terrain 25-5
25.2.1. Considerations and Limitations for Building Terrains 25-5
25.2.2. Saving a Terrain 25-6
25.3. Adding Elevation Data (Terrain Patches) to a Terrain 25-7
25.4. Adding Images to a Terrain 25-9
25.4.1. Changing the Display Order of Raster Maps 25-13
25.5. Adding a Feature Layer 25-14
25.6. Adding a Dynamic Ocean Layer 25-15
25.6.1. Extracting Water Textures and Adding an Ocean Layer 25-16
25.6.2. Adding a Dynamic Ocean Layer Directly 25-17
25.6.3. Setting the Ocean LOD Elevation 25-18
25.6.4. Configuring the Ocean Height Map 25-18
25.7. Adding Props to a Terrain 25-21
25.7.1. Extracting Props from a Terrain Patch 25-21
25.7.2. Adding Props from a Feature Layer 25-23
25.7.3. Viewing a List of Props 25-25
25.7.4. Selecting Props 25-25

25.7.5. Setting the Opacity of Props 25-26
25.7.6. Changing a Prop's Type 25-27
25.7.7. Changing a Prop's Position or Orientation 25-27
25.7.8. Changing a Prop's Model Definition 25-27

Chapter 26. Streaming Data Using Earth Files

26.1. Earth Files 26-2
26.2. Connecting to Terrain Servers 26-2
26.2.1. Adding Terrain Server Connections 26-5
26.2.2. Connecting to Terrain Servers through a Proxy 26-5
26.3. A Simple Earth File 26-6
26.3.1. Including External Files into Earth Files 26-8
26.4. Supported Drivers 26-9
26.5. Loading Multiple DTED Files 26-9
26.6. Processing Streamed Data 26-10
26.7. Adding Feature Layers to an Earth File 26-11
26.7.1. Configuring Point Features 26-13
26.7.2. Configuring Linear Features 26-14
26.7.3. Configuring Areal Features 26-15
26.8. Using Cut-in Sites for High Resolution Insets 26-16
26.8.1. Using the Boundary Generation Tool 26-17
26.9. Extruded Buildings 26-18
26.10. Earth File Options 26-19
26.11. Streaming Buoys and Beacons 26-20
26.11.1. Specifying Model Definitions for Streamed Buoys 26-21
26.11.2. Modeling Buoys Using Texture Atlases 26-24
26.11.3. Configuring Streaming Beacons 26-25
26.11.4. Configuring Streamed Navigation Lights 26-26
26.11.5. Configuring Seasonal Buoys and Beacons 26-27
26.11.6. Streaming Daymarks 26-28
26.11.7. Streaming Topmarks 26-28
26.11.8. Generating Signal Sequences 26-29

Chapter 27. Using MetaFlight Terrains

27.1. Loading MetaFlight Terrains 27-2
27.2. Building Efficient MetaFlight Terrains for VR-Vantage 27-2
27.3. Processing MetaFlight Files for Use in VR-Vantage 27-3
27.4. Splitting Datasets into Individual MetaFlight Files 27-3
27.5. Converting Virtual Texture Datasets to Tiled Textures 27-4
27.5.1. The mftTool 27-4
27.5.2. Using the mftTool to Process MetaFlight Files 27-6
27.5.3. Convert Geometry Grid Datasets with Tiled Textures . 27-7
27.6. Convert Source Data into MEDF Format 27-7
27.7. Create an MTF File for the MetaFlight Terrain 27-8

Chapter 28. Configuring CDB Terrain Databases

28.1. Loading CDB Databases 28-2
28.2. Configuring CDB earth Files 28-2
28.2.1. Configuring the Elevation Dataset (001) 28-3

28.2.2. Configuring the Imagery Dataset (004) 28-5
28.2.3. Configuring the GeoSpecific (GS) Dataset (100) 28-6
28.2.4. Configuring the GeoTypical (GT) Dataset (101) 28-13
28.2.5. Configuring the Extended Imagery Dataset (900) 28-15
28.3. CDB Configuration Files 28-17

Chapter 29. Terrain Performance and Configuration

29.1. Configuring File Caching 29-2
29.1.1. Specifying the Location to Cache Terrain Server Data 29-2
29.1.2. Caching earth File Terrain Data 29-3
29.1.3. Caching earth File Terrain Data Offline 29-4
29.1.4. Enabling Texture Compression 29-5
29.1.5. Clearing the File Cache 29-5
29.2. Displaying DDS Textures Correctly 29-6
29.2.1. Flipping DDS Textures Globally 29-7
29.3. Preprocessing Paged Terrains 29-8
29.4. Configuring Paged and Streaming Terrains 29-8
29.5. Debugging Earth Files 29-9
29.6. Using Shader-based Effect Maps 29-10
29.6.1. Composite Effects Maps 29-12
29.6.2. Debugging Shaders 29-13
29.6.3. Reloading Shaders 29-14
29.7. Displaying Terrain Intersection Lines 29-15

Chapter 30. Terrain Tutorials

30.1. Creating a Terrain Server 30-2
30.1.1. Add a Map Element 30-2
30.1.2. Add Elevation Data to the Earth File 30-2
30.1.3. Add Imagery to the Earth File 30-3
30.1.4. Add Feature Data to the Earth File 30-3
30.1.5. Add Options to the Earth File 30-6
30.1.6. Save the Earth File 30-7
30.1.7. Load the Earth File and Save is as an MTF File 30-7
30.2. Extracting Props from a Terrain 30-8
30.2.1. Create a New Terrain and Add a Terrain Patch 30-9
30.2.2. Extract the Props 30-9
30.2.3. Change the Model Definition for a Prop 30-11

Section VIII Modeling Simulation Objects

Chapter 31. Schemas

31.1. Creating and Editing Schemas 31-2
31.2. Creating Schemas 31-3
31.3. Deleting Schemas 31-4
31.4. Copying Schemas 31-4
31.5. Adding a Parameter to a Schema 31-5
31.6. Deleting a Parameter from a Schema 31-6
31.7. Editing a Schema Parameter 31-6

Chapter 32. Model Definitions

32.1. Creating and Editing Model Definitions 32-2
32.2. Work Flow Issues Model Definitions 32-3
32.3. Creating a Model Definition 32-5
32.4. Deleting a Model Definition 32-6
32.5. Copying a Model Definition 32-7
32.6. Filtering the List of Model Definitions 32-8
32.7. Editing a Model Definition 32-8
32.7.1. Adding Parameters to a Model Definition 32-8
32.7.2. Editing a Model Definition Parameter 32-10
32.7.3. Deleting Parameters from a Model Definition 32-11
32.8. Saving Model Definitions 32-11
32.9. Importing Model Definitions 32-12
32.9.1. Importing Model Definitions from the Command Line 32-12
32.10. Exporting Model Definitions 32-13
32.10.1. Exporting Selected Model Definitions 32-13

Chapter 33. Element Definitions

33.1. Creating and Editing Element Definitions 33-2
33.2. How Entity, Unit, and Tactical Graphics Element Definitions are Organized 33-2
33.2.1. Leaf File Descriptions 33-3
33.3. Work Flow Issues for Creating and Editing Element Definitions 33-4
33.4. Creating an Element Definition 33-5
33.5. Editing an Element Definition 33-7
33.5.1. Adding a Visual Definition 33-7
33.5.2. Editing a Visual Definition 33-8
33.5.3. Adding an Attribute to a Visual Definition 33-11
33.5.4. Deleting an Attribute from a Visual Definition 33-12
33.6. Deleting an Element Definition 33-13
33.7. Saving Element Definitions 33-13
33.8. Importing Element Definitions 33-14
33.8.1. Importing a Hierarchy File 33-14
33.8.2. Importing and Merging Element Definitions 33-15
33.8.3. Importing Element Definitions from the Command Line 33-15
33.9. Exporting Element Definitions 33-16
33.9.1. Exporting Element Definitions to *.elem* Files 33-16
33.9.2. Exporting Entity, Unit, or Tactical Graphics Element Definitions 33-17

Chapter 34. Mapping Object Types to Element Definitions

34.1. Introduction to Object Type Mapping 34-2
34.2. Work Flow Issues for Object Type Mappings 34-2
34.3. Adding an Object Type Mapping 34-4
34.4. Editing an Object Type Mapping 34-5
34.5. Filtering the Element Definition List 34-7

34.6. Deleting an Object Type Mapping 34-7
34.7. Saving Object Type Mappings 34-8
34.8. Importing Object Type Mappings 34-8
34.9. Exporting Object Type Mappings 34-9
34.10. Starting without Default Object Type Mappings 34-9
34.11. How VR-Vantage Maps DI-Guy Models 34-10

Chapter 35. Wakes, 2D Icons, Cockpits, and Other Models

35.1. Configuring 2D Icons 35-2
35.1.1. Editing Font-Based 2D Icons 35-2
35.1.2. Using Images for 2D Icons 35-3
35.1.3. Adding Images for Simulation Object Icons 35-9
35.2. Adding New Cockpit Display Models 35-10
35.2.1. Installing a Cockpit DLL 35-11
35.2.2. Creating a Model Definition for a Cockpit 35-11
35.3. Configuring Wakes 35-14
35.3.1. Configuring Tidal Stream Wakes 35-16
35.4. Configuring Light Points 35-16
35.4.1. Configuring Light Point Type Models 35-18
35.4.2. Configuring Light Point Group Models 35-18
35.4.3. Adding a New Light Point Type 35-19
35.5. Adding Wind-based Controls to Models 35-20
35.6. Using Texture Atlases to Skin Models 35-21
35.6.1. Configuring Texture Mapping in the Model Definition 35-21
35.6.2. Metafiles for Texture Mapping 35-22
35.7. Flipping DDS Textures for a Model 35-23
35.8. Configuring SpeedTree Trees 35-24
35.8.1. Randomizing the Size of SpeedTrees 35-25
35.8.2. Displaying SpeedTrees in Planar Reflections 35-26
35.9. Best Practices for Creating Models for VR-Vantage 35-27
35.10. Compressing Model Files 35-28

Chapter 36. Configuring Emitter Volumes

36.1. Configuring Emitter Volumes 36-2
36.1.1. Configuring Emitter Volume Color 36-2
36.1.2. Controlling Emitter Volume Radius 36-3
36.1.3. Configuring Emitter Volume Segments 36-4

Chapter 37. Adding DI-Guy Models to VR-Vantage

37.1. Adding New DI-Guy Models to VR-Vantage 37-2
37.1.1. Copy DI-Guy Files to the Data Directory 37-2
37.1.2. Add Model Definitions and Element Definitions 37-2

Chapter 38. Model Tutorials

38.1. Adding a Model to VR-Vantage 38-2
38.1.1. Create a Model Definition 38-2
38.1.2. Add an Element Definition 38-5
38.1.3. Add a Model Mapping 38-8

38.1.4. Test the Model Mapping 38-10

Chapter 39. The WRM Specification (DIS Notes)

39.1. Introduction 39-2
39.2. The OpenFlight File Format 39-2
39.2.1. Node Name and Comment Fields 39-2
39.2.2. External References and Instancing 39-3
39.3. Modeling for DIS Interoperability 39-3
39.3.1. The DIS Attribute Lexicon (DAL) 39-3
39.3.2. DAL Keywords 39-4
39.3.3. Model Coordinate Systems 39-5
39.4. Articulated Parts 39-7
39.5. Entity Appearances 39-8
39.6. Animation 39-11
39.7. DIS Keyword Values 39-12

Section IX Graphics

Chapter 40. Tactical Graphics and Remote Graphics

40.1. Displaying Tactical Graphics 40-2
40.1.1. Overlays 40-3
40.1.2. Showing and Hiding Tactical Graphics 40-4
40.1.3. Showing and Hiding Tactical Graphics Labels 40-5
40.1.4. Displaying Height-Above-Terrain Lines for Vertices 40-5
40.2. Displaying Remote Graphics 40-6
40.2.1. Viewing a List of Remote Graphics 40-7

Section X VR-Vantage Support for CIGI

Chapter 41. Mapping CIGI Models and Components

41.1. Introduction to CIGI 41-2
41.1.1. The VR-Vantage CIGI Driver 41-2
41.1.2. CIGI Packet Support 41-3
41.1.3. Mapping CIGI Input to VR-Vantage 41-3
41.2. Mapping Entity Models for CIGI 41-4
41.2.1. Adding an Articulated Part Mapping to an Entity Model 41-5
41.3. Mapping CIGI Components 41-6
41.3.1. Component Control Message Formats 41-6
41.4. Mapping a Database 41-11
41.5. Mapping CIGI Views 41-12
41.5.1. Mapping a CIGI View to an Observer Mode 41-12
41.6. Prototypical Host – IG Configurations 41-13

Chapter 42. CIGI Host Emulator Tutorial

42.1. Using the CIGI Host Emulator (Windows Only) 42-2
42.1.1. Start the Host Emulator 42-3
42.1.2. Connect VR-Vantage to the CIGI Host Emulator 42-4

42.1.3. Create an Ownship Entity on the Host Emulator 42-5
42.1.4. Create a Simulated Entity on the Host Emulator 42-6
42.1.5. Change an Entity's Model State 42-6
42.1.6. Add an Effect 42-8
42.1.7. Change the Weather 42-9

Chapter 43. CIGI Utilities

43.1. Using asynchHostTest.py 43-2
43.2. Using logger.py 43-3
43.3. Using snooper.py 43-3
43.4. The pyCigi Module 43-5

Chapter 44. CIGI Packets Supported

44.1. CIGI Packet Support 44-2
44.1.1. Support for CIGI Weather Control Packet 44-9
44.2. User Defined CIGI Packet 44-9

Section XI Configuring the Graphical User Interface

Chapter 45. The VR-Vantage User Interface

45.1. Dockable Panels and Toolbars 45-2
45.1.1. Docking and Undocking a Panel 45-2
45.1.2. Docking and Undocking a Toolbar 45-3
45.1.3. Displaying and Hiding Panels and Toolbars 45-3
45.1.4. Restoring the Layout of Toolbars and Panels 45-4
45.2. Viewing the Window in Full Screen Mode 45-4
45.3. Using Context-Sensitive Menus 45-5
45.4. Common VR-Vantage Buttons 45-5
45.5. Specifying Display Units 45-6

Chapter 46. Creating and Editing Key Mappings

46.1. Introduction 46-2
46.2. The Key Mapping Editor 46-2
46.2.1. Binary Key Mappings 46-3
46.3. Editing a Key Map 46-3
46.3.1. Adding a Key Mapping 46-3
46.3.2. Changing a Key Mapping 46-5
46.3.3. Deleting a Key Mapping 46-5
46.4. Using Combined Key Mappings 46-6
46.5. Filtering the Function List 46-7
46.6. Creating a Key Map 46-7
46.7. Deleting a Key Map 46-7

MAK Products Glossary

Index

VR-Vantage Quick Reference Card 1



Preface

This manual is for persons who will install VR-Vantage and use it to view simulations. The manual assumes that you are familiar with basic administrative tasks and the graphical window environment for your operating system.

For the latest product information, please see release-specific documentation for your version of VR-Vantage.

How the Manual Is Organized

This manual is organized thematically in the following major sections:

- Installation, startup, and basic concepts.
  - Chapter 1, *Introduction to VR-Vantage*, describes the basic features of VR-Vantage.
  - Chapter 2, *Installing VR-Vantage*, explains how to install VR-Vantage software, the license manager, and an RTI.
  - Chapter 3, *Running VR-Vantage Applications*, explains how to start and stop VR-Vantage, load a terrain, and connect to a simulation.
  - Appendix 4, *Managing Network Connections*, explains how to configure network connections.
  - Chapter 5, *Optimizing Performance*, describes issues that may affect performance.
- The display engine, streaming video, and Occulus support:
  - Chapter 6, *Windows, Channels, and Display Engines*, explains how to add windows and channels and how to configure them. It also explains how to set up multi-channel displays.
  - Chapter 7, *Display Engine Tutorials*, shows how to create display engine configurations, save them, and load them.
  - Chapter 8, *Using Streaming Video* explains how to send the view in the VR-Vantage window to a video stream.

  - Chapter 9, *Using Head Mounted Displays for Virtual Reality*, explains how to set up VR-Vantage to work with the Oculus Rift head mounted display.
  - Chapter 10, *Using Multiple Projectors with VR-Vantage*, explains how to set up VR-Vantage to set up two PC's and two projectors to project a large, seamless, edge blended display of VR-Vantage.
- Scenes and environmental effects:
  - Chapter 11, *Scenes and Environmental Effects*, explains how to configure the environment, including date and time, clouds, and precipitation.
  - Chapter 12, *Using Sensors* explains how to view the terrain as if you had night vision goggles or an infrared view.
  - Chapter 13, *Lighting Effects* describes the lighting effects in VR-Vantage and how to configure them.
- The observer.
  - Chapter 14, *The Observer and Observer Modes*, explains how to create and change observer modes, how to add observers, and how to communicate information about the observer.
  - Chapter 15, *Moving the Observer*, explains how to navigate the observer through the 2D and 3D scenes.
  - Chapter 16, *Attaching the Observer to Objects*, explains attach modes and how to attach the observer to entities and props.
  - Chapter 17, *Inset Views, Sensor FOV, and HUDs*, describes several types of special secondary windows – inset views, instrument panels, and sensor fields of view. It also covers cockpit displays.
  - Chapter 18, *Exaggerating Models and Terrain* describes features that only apply to XR observer mode.
- Visual effects.
  - Chapter 19, *Viewing Simulation Objects*, explains how to select entities and how to view information about them, including entity labels, force coloring, and height-above-terrain lines.
  - Chapter 20, *Displaying Units* explains how to manage the display of aggregate entities.
  - Chapter 21, *Viewing Entity Effects*, describes how to display various types of entity-related effects, such as ground clamping and smoothing, and entity related graphics, such as trailing effects and decal effects.
  - Chapter 22, *Intervisibility Objects* explains how to create and configure intervisibility objects.
  - Chapter 23, *Sound Effects* explains how to enable and configure sound effects for entities and interactions.
- Terrain modeling.
  - Chapter 24, *Changing How the Terrain is Displayed* explains how to learn more about the terrain by displaying contour lines, grid lines, and coloring by elevation.
  - Chapter 25, *Composing Terrains*, explains how compose terrains in the GUI.

  - Chapter 26, *Streaming Data Using Earth Files*, explains some of the details of how to set up *.earth* files to configure terrain servers.
  - Chapter 27, *Using MetaFlight Terrains*, explains how to process MetaFlight files so that you can open them in VR-Vantage.
  - Chapter 28, *Configuring CDB Terrain Databases*, explains how to create earth files for CDB terrains.
  - Chapter 29, *Terrain Performance and Configuration*, describes strategies for improving terrain performance.
  - Chapter 30, *Terrain Tutorials*, shows how to configure a terrain server using local data.
- Object modeling.
  - Chapter 31, *Schemas*, explains how to create and edit schemas.
  - Chapter 32, *Model Definitions*, explains how to create and edit model definitions.
  - Chapter 33, *Element Definitions*, explains how to create and edit element definitions
  - Chapter 34, *Mapping Object Types to Element Definitions*, how to map obejct type enumerations to visual models.
  - Chapter 35, *Wakes, 2D Icons, Cockpits, and Other Models*, explains how to configure some specific model types, such as 2D icons, cockpits, and SpeedTrees.
  - Chapter 36, *Configuring Emitter Volumes*, explains how to configure sensor volume color and segment size.
  - Chapter 37, *Adding DI-Guy Models to VR-Vantage*, explains how to add new DI-Guy models.
  - Chapter 38, *Model Tutorials*, has tutorials that show how to add an entity model and how to add a SpeedTree.
  - Chapter 39, *The WRM Specification (DIS Notes)*, explains the model specification for articulated parts.
- Graphics.
  - Chapter 40, *Tactical Graphics and Remote Graphics* explains how to manage tactical graphics and remote graphics.
- CIGI.
  - Chapter 41, *Mapping CIGI Models and Components*, explains how to map CIGI models and components, which are different from the mappings used for DIS and HLA.
  - Chapter 42, *CIGI Host Emulator Tutorial*, shows how to use the CIGI host emulator with VR-Vantage.
  - Chapter 43, *CIGI Utilities*, describes a set of scripts that you can use to test your CIGI connection to VR-Vantage.
  - Chapter 44, *CIGI Packets Supported*, lists CIGI packets and the extent to which they are supported by VR-Vantage.

- Graphical user interface (GUI) configuration.
  - Chapter 45, *The VR-Vantage User Interface* describes some of the basic features of the VR-Vantage graphical user interface (GUI).
  - Chapter 46, *Creating and Editing Key Mappings*, explains how to use the Key Map Editor to create and edit key maps for keyboard navigation.

MAK Products Glossary defines DIS and HLA terms and terms specific to VR-Vantage and other MAK products.

The *VR-Vantage Quick Reference Card*, at the end of the manual, lists command-line options, key mappings, and mouse button mappings.

Documentation Set

Electronic versions of VR-Vantage documentation are in *vrvantagex.x/doc*. On Windows, the documentation is accessible from the VR-Vantage folder on the Start menu. The VR-Vantage documentation set is as follows:

- *VR-Vantage Users Guide* describes how to use the VR-Vantage applications.
- *VR-Vantage Release Notes* lists system requirements, release-specific requirements, new features and updates, bug fixes, and known problems.
- Online help. The online help, accessible from the Help menu, replicates most of the information in *VR-Vantage Users Guide*.
- First Experience Guides. First Experience Guides are brief pamphlets that provide a guided tour of the most important features of each application. They are primarily for new users and persons evaluating VR-Vantage.
- *VR-Vantage Developers Guide*. The developers guide provides a high level introduction to the APIs, and class documentation, which is generated from the header files. You can open the developers guide and class documentation from the Windows Start menu on the VR-Vantage Documentation submenu or by opening *./vrvantage2.3/doc/classdoc/index.html*.

MAK Products

VR-Vantage is a member of the VT MAK line of software products designed to streamline the process of developing and using networked simulated environments. The VT MAK product line includes the following:

- VR-Link® Network Toolkit. VR-Link is an object-oriented library of C++ functions and definitions that implement the High Level Architecture (HLA) and the Distributed Interactive Simulation (DIS) protocol. VR-Link has built-in support for the RPR FOM and allows you to map to other FOMs. This library minimizes the time and effort required to build and maintain new HLA or DIS-compliant applications, and to integrate such compliance into existing applications.

  VR-Link includes a set of sample debugging applications and their source code. The source code serves as an example of how to use the VR-Link Toolkit to write applications. The executables provide valuable debugging services such as generating a predictable stream of HLA or DIS messages, and displaying the contents of messages transmitted on the network.

- MAK RTI. An RTI (Run-Time Infrastructure) is required to run applications using the High Level Architecture (HLA). The MAK RTI is optimized for high performance. It has an API, RTIspy®, that allows you to extend the RTI using plug-in modules. It also has a graphical user interface (the RTI Assistant) that helps users with configuration tasks and managing federates and federations.
- VR-Forces®. VR-Forces is a computer generated forces application and toolkit. It provides an application with a GUI, that gives you a 2D and 3D views of a simulated environment.

  You can create and view entities, aggregate them into hierarchical units, assign tasks, set state parameters, and create plans that have tasks, set statements, and conditional statements. You can simulate using entity-level modeling, which focuses on the actions of individual people and vehicles, and aggregate-level modeling, which focuses on the interaction of large hierarchical units.

  VR-Forces also functions as a plan view display for viewing remote simulation objects taking part in an exercise. Using the toolkit, you can extend the VR-Forces application or create your own application for use with another user interface.

- VR-Vantage™. VR-Vantage is a line of products designed to meet your simulation visualization needs. It includes three end-user applications (VR-Vantage Stealth, VR-Vantage PVD, and VR-Vantage IG) and the VR-Vantage Toolkit.
  - VR-Vantage Stealth displays a realistic, 3D view of your virtual world, a 2D plan view, and an exaggerated reality (XR) view. Together these views provide both situational awareness and the big picture of the simulated world. You can move your viewpoint to any location in the 3D world and can attach it to simulation objects so that it moves as they do.

  - VR-Vantage IG is a configurable desktop image generator (IG) for out the window (OTW) scenes and remote camera views. It has most of the features of the Stealth, but is optimized for its IG function.
  - VR-Vantage PVD provides a 2D plan view display. It gives you the big picture of the simulated world.
  - SensorFX. SensorFX is an enhanced version of VR-Vantage that uses physics based sensors to view terrain and simulation object models that have been materially classified. It is built in partnership with JRM Technologies.
  - The VR-Vantage Toolkit is a 3D visual application development toolkit. Use it to customize or extend MAK's VR-Vantage applications, or to integrate VR-Vantage capabilities into your custom applications. VR-Vantage is built on top of OpenSceneGraph (OSG). The toolkit includes the OSG version used to build VR-Vantage.
- MAK Data Logger. The Data Logger, also called the Logger, can record HLA and DIS exercises and play them back for after-action review. You can play a recorded file at speeds above or below normal and can quickly jump to areas of interest. The Logger has a GUI and a text interface. The Logger API allows you to extend the Logger using plug-in modules or embed the Logger into your own application. The Logger editing features let you merge, trim, and offset Logger recordings.
- VR-Exchange™. VR-Exchange allows simulations that use incompatible communications protocols to interoperate. For example, within the HLA world, using VR-Exchange, federations using the HLA RPR FOM 1.0 can interoperate with simulations using RPR FOM 2.0, or federations using different RTIs can interoperate. VR-Exchange supports HLA, TENA, and DIS translation.
- VR-TheWorld™ Server. VR-TheWorld Server is a simple, yet powerful, web-based streaming terrain server, developed in conjunction with Pelican Mapping. Delivered with a global base map, you can also easily populate it with your own custom source data through a web-based interface. The server can be deployed on private, classified networks to provide streaming terrain data to a variety of simulation and visualization applications behind your firewall.
- DI-Guy™. The DI-Guy product line is a set of software tools for real-time human visualization, simulation, and artificial intelligence. Every DI-Guy software offering comes with thousands of ready-to-use characters, appearances, and motions. DI-Guy enables the easy creation of crowds and individuals who are terrain aware, autonomous, and react intelligently to ongoing events. Save time, money and create outstanding simulations with DI-Guy. The DI-Guy product line includes the following products:
  - The DI-Guy SDK. Embed the DI-Guy library in your real-time application and populate your world with lifelike human characters.
  - DI-Guy Scenario™. Author and visualize human performances in a rich, user-friendly graphical environment. Use DI-Guy Scenario as an end visualization application or save scenarios and load them into your DI-Guy SDK enabled application.

  - ECOSim. Enhanced Company Operations Simulation (ECOSim) is a company-level training simulation that teaches leaders how best to deploy troops, UAVs, convoys, and other assets. ECOSim focuses on ease-of-use, rapid scenario generation, runtime operator control, and realistic and reactive human simulation.
  - DI-Guy AI. Generate crowds of autonomous characters to quickly populate your worlds with hundreds and thousands of terrain-aware, collision avoiding DI-Guys. Used as a module on top of DI-Guy Scenario and DI-Guy SDK.
  - Expressive Faces Module. Enable DI-Guy characters to have faces that display emotion, eyes that look in directions and blink, and lips that sync to sound files.
  - DI-Guy Motion Editor. Create or customize motions to your particular needs in an easy-to-use graphical application.
- RadarFX. RadarFX is a client-server application that simulates synthetic-aperture radar (SAR). The server application, which is based on VR-Vantage and SensorFX, loads a terrain database and, optionally, connects to simulations. A client application requests SAR images from the server. RadarFX includes a sample client application.
- VR-Engage. VR-Engage is a multi-role virtual simulator that lets users play the role of a first person human character, a ground vehicle driver, gunner or commander, or the pilot of a fixed wing aircraft or helicopter. It incorporates the VR-Force simulation engine and the VR-Vantage graphics rendering capabilities.
- WebLVC Server. WebLVC Server implements the server side of the WebLVC protocol so that web-based simulation federates can participate in a distributed simulation. It is part of the WebLVC Suite, which includes the server and several sample applications that work with VR-Forces and VR-Vantage.

How to Contact Us

For VR-Vantage technical support, information about upgrades, and information about other MAK products, you can contact us in the following ways:

Telephone

Call or fax us at:	Voice:	617-876-8085 (extension 3 for support)
	Fax:	617-876-9208

E-mail

Sales and upgrade information:	info@mak.com
Technical support:	support@mak.com

Internet

MAK web site home page:	www.mak.com
License key requests:	www.mak.com/support/get-licenses
Product version and platform information:	www.mak.com/support/product-versions
For the free, unlicensed MAK RTI:	www.mak.com/support/bonus-material

Post

Send postal correspondence to:	VT MAK 150 Cambridge Park Drive, 3rd Floor Cambridge, MA, USA 02140

When requesting support, please tell us the product you are using, the version, and the platform on which you are running.

Document Conventions

This manual uses the following typographic conventions:

`Monospaced`	Indicates commands or values you enter.
`Monospaced Bold`	Indicates a key on the keyboard.
`Monospaced Italic`	Indicates command variables that you replace with appropriate values.
Blue text	A hypertext link to another location in this manual or another manual in the documentation set.
`{ }`	Indicates required arguments.
`[ ]`	Indicates optional arguments.
`|`	Separates options in a command where only one option may be chosen at a time.
(|)	In command syntax, indicates equivalent alternatives for a command-line option, for example, `(-h | --help)`.
/	Indicates a directory. Since MAK products run on both Linux and Windows PC platforms, we use the / (slash) for generic discussions of pathnames. If you are running on a PC, substitute a \ (backslash) when you type pathnames.
Italic	Indicates a file name, pathname, or a class name.
sans Serif	Indicates a parameter or argument.
➤	Indicates a one-step procedure.
Menu → Option	Indicates a menu choice. For example, an instruction to select the Save option from the File menu appears as: Choose **File → Save**.
	Click the icon to run a tutorial video in the default browser.
	Indicates supplemental or clarifying information.
	Indicates additional information that you must observe to ensure the success of a procedure or other task.

Directory names are preceded with dot and slash characters that show their position with respect to the VR-Vantage home directory. For example, the directory *vrvantagex.x2.3/doc* appears in the text as *./doc*.

Mouse Button Naming Conventions

An instruction to click the mouse button, refers to clicking the primary mouse button, usually the left button for right-handed mice and the right button for left-handed mice. The context-sensitive menu, also called a popup menu or right-click menu, refers to the menu displayed when you click the secondary mouse button, usually the right button on right-handed mice and the left button on left-handed mice.

Third Party Licenses

MAK software products may use code from third parties. This section contains summary license information and where required, specific license documentation required by these third parties.

The following libraries are covered by the GNU Lesser General Public License (LGPL) license:

- CIGI
- DevIL
- GEOS
- GStreamer
- LibIconV
- OPCODE
- OpenAL
- OSG
- OsgEarth
- Pthread
- Qt
- Qwt
- LibWebSockets.

The following libraries are covered by the ZLib license:

- Bullet Physics
- Zlib
- LibPNG
- LibSDL
- Minizip.

The following libraries are covered by the MIT license:

- Expat
- FreeGLUT
- GDAL
- GeoTiff
- HarfBuzz
- LibCURL
- LibSquish

- LibXML
- LUA
- LuaBind
- Proj.

The following libraries are covered by the BSD license:

- FreeType
- GLEW
- JPEG
- LibTiff
- Protobuf
- PCRE.

The following libraries are covered by the commercial licenses:

- SpeedTree
- Sundog Triton
- Sundog SilverLining
- FlexLM
- Gameware
- JRM Technologies SenSimRT and SigSimRT
- GLStudio
- OpenFlight
- CDB API.

The following libraries are covered by custom licenses:

- NVidia CG Toolkit
- FreeImage (FreeImage Public License)
- Boost (Boost Software License)
- Poco (Boost Software License).

COLLADA DOM is covered by the SCEA Shared Source license.

Thread Building Blocks is covered by the GPL license:

SQLLite is in the public domain.

Boost License

VR-Link, and all MAK software that uses VR-Link uses some code that is distributed under the Boost License. All header files that contain Boost code are properly attributed. The Boost web site is: www.boost.org.

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

libXML and libICONV

VR-Link and all MAK software that uses VR-Link, links in libXML and libICONV. On some platforms the compiled libraries and header files are distributed with MAK Products. MAK has made no modifications to these libraries. For more information about these libraries please see the following web sites:

- The LGPL license is available at: http://www.gnu.org/licenses/lgpl.html.
- Information about IconV is at: http://www.gnu.org/software/libiconv/.
- Information about LibXML is at: http://xmlsoft.org/.

Freefont OpenType Font Set

VR-Vantage applications and VR-Forces use the Freefont OpenType font set from the Free Software Foundation. It is covered by the General Public License (GPL). For details, please see: http://www.gnu.org/licenses/gpl.html

NVIDIA

The HDR implementation uses the HDR NVIDIA gameworks SDK example that has the following copyright notice:

Multiple Files in folder: es3aep-kepler\HDR/

SDK Version: v2.11

Email: gameworks@nvidia.com

Site: http://developer.nvidia.com/

Copyright (c) 2014-2015, NVIDIA CORPORATION. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of NVIDIA CORPORATION nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Third-Party Licenses for VR-Vantage Applications

VR-Vantage applications use a variety of third-party libraries. Developers who want to use these libraries may be required to purchase developer's licenses. Please see "Third-Party Software and Content," on page 1-8, in *VR-Vantage Users Guide* for details.



I. Installation, Startup, and Concepts

VR-Vantage is VT MAK's line of visualization products. It is designed to meet your needs for visualizing the simulated environment. The VR-Vantage product line consists of the VR-Vantage Toolkit and the following applications that are built with it:

- VR-Vantage Stealth. VR-Vantage Stealth is your 2D/3D information station. It has a 3D view for situational awareness, simulation debugging, or after action review; a 2D view to provide the "big picture" of a simulation; and an exaggerated reality (XR) mode to improve your understanding of the 3D view when the observer is not close to entities. The Stealth provides the most data about your networked virtual world, and presents it in a clear and accessible way.
- VR-Vantage IG. VR-Vantage IG is a configurable desktop image generator for customers who need an out-the-window (OTW) view or a picture from a remote camera. Vantage IG is designed for easy integration into desktop trainers and war gaming simulations.
- VR-Vantage PVD. VR-Vantage PVD provides a 2D plan view display (PVD) of terrain and simulated objects. Plan view displays provide the "big picture" of simulated environments. They sacrifice the situational awareness of the 3D environment, but let you see the entire simulated world at a glance. Entities are represented using MIL-STD 2525B icons that are color coded to represent their force.



- The VR-Vantage applications share many features in common. Therefore, for simplicity, in this manual we use the term VR-Vantage to refer generically to the VR-Vantage applications when we describe common features.
- Some screen captures of dialog boxes may show features that are not supported by all VR-Vantage applications. For example, VR-Vantage IG does not support intervisibility or grid lines, while VR-Vantage PVD does not support cockpit displays.

The VR-Vantage Toolkit is an API that comes with all the software you need to completely rebuild visual applications like VR-Vantage Stealth and allows you to customize and extend these applications to fit your unique requirements. The toolkit also gives you the power to embed any of VR-Vantage's capabilities directly into your simulation applications. The VR-Vantage Toolkit is based on OpenSceneGraph[1], so you can leverage value-added plug-ins built by the OSG community and MAK partners.

For details about the VR-Vantage Toolkit, please refer to the API documentation.

The chapters in this section get you started with VR-Vantage. They describe the features of VR-Vantage, how to install it, and how to start it. This section also provides some conceptual background and details about improving performance.

1. "OpenSceneGraph is an Open Source, cross-platform graphics toolkit for the development of high-performance graphics applications such as flight simulators, games, virtual reality and scientific visualization. It is based on the concept of a SceneGraph, providing an object-oriented framework on top of OpenGL." (http://www.openscenegraph.org/projects/osg/wiki/About/Introduction)



1. Introduction to VR-Vantage

This chapter introduces you to the main features of VR-Vantage.

VR-Vantage Features 1-2
- Support for HLA, DIS, and CIGI 1-3
- Realistic Display of Vehicles and Terrain 1-4
- Multiple Attach Modes 1-4
- Accurate Vehicle Positioning 1-4
- eXaggerated Reality (XR) 1-5
- Plan View Display 1-5
- Terrain Agility and Composability 1-5
- Dynamic Ocean 1-6
- Lighting Effects 1-6
- Streaming Video 1-6
- Remote Control 1-6
- DI-Guy 1-7
- Flexible, Intuitive Graphical User Interface 1-7

Third-Party Software and Content 1-8
- SilverLining 1-8
- Triton Ocean SDK 1-9
- GL Studio 1-9
- SpeedTree 1-10
- 3D Models, Terrain, and Graphical Content 1-10
- OpenSceneGraph 1-11
- osgEarth 1-11

1.1. VR-Vantage Features

VR-Vantage Stealth and IG deliver a realistic, three-dimensional view of terrain, vehicles, and other objects (Figure 1-1). VR-Vantage's navigation controls give you complete freedom to travel anywhere as an unobtrusive observer. You can place the observer at a specific location, or link it to a vehicle for automatic tracking of battlefield activities. While an exercise is underway, you can switch attach modes quickly to see the scene from any perspective. VR-Vantage can be visible to other applications or invisible, at your discretion.

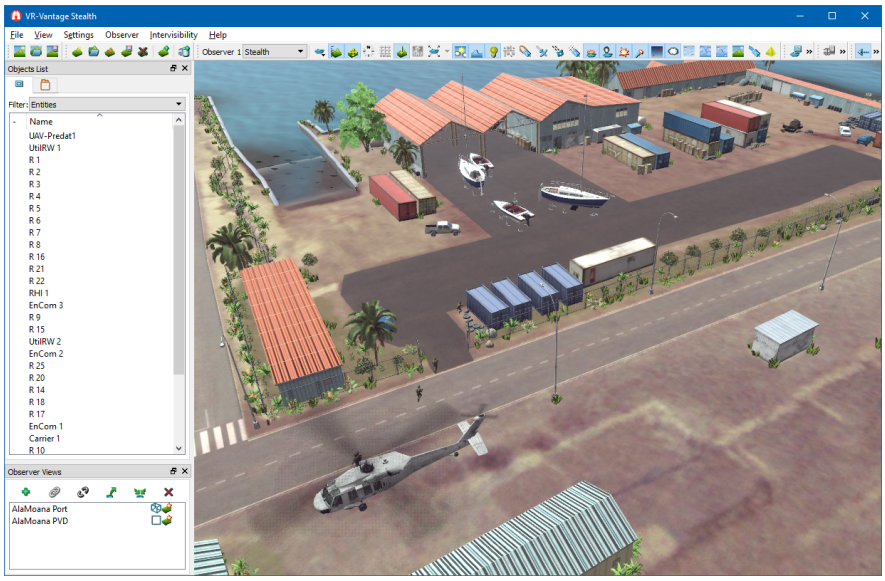


Figure 1-1. VR-Vantage Stealth (3D)

VR-Vantage Stealth and PVD provide a two-dimensional plan view-style display (Figure 1-2).

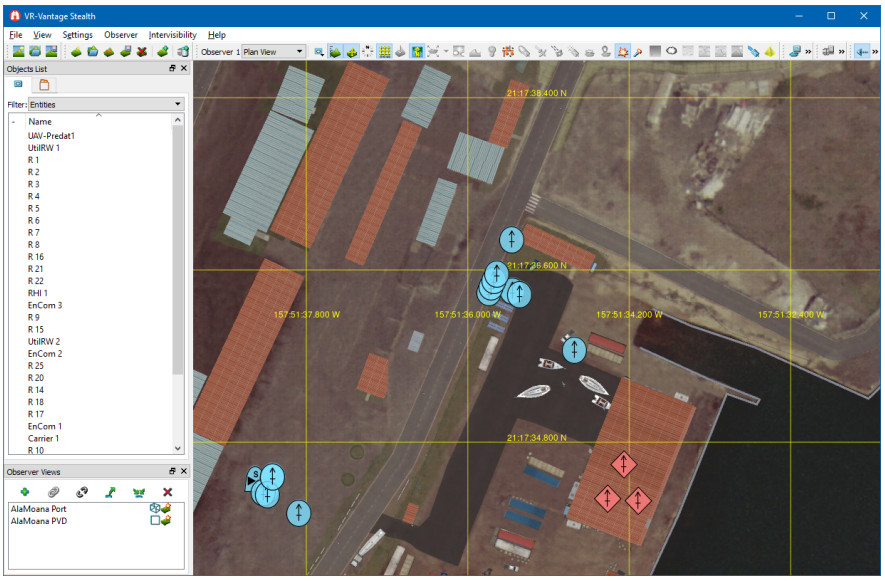


Figure 1-2. VR-Vantage Stealth (2D)

VR-Vantage provides a range of features that make it an essential tool during any phase of a Distributed Interactive Simulation (DIS), Common Image Generator Interface (CIGI), or High Level Architecture (HLA) project.

1.1.1. Support for HLA, DIS, and CIGI

You can connect and disconnect from simulations during runtime and can configure the connections in the graphical user interface (GUI). VR-Vantage supports the following simulation network protocols:

- HLA 1.3, HLA 1516, and HLA Evolved. VR-Vantage supports the HLA 1.3 specification, the current draft of the IEEE 1516 C++ API maintained by the SISO Dynamic Link Compatible RTI API Product Development Group, and HLA Evolved (IEEE 1516-2010). It has built-in support for the HLA RPR FOM and can support other FOMs through the FOM Mapping feature.
- Distributed Interactive Simulation (DIS) protocols DIS 4, 5, 6, and 7.
- Common Image Generator Interface (CIGI) 3.2 and 3.3. CIGI is an interface that allows an image generator (IG) to receive messages from a host to control what is drawn by the IG.

1.1.2. Realistic Display of Vehicles and Terrain

The VR-Vantage 3D projection provides a realistic display using terrain and moving-model geometry in the OpenFlight format. It includes many 3D vehicle models, some of which display movement of articulated parts, such as turrets and rotors. These models can change to show a damaged or destroyed state. You can provide your own models and map them to entity types. For details about terrain, please see Section VII, *Terrain*. For details about models, please see Chapter 35, *Wakes, 2D Icons, Cockpits, and Other Models*.

VR-Vantage also displays:

- Smoking and flaming effects.
- Trailing effects.
- Muzzle flashes.
- Track histories.
- Explosions.

VR-Vantage presents terrain realistically, and lets you control visibility, time-of-day, and other effects. For details, please see Chapter 11, *Scenes and Environmental Effects*.

1.1.3. Multiple Attach Modes

VR-Vantage offers attach modes that let you:

- Manually control the observer's position and orientation.
- Follow an entity, maintaining a consistent positional offset, with a matching heading.
- Mimic a vehicle's position and orientation. You can place the observer inside the vehicle for a driver's-eye view, or fly outside of it, moving in tandem with the vehicle.
- Tether the observer to an entity, maintaining a consistent positional offset without changing the observer's heading.
- Automatically track a vehicle's movement from a fixed viewpoint, as if watching from a control tower.

For more information, please see Chapter 15, *Moving the Observer*.

1.1.4. Accurate Vehicle Positioning

VR-Vantage can smooth the trajectories of moving vehicles to compensate for discontinuous positional data. The ground clamping feature can ensure that all ground entities are placed correctly on the terrain surface. For details, please see "Configuring Trajectory Smoothing," on page 21-5 and "Using Ground Clamping," on page 21-2.

1.1.5. eXaggerated Reality (XR)

The XR observer mode enhances your ability to understand the layout of forces in the 3D projection. The colorized notional icons are scaled to be visible regardless of the observer's distance from the entities. You can resize the icons to ensure that you have the right view for your needs. For more information, please see Chapter 18, *Exaggerating Models and Terrain*.

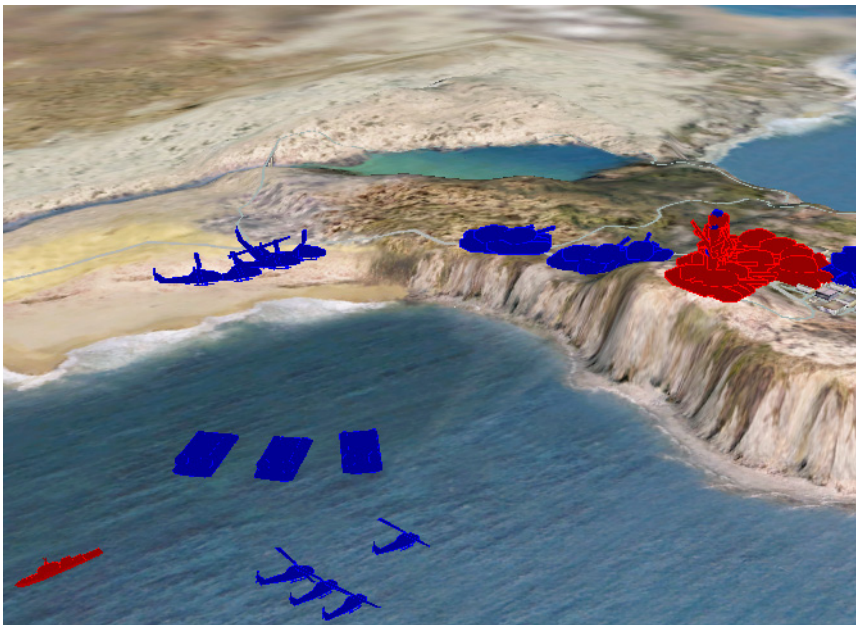


Figure 1-3. Model scaling and colorized models

1.1.6. Plan View Display

VR-Vantage PVD and the Plan View observer mode in VR-Vantage Stealth support many of the entity information features of the 3D projection plus features unique to a 2D view, such as grid lines. It supports the same terrain databases as the 3D projection.

1.1.7. Terrain Agility and Composability

VR-Vantage allows you to build your terrain at runtime using a variety of database and vector formats. It can load OpenFlight, DTED, CTDB, MetaFlight, CDB, and GDB (MAK terrain format) databases. It can load feature data from shapefiles and GDB files. You can also superimpose geospatial images such as GeoTiffs or other raster image files on the terrain. You can save these composed terrains in the VR-Vantage MTF file format. For details, please see Chapter 29, *Terrain Performance and Configuration*.

1.1.8. Dynamic Ocean

VR-Vantage provides dynamic ocean visualization in the Stealth and Out-the-Window observer modes. The ocean shows waves, swells, and spray effects. Surface entities have realistic wakes and buoyancy behavior. Changes to wind speed and direction affect wave action.

1.1.9. Lighting Effects

VR-Vantage supports a variety of lighting effects:

- Dynamic lighting
- Forward+ lighting
- Ocean planar reflection
- High dynamic range (HDR)
- Lens flare
- Crepuscular (sun) rays
- Shadows.

Using Shader-based Effects Maps

VR-Vantage supports several types of shader-based effects texture maps. Shaders are computer programs that simulate the effect of light on the terrain and objects in the simulation. VR-Vantage makes extensive use of them for visualization effects.

Texture maps are raster images that apply highly realistic textures to the terrain and models. By applying different types of texture maps to terrain and models, you can improve the visual qualities of your simulation without the overhead of high polygon counts.

To see the effect of effects maps, you must enable advanced lighting and shading. For more information about shader based effects maps, please see "Using Shader-based Effect Maps," on page 29-10.

1.1.10. Streaming Video

VR-Vantage can send the view in the display window to a video stream. It supports several different open standards. If you have a supported viewer, you can use the simulated video as a flexible alternative to live video for demonstration, development, testing, and embedded training of operational video exploitation systems.

1.1.11. Remote Control

You can control VR-Vantage remotely through view control messages, which can be generated by MAK applications such as the Data Logger and VR-Forces, and by programs that use the VR-Vantage Control Toolkit. For details, please see "Controlling the Observer from Other Applications," on page 15-32.

1.1.12. DI-Guy

VR-Vantage uses DI-Guy software and content for human character visualization.

VR-Vantage comes with DI-Guy functionality built-in and with a large set of DI-Guy characters and appearances,. A VR-Vantage customer does not need a DI-Guy license of any kind to use DI-Guy animated characters in a VR-Vantage-based application (including custom applications built using the VR-Vantage Toolkit).

However, you need a DI-Guy Developer's license if you want to:

- Develop new character types, or alter the geometry, appearances, textures, or motions of DI-Guy characters.
- Directly access the DI-Guy API to modify or extend the way that VR-Vantage uses the DI-Guy API or access DI-Guy functionality that MAK has not already integrated into VR-Vantage.

DI-Guy libraries, models, textures, motion files, and other graphics resources are distributed in the VR-Vantage package solely so that VR-Vantage can use them. Use of DI-Guy software or content outside of VR-Vantage-based applications is not permitted by the VR-Vantage license.

1.1.13. Flexible, Intuitive Graphical User Interface

VR-Vantage's graphical user interface (GUI) allows you to manipulate VR-Vantage from menus, dockable control panels, and the keyboard.

You can hide and display the control panels, keep them docked to the VR-Vantage window, or undock them to make more space available for the display window. The GUI controls can be hidden for full-screen demonstrations.

The GUI follows standard conventions for modern windowing systems. For additional details, please see Chapter 45, *The VR-Vantage User Interface*. For descriptions of individual windows and dialog boxes, please see online help.

1.2. Third-Party Software and Content

VR-Vantage includes software and content licensed from third parties, including:

- SilverLining™: real-time sky and 3D cloud rendering from Sundog Software.
- Triton Ocean SDK™: 3D ocean and ship wakes from Sundog Software.
- GL Studio®: interactive cockpit instrumentation and HMI from DiSTI®.
- SpeedTree®: animated, 3D foliage from Interactive Data Visualization (IDV).
- 3D models, terrain, and graphical content from Simthetiq, RealDB, TurboSquid, and TerraSim.
- OpenSceneGraph: an open source 3D graphics toolkit hosted at http://www.openscenegraph.org.
- osgEarth: an open source streaming terrain plug-in by Pelican Mapping, at http://osgEarth.org.

The run-time and developers' rights for VR-Vantage customers vary from vendor to vendor. Please read this section carefully to understand which rights are included with VR-Vantage licenses, and which rights require additional third-party licenses.

1.2.1. SilverLining

VR-Vantage uses SilverLining software and content to compute lighting and to render the sky, clouds, sun, and moon. SilverLining is developed by Sundog Software (http://www.sundog-soft.com).

Most VR-Vantage customers do not need to buy a SilverLining license of any kind. In general, a VR-Vantage customer has the right to use the SilverLining technology that is built into VR-Vantage in any VR-Vantage-based application (including custom applications built using the Toolkit). However, there are two exceptions:

- If you want to extend or change the way that MAK has integrated SilverLining into VR-Vantage by writing directly to the SilverLining API, then you need a SilverLining Developer's License.
- If you are building a commercial video game or other "mass-market" (consumer-level) commercial product on top of the VR-Vantage Toolkit, and the resulting product includes the SilverLining functionality, then you need a SilverLining Developer's License.

SilverLining libraries, textures and other graphics resources are distributed in the VR-Vantage package solely so that VR-Vantage can use them. Use of SilverLining software or content outside of VR-Vantage-based applications is not permitted by the VR-Vantage license.

1.2.2. Triton Ocean SDK

VR-Vantage uses the Triton Ocean SDK to render 3D oceans and ship wakes. Triton Ocean is developed by Sundog Software (http://www.sundog-soft.com).

Most VR-Vantage customers do not need to buy a Triton Ocean license of any kind. In general, a VR-Vantage customer has the right to use the Triton Ocean technology that is built into VR-Vantage in any VR-Vantage-based application (including custom applications built using the Toolkit). However, there are two exceptions:

- If you want to extend or change the way that MAK has integrated Triton Ocean into VR-Vantage by writing directly to the Triton Ocean API, then you need a SilverLining Developer's License.
- If you are building a commercial video game or other "mass-market" (consumer-level) commercial product on top of the VR-Vantage Toolkit, and the resulting product includes the Triton Ocean functionality, then you need a Triton Ocean Developer's License.

Triton Ocean libraries, textures and other graphics resources are distributed in the VR-Vantage package solely so that VR-Vantage can use them. Use of Triton Ocean software or content outside of VR-Vantage-based applications is not permitted by the VR-Vantage license.

1.2.3. GL Studio

VR-Vantage uses GL Studio software and content to render interactive cockpit instrumentation displays. GL Studio is developed by DiSTI (http://www.disti.com).

VR-Vantage is delivered with several generic cockpit displays. You can use the built-in cockpit displays in any VR-Vantage-based application (including custom applications built using the Toolkit), without a GL Studio license of any kind.

If you want to build custom cockpits using the GL Studio editor or GL Studio API, you need a GL Studio Developer's license. If you want to execute custom cockpits in VR-Vantage-based applications (regardless of whether the custom cockpits are built by you or a third party), you need GL Studio Run-Time licenses.

GL Studio libraries and graphics resources are distributed in the VR-Vantage package solely so that VR-Vantage can use them. Use of GL Studio software or content outside of VR-Vantage-based applications is not permitted by the VR-Vantage license.

1.2.4. SpeedTree

VR-Vantage uses SpeedTree software and content for animated real-time 3D foliage and vegetation. SpeedTree is developed by Interactive Data Visualization (IDV) (http://www.speedtree.com).

A VR-Vantage customer does not need a SpeedTree license of any kind to use the SpeedTree functionality that is built into the VR-Vantage applications that MAK delivers. You may build and run plug-ins to our out-of-the-box applications without a SpeedTree license, as long as your plug-ins do not need to directly access the SpeedTree API.

However, you need a SpeedTree Developer's license if you want to:

- Write a plug-in that uses the SpeedTree API directly to extend or modify the way that SpeedTree is integrated into VR-Vantage.
- Use the VR-Vantage Toolkit to develop new applications, and you want your applications to include SpeedTree functionality.

SpeedTree libraries, models, textures, and other graphics resources are distributed in the VR-Vantage package solely so that VR-Vantage can use them. Use of SpeedTree software or content outside of VR-Vantage-based applications is not permitted by the VR-Vantage license.

1.2.5. 3D Models, Terrain, and Graphical Content

VR-Vantage includes a rich set of 3D models for vehicles, weapons, cultural features and urban clutter (signs, barriers, lampposts, and so on). It also includes several sample terrain databases to help you get started with VR-Vantage, and to help demonstrate VR-Vantage. Much of this content is derived from 3D data that we have licensed from third parties:

- Many of the high-quality vehicle models come from Simthetiq (http://www.simthetiq.com).
- Some models come from Real DB (http://www.realdb.qc.ca/company).
- Some of the middle-eastern building models and urban clutter objects that are used in our sample terrain databases were licensed from Turbosquid (http://www.turbosquid.com).

All of this content is distributed in the VR-Vantage package solely so that VR-Vantage can use it. Use of any of the VR-Vantage models, textures, terrains, or other content outside of VR-Vantage-based applications is not permitted by the VR-Vantage license.

1.2.6. OpenSceneGraph

VR-Vantage uses OpenSceneGraph for its underlying scene graph representation, rendering, file loading, and many other capabilities. OpenSceneGraph is an open source, cross-platform graphics toolkit for the development of high-performance graphics applications. The OpenSceneGraph source repository is maintained by Robert Osfield at http://www.openscenegraph.org/projects/osg. It is distributed under the OpenSceneGraph Public License (OSGPL), which is based on the GNU Lesser General Public License (LGPL).

MAK will provide modified source code for OSG upon request, as per the OSGPL license. Please contact support@mak.com to obtain the download links for our modified source.

1.2.7. osgEarth

VR-Vantage uses osgEarth to import streaming terrain elevation and imagery data. The data can be streamed from external servers and sources or from a directory on the computer running VR-Vantage. osgEarth is an open source plug-in to OpenSceneGraph, maintained by Pelican Mapping at http://osgEarth.org. osgEarth is distributed under the GNU Lesser General Public License (LGPL).

MAK will provide modified source code for osgEarth upon request, as per the OSGPL license. Please contact support@mak.com to obtain the download links for our modified source.



2. Installing VR-Vantage

This chapter explains how to install the VR-Vantage applications, the VR-Vantage Toolkit, and data. You must also install the MAK license manager. To use VR-Vantage with HLA, you must install an RTI.

Installing VR-Vantage 2-2
- Installing VR-Vantage on Windows 2-2
- Installing VR-Vantage on a Linux System 2-3
- Uninstalling VR-Vantage on Windows 2-3

The VR-Vantage Directory Structure 2-4

Installing and Setting Up the MAK License Manager 2-5
- Specifying the License Server 2-6

Installing an RTI 2-8
- Installing the MAK RTI 2-9

Localizing the Graphical User Interface 2-10
- Translating Other Interface Files 2-13
- Applying the Language Files 2-13

2.1. Installing VR-Vantage

A full VR-Vantage installation includes three components: the application, the License Manager software, and for HLA operation, an RTI (or Runtime Infrastructure.)

Before you install VR-Vantage, please read the Release Notes to see if there are any special instructions for installation.

VR-Vantage has an application installer and a data package (*.MAK-targz*) that has multiple files. You must run the application installer and then install the data package. The application installer can automatically install the data package if you configure it to do so. MAK also provides optional data packages that have additional terrains.

To install the VR-Vantage toolkit, you must enter an installation code. Be sure you have received your code from your MAK salesperson before you start the installation process. If you do not have an installation code, you can install the runtime files. When you receive your installation code you can install the toolkit.

2.1.1. Installing VR-Vantage on Windows

When you install large applications on Windows, there may be a delay of up to several minutes from the time you try to run the setup program to the time that an installation dialog box is displayed. This is due to how Windows scans setup programs before it executes them. If you experience this problem, turning off User Access Control can reduce or eliminate this delay.

Windows versions of VR-Vantage are provided as executable installer files on DVD, or as downloaded files. The installers are named to indicate the compiler used to build that version of VR-Vantage.

➤ To install VR-Vantage, run the application installer. Follow the instructions in the installation wizard. You can specify that the application installer automatically find and install the data package (it must be in the same directory as the application installer), or you can specify that the MAK Data Installer open and let you select the data package to install.

2.1.2. Installing VR-Vantage on a Linux System

Linux versions of VR-Vantage are provided as compressed tar files on DVD, or as downloaded files.

To install VR-Vantage on Linux:

1. Create the directory in which you want to install VR-Vantage.
2. Copy the tar file to the install directory.
3. Uncompress and untar the application file:

   `tar -vxzf application.tar.gz`

   where *application* is the product release identification.
4. The data package consists of multiple files that have the extension *MAK-targz*. Uncompress and untar the data package files and copy them into the installation directory, or uncompress them and redirect the output, as follows:

   `tar -xzf makData<version>.MAK-targz -C /path-to-stealth2.3`

2.1.3. Uninstalling VR-Vantage on Windows

When you uninstall VR-Vantage, the uninstaller does not uninstall the data package. You must delete the files manually.

2.2. The VR-Vantage Directory Structure

The installation process creates the VR-Vantage directory tree shown in Figure 2-1 (runtime and toolkit). Table 2-1 describe the contents of the directories.

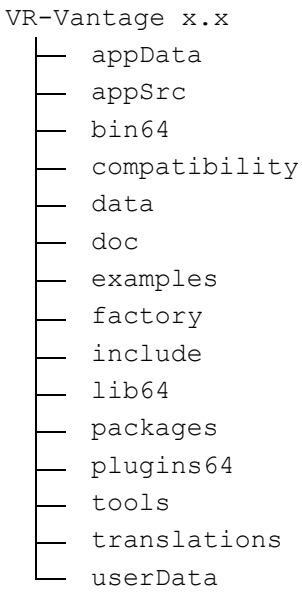


Figure 2-1. VR-Vantage directory tree

Table 2-1 describes the contents of the directories.

Table 2-1: Contents of VR-Vantage directories

Directory	Directory contents
./appData	Subdirectories contain data required by the VR-Vantage applications, such as model definitions and saved settings.
./appSrc	Source code for rebuilding the VR-Vantage applications.
./bin64	Executables and dynamic link libraries.
./compatibility	Scripts for updating include files to be compatible with newer versions of VR-Link.
./data	Sample databases, images, and vehicle models.
./doc	VR-Vantage documentation and class reference files.
./examples	Examples for the VR-Vantage Toolkit API.
./factory	Default settings (for reverting to “factory” settings).
./include	Header files for the VR-Vantage Toolkit API.
./lib64	Libraries for building applications with the VR-Vantage Toolkit API.

Table 2-1: Contents of VR-Vantage directories

Directory	Directory contents
./packages	Header files and libraries for third party and supplementary APIs that developers may need to build against.
./plugins64	Plug-in shared libraries.
./tools	Helper applications such as the CIGI Host Emulator.
./translations	Translation files for localizing VR-Vantage menus.
./userData	Terrain and scene files saved by application users.

2.3. Installing and Setting Up the MAK License Manager

Before you can use a MAK product, you must obtain a valid license file, install the MAK License Manager, and configure the license server and client machines. The License Manager uses a client-server architecture, so you do not need to install the License Manager on every computer on which you install MAK products. You only need to install it on the computer that you will use as the license server.



If you have already installed the License Manager for another MAK product, you do not have to install it again. You just need to make sure you have licenses for your newly installed products.

The License Manager installer is included on MAK installation media. It is separate from the product installers. You can download the installers from our web site at https://www.mak.com/support/license-support or you can download directly from:

ftp://ftp.mak.com/out/MAKLicenseManager-win64-setup.exe

Complete installation and configuration instructions are included with the License Manager installer. Instructions are also available at the license support page.

Some customers use dongle licenses instead of running a license server. Instructions for using dongles are in the License Manager documentation.

2.3.1. Specifying the License Server

The first time you run a MAK application on a particular computer, the License Setup dialog box opens (Figure 2-2). It prompts you to enter the hostname of the license server and optionally, a port number.

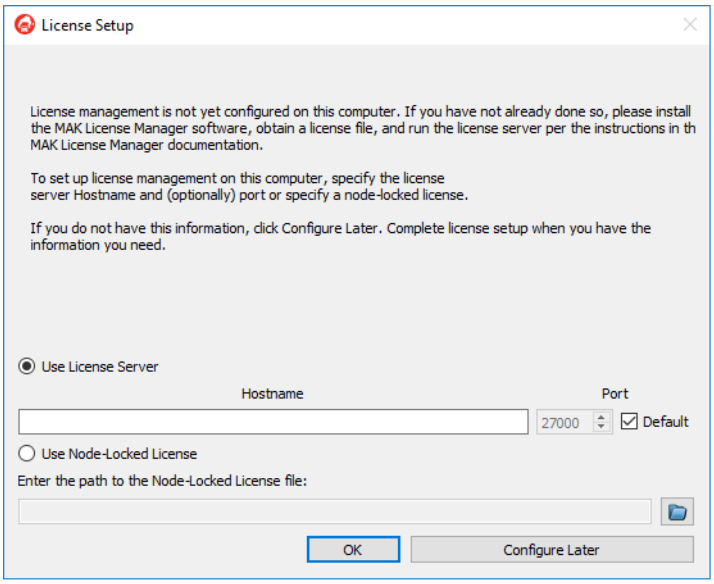


Figure 2-2. License Setup dialog box

If you do not know the hostname of the license server, click Configure Later. When you have the hostname, you can start the application again and complete the dialog box. You will not be able to run any MAK applications until you set up license management.

If you know the hostname, type it in the Hostname box. Then click OK. The application will start.

Under limited circumstances, MAK issues node-locked licenses. If you have a node locked license, select the Use Node Locked License option and enter the path to the license file.

- If you are running MAK products on the license server machine, it is also a client, so you must specify the license server on that machine too.
- If you change the license server, the saved configuration will no longer be valid and the License Setup dialog box will open the next time you start a MAK application.
- You can clear the saved license configuration by deleting the cache file. On Windows, it is *C:\Users\user_name\AppData\Roaming\MAK\licenses<n>.txt*. (The *AppData* directory is hidden by default.) On Linux, it is *.mak/license<n>.txt*. (There may be more than one cache file, for example, *licenses1.txt* and *licenses2.txt*.)

The MAKLMGRD_LICENSE_FILE Environment Variable

As an alternative to using the License Setup procedure described in the previous section, you can configure the license server in an environment variable. The MAKLMGRD_LICENSE_FILE environment variable identifies the server machine. If you set this environment variable, it overrides the settings stored by the License Setup procedure.

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 MAK products run on.

Windows

To add the MAKLMGRD_LICENSE_FILE in Windows:

1. Open the Control Panel.
2. Click System and Security.
3. Click System.
4. In the sidebar menu, click the Advanced System Settings. The System Properties dialog box opens.
5. Click Environment Variables. The Environment Variables dialog box opens.
6. Click New. The New System Variable dialog box opens.
7. In the Variable Name field, enter MAKLMGRD_LICENSE_FILE.
8. In the Variable Value field, enter `@server_name`, where `server_name` is the name of the license server.
9. Click OK to back out of each dialog box and set the variable.

Different versions of Windows have slightly different wording of the various Control Panel options.

Linux

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

```
setenv MAKLMGRD_LICENSE_FILE @oak
```

If you are using the sh or bash shells, you set environment variables in your *.profile* file (or *.bashrc*). Set the variable similarly to the following example:

```
MAKLMGRD_LICENSE_FILE=@oak
export MAKLMGRD_LICENSE_FILE
```

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

You are ready to run the license server and use your new licenses or MAK products.

2.4. Installing an RTI

An RTI is a software library (and perhaps supporting executables) that implements an HLA Interface Specification. In HLA, applications exchange FOM data through RTI calls, which means that all HLA applications need to use an RTI.



Because of differences in the low-level network mechanisms used by different RTI implementations (which include, but are not limited to packet layout), applications that want to interoperate in the same federation execution must use the same RTI implementation.

Because RTIs are usually provided as dynamic libraries that implement a fixed API, a federation can often switch from one RTI implementation to another between runs (without even recompiling the applications), but during each run, all participants must agree on which RTI to use, much as they must also agree on which FOM to use.

For the most recent information about the RTI versions supported by MAK products, please see the release notes for your MAK application.

2.4.1. Installing the MAK RTI

To install the MAK RTI, follow the instructions in Chapter 2 of *MAK RTI Users Guide*.

Configuring Your System to Use the MAK RTI

The RTI dynamic libraries must be located somewhere on your dynamic library search path. The path is specified in the PATH environment variable.The path is indicated by an environment variable as follows:

- Linux — LD_LIBRARY_PATH.
- Windows — PATH.

The MAK RTI needs to know where to find the following configuration files:

- The federation configuration file (*.fed*, *.fdd*, or *.xml*) for your federation execution (required).
- The RID file (*rid.mtl*) (optional).

Put the configuration files in the directory from which you are running, or set the environment variable RTI_CONFIG to the directory that contains them.

Running Applications with the MAK RTI

To run a MAK application with the MAK RTI:

1. Be sure the license server is running.
2. Start the application. The RTI Assistant will prompt you to choose an RTI configuration.
3. Choose a configuration. If necessary start the rtiexec.
4. Click Connect. The application should run.

In many cases, you do not need to run the rtiexec to use the MAK RTI, however you can run the rtiexec if you want to. (It is required to use certain features of the MAK RTI.) For more information, please see your RTI documentation.

2.5. Localizing the Graphical User Interface

You can localize your MAK product's graphical user interface so that it uses languages other than English. Your product includes a file called *./translations/product_untranslated.ts*, for example *logger_untranslated.ts* or *vrforces_untranslated.ts*. The file contains entries for all of the application-specific menu and dialog box text. The Qt Linguist tool, which is included with VR-Vantage, lets you map the English text provided by MAK to your chosen language.

To edit a resource file:

1. Start the Qt Linguist utility. It is *./translations/linguist.exe*
2. In the Qt Linguist window, choose **File** → **Open**. Select the file *./translations/product_untranslated.ts*. The Settings for filename dialog box opens (Figure 2-3).

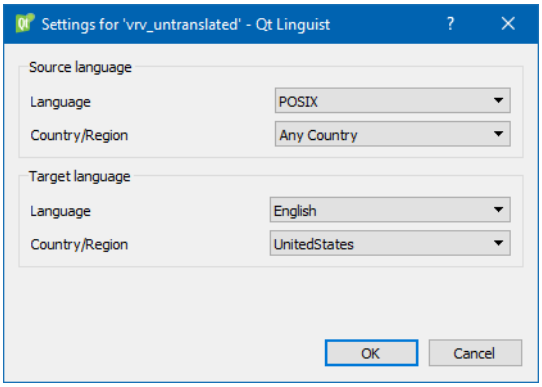


Figure 2-3. Linguist settings dialog box

3. Select values for the source language, target language, and target region or country.
4. Click OK. The translation file is opened and a list of classes is displayed on the left side of the window (Figure 2-4).

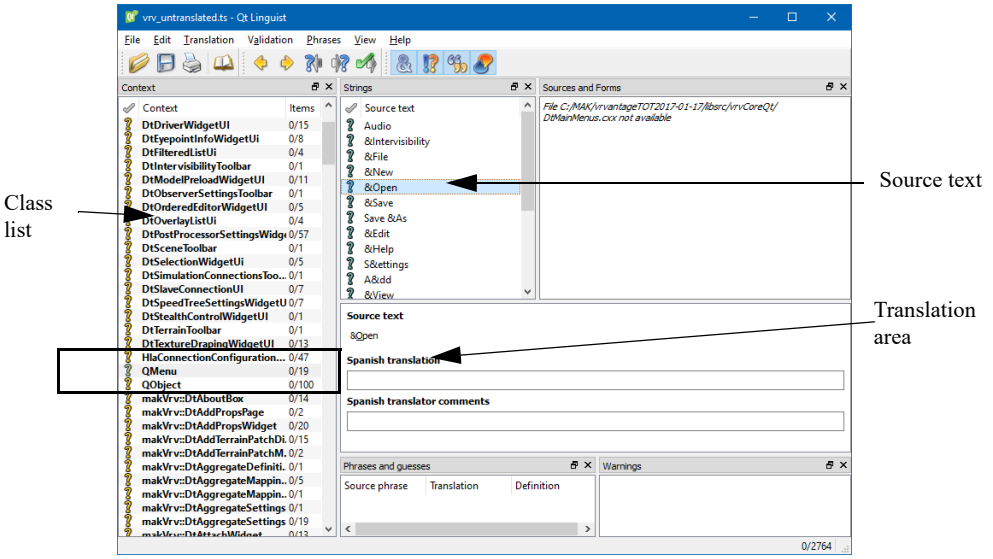


Figure 2-4. Linguist window

5. To translate text, you select each class in turn and translate the text associated with that class. You do not have to understand anything about what the classes mean, you just have to translate the text. In Figure 2-4, *QMenu* is selected. The Strings window lists text that can be translated. Text that has not been translated has a question mark in the checkmark column.

6. In the Source Text list, select the word you want to translate. The text is displayed under Source Text in the translation area (the window below the source text list.) Figure 2-5 shows the text Open selected.

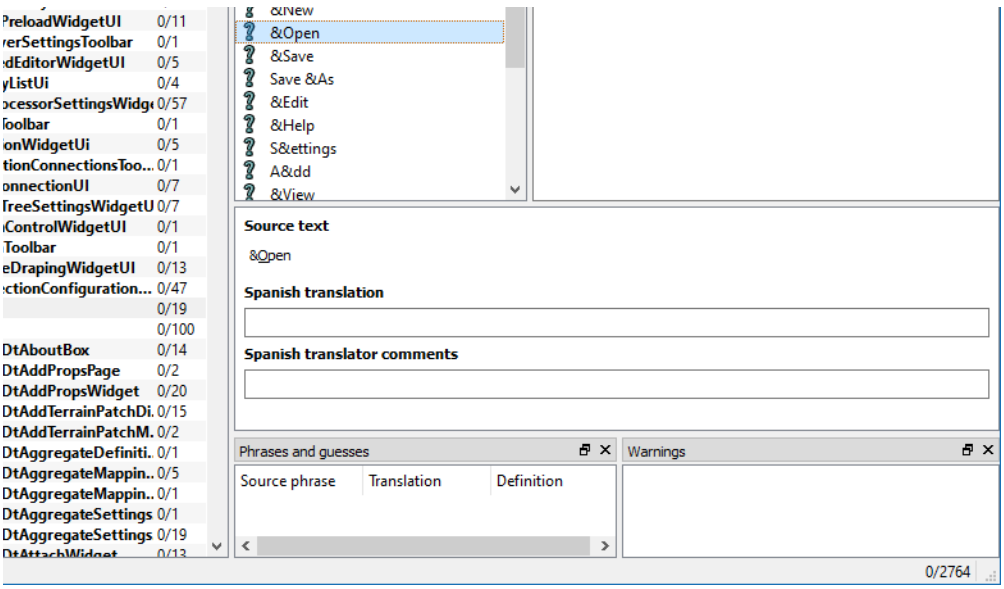


Figure 2-5. Text selected for translation

7. Place the cursor in the text box below *Language* Translation, where *Language* is the target language that you selected when you opened the translation file.
8. Type the translation text.
9. When you are done and you are certain of the translation, select the Done and Next button on the toolbar. A green check mark replaces the question mark (Figure 2-6). If you move to a different text string without clicking Done and Next, a yellow question mark is displayed for the string you translated. When you translate all elements for a class, a green check mark replaces the question mark next to the class name.

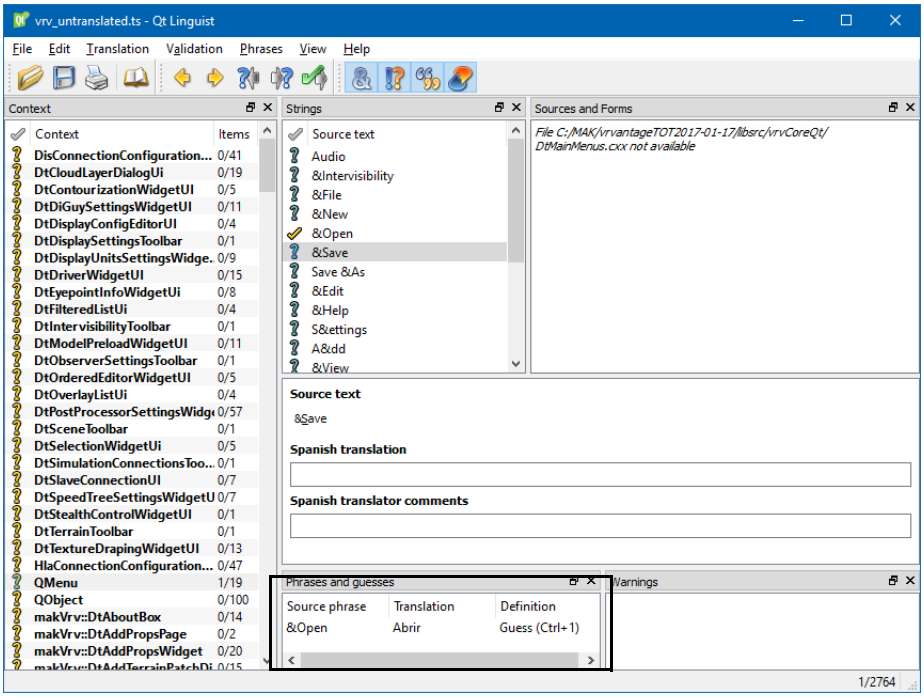


Figure 2-6. Completed translation for Open

10. Repeat this process for each element of source text for each class.
11. Choose **File** → **Release**. The file is saved in the format *filename.qm*.
12. Rename the file to a name that indicates the language to which you have translated the file, using the two-character abbreviation typically used with your language. For example, a Spanish translation would be renamed *product_es.qm*.
13. Save your translated *.ts* file.

2.5.1. Translating Other Interface Files

The *vr-vantage_untranslated.ts* file contains all the text used in the VR-Vantage interface. However, there may be other text and messages that are not specific to VR-Vantage that are generated by the software that implements the VR-Vantage GUI. To translate this text, you need to translate the file *qt_untranslated.ts*. Follow the same procedure as you did for *vr-vantage_untranslated.ts*.

2.5.2. Applying the Language Files

To put the language file into effect, use one of the following methods:

- Set your computer's environment to the proper language (LANG environment variable).
- Set the locale to the proper language.
- Start VR-Vantage with the `-G locale` command line option, where `locale` is the two-letter country code for the local language.



3. Running VR-Vantage Applications

This chapter explains how to start and stop VR-Vantage, how to load a terrain, and how to connect to a simulation network.

Starting VR-Vantage Applications 3-2
- Starting VR-Vantage on Windows 3-2
- Starting VR-Vantage on Linux or the Windows Command Line 3-4
- Command-Line Options 3-5
- Shutting Down VR-Vantage 3-9

Introduction to the VR-Vantage Graphical User Interface 3-9

Opening a Terrain 3-10
- Closing a Terrain 3-10

Connecting to a Simulation Network 3-11
- Setting Up Auto Connect 3-13
- Disconnecting from the Simulation Network 3-14
- Using the Simulation Connections Toolbar 3-14

Managing VR-Vantage Settings 3-15
- Synchronizing Settings Among VR-Vantage Installations 3-17
- Global Settings and Observer-Specific Settings 3-17

3.1. Starting VR-Vantage Applications

The typical sequence for using a VR-Vantage application is to:

1. Start the application.
2. Load a terrain database.
3. Connect to a simulation network.

- If you want to connect to an HLA simulation, you must have an RTI installed. For details, please see .
- Before you can start VR-Vantage, the license manager must be running.

3.1.1. Starting VR-Vantage on Windows

To start VR-Vantage on Windows:

1. If you are using Windows 7, on the Windows Start menu, choose **MAK Technologies** → **VR-Vantage 2.3** → *application*, where *application* is one of the VR-Vantage applications, such as VR-Vantage Stealth or VR-Vantage IG.

   If you are using Windows 10, on the Start menu, choose **MAK Technologies** → *application*.

Windows 10 has a flatter Start menu structure than previous versions of Windows. It does not support nested folders. In the interest of simplicity, for the rest of this manual, instructions for starting applications will use the full folder path of Windows 7. Windows 10 users should assume that all applications, tools, and documents are under **MAK Technologies**. If you are interested in using a nested menu for starting application on Windows 10, please read this blog: http://www.mak.com/connect/blog/entry/tech-tip-from-fred-starting-mak-applications-on-windows-10

The VR-Vantage window opens (Figure 3-1). The Load Terrain dialog box opens on top of the window.

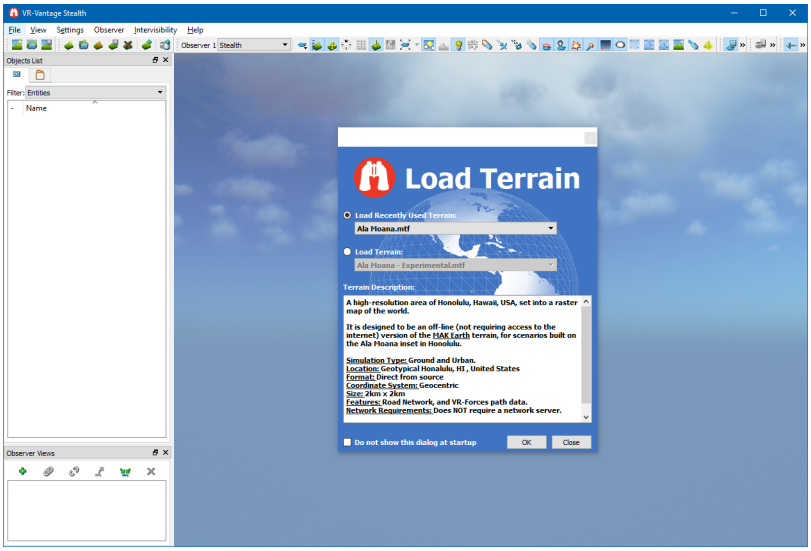


Figure 3-1. VR-Vantage window at startup

! When you start VR-Vantage, if you receive the following message, it means that you did not install the data package:

```
Data package not found in specified data directory: 'c:\MAK\vrvan-
  tage2.3\data'.
Please make sure the data package is installed.
```

VR-Vantage has an application installer and a data installer. You must install both of them.

---

2. Select one of the terrain loading options:
   - Load a recently used terrain.
   - Select a local terrain database from the list.

   If you do not want to load a terrain, click Close, the dialog box closes and no terrain loads. Clicking Close does not close VR-Vantage.
3. Click OK. VR-Vantage loads the selected terrain (Figure 3-2).)

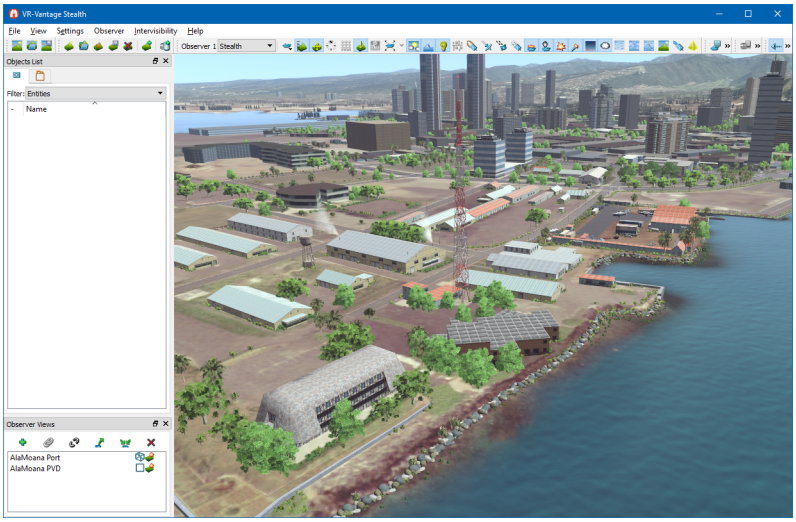


Figure 3-2. VR-Vantage window with a terrain loaded

3.1.2. Starting VR-Vantage on Linux or the Windows Command Line

To start VR-Vantage on Linux or the Windows command line:

1. Change directory to *./bin64*.
2. Enter:

   application [*options*]

   where *application* is `vrvStealth`, `vrvIG`, or `vrvPVD`.

   VR-Vantage starts as described in the previous section.

"Command-Line Options," on page 3-5 describes the VR-Vantage command-line options.

3.1.3. Command-Line Options

Table 3-1 lists VR-Vantage's command-line options.

Table 3-1: VR-Vantage command-line options (Sheet 1 of 5)

Option	Description
`(-- | --ignore_rest)`	Ignore all command-line options after this one. This option is useful for disabling part of a command line in a script for test purposes so that you do not have to retype all of the command options at a later point.
`--alphaBits` *bits*	Sets the number of alpha bits (0, 8).
`--antiAliasing` *level*	Set the anti-aliasing level (0, 2, 4, 8, or 16). Default: 4. Depending on your graphics card and driver, some combinations of anti-aliasing and bit settings (`--depthBits`, `--stencilBits`, and `--alphaBits`) may be incompatible. For example, if you set stencil bits to 8, the depth must be set to 24 and anti-aliasing may need to be set to 0 or 2.
`--appDataDir` *directory*	Specifies the location of application data.
`--autoReconnectPeriod` *period*	Specifies the amount of time, in seconds, to wait before trying to attach to disconnected remote displays.
`(-B | --bindObservers)` *"observer1,..., observern,saved_views_file"*	If you load a scene that has multiple observers, this command assigns the views in the specified saved views file to the specified observers. The first saved view is assigned to the first observer, the second to the second observer, and so on. If the observers are not specified as arguments, only the first observer in the display engine configuration gets a view assigned to it. The entire string must be quoted. There are no spaces after commas. You would ordinarily also specify a display configuration that contains the observers with `--dispSetting`. You must specify a terrain database for this command to work. For more information about using this command-line option, please see "Attaching the Observer at Startup," on page 16-15.
`(-c | --showConsole)`	Display a console window.
`(-C | --autoConnect)` *{filename | connector_name}*	Automatically connect to the network using the specified connector.
`--dataDir` *directory*	Specifies the location of the data directory (terrains and so on).
`--depthBits` *bits*	Sets the depth of graphics to 24 bits or 32 bits. Default: 24.

Table 3-1: VR-Vantage command-line options (Sheet 2 of 5)

Option	Description
`--disableDynamicTerrain`	Disables the dynamic terrain engine. May slightly speed up terrain loading and paging.
`--discoveredModelPreload-FileName` *filename*	Specifies a file to which the list of models discovered during a session that were not preloaded is written.
`--displayEnginePort` *port*	Specifies the display engine port used by a remote display engine. Used only if `--slave` is specified.
`--dispSetting` *filename*	Specifies the display engine configuration to load.
`(-e | --extraArgs)` *string*	Allows you to send arbitrary strings to the initializer, which can be queried for by plug-ins. This option works around the restriction on adding new command-line arguments.
`--enableOculus`	Enable support for the Oculus Rift HMD. Implies `--noVSync`.
`--enableVSync`	Turns on VSync.
`--entDispSetting` *filename*	Specifies the entity display setting to load.
`--entityDefsDir` *directory*	Loads the *.hier* and *.leaf* files in the specified directory.
`--envSetting` *filename*	Specifies the environment configuration file to load.
`(-E | --entTypeMap)` *filename*	Load the specified object type mapping file. The file extension determines the type of mapping.
`--factoryRootDir`	Set the root directory used to restore settings to factory defaults.
`--forceTextShaping`	Force all text to use HarfBuzz shaping. Use this option if you need support for right-to-left languages. It may affect performance.
`--fullScreen`	Start with the window in full screen mode.
`(-F | --frameRate)` *rate*	If rate is 0, the frame rate is variable. Otherwise, specifies a fixed frame rate for rendering.
`(-G | --locale)` *language*	Specifies the locale (for localization).
`(-h | --help)`	Display command-line usage information.
`--highestPriority`	Changes the priority of the application and the main render thread to the highest available priority. This command may starve other processes.
`(-I | --unbatchedRendering)`	Disables instance rendering. Default: Enabled.
`(-K | --disableKDTrees)`	Disable creation of KD trees for intersections.
`(-l | --logFileName)` *filename*	Specifies the log file name.

Table 3-1: VR-Vantage command-line options (Sheet 3 of 5)

Option	Description
(-L | --loadObservers) *filename*	Loads the saved observers in the specified file. You can use this command multiple times, for example: `--loadObservers file1 --loadObservers file2`
--mainScreenNum *num*	Specifies the display that the main window should use. The screen number can be a number from 0 to the number of screens -1. This number matches the screen number used in the Display Engine Configuration Editor for a window.
--mergeEntityTypeMap {*directory* | *filename*}	Merges all entity type mapping files (**.metx*) in a directory or individual entity type mapping files. Can be used multiple times.
--mergeHierarchyEntityDef *filename*	Merges the specified *.hier* file into the default hierarchy (usually loaded from *./appData/definitions*.) Merging is additive only. No nodes in a previously loaded hierarchy are replaced.
--mergeHierarchyTactical-GraphicsDef *filename*	Merges the specified *.hier* file into the default hierarchy (usually loaded from *./appData/definitions*.) Merging is additive only. No nodes in a previously loaded hierarchy are replaced.
--mergeHierarchyUnitDef *filename*	Merges the specified *.hier* file into the default hierarchy (usually loaded from *./appData/definitions*.) Merging is additive only. No nodes in a previously loaded hierarchy are replaced.
--mergeLeafEntityDef *filename*	Merges the specified leaf file into the default entity element definitions.
--mergeLeafTacticalGraph-icsDef *filename*	Merges the specified leaf file into the default tactical graphics element definitions.
--mergeLeafUnitDef *filename*	Merges the specified leaf file into the default unit element definitions.
--mergeModelDefs {*directory* | *filename*}	Merges all model definition files (**.ommx*) in the specified directory or individual model definition files. Can be used multiple times.
--mergeTacticalGraphics-TypeMap {*directory* | *filename*}	Merges all tactical graphics type mapping files (**.mtgx*) in a directory or individual tactical graphics type mapping files. Can be used multiple times.
--mergeUnitTypeMap {*directory* | *filename*}	Merges all unit type mapping files (**.magx*) in a directory or individual unit type mapping files. Can be used multiple times.
--modelPreloadFileName *filename*	Specifies a file that has a list of models to preload.

Table 3-1: VR-Vantage command-line options (Sheet 4 of 5)

Option	Description
(-n | --notifyLevel) *notify_level*	Specifies the notification level for application messages, where *notify_level* is a number from 0 through 4. The least verbose message level is 0, the highest is 4. The default is 2.
--noAppDataEntityDefs	Do not load the entity element definitions in *./appData/definitions*.
--noAppDataTacticalGraphicsDefs	Do not load the tactical graphics element definitions in *./appData/definitions*.
--noAppDataUnitDefs	Do not load the unit element definitions in *./appData/definitions*.
--noAppDataEntityTypeMap	Do not load entity type mappings in *./appData/definitions*.
--noAppDataModelDefs	Do not load model definitions in *./appData/definitions*.
--noAppDataTacticalGraphicsTypeMap	Do not load tactical graphics type mappings in *./appData/definitions*.
--noAppDataUnitTypeMap	Do not load unit type mappings in *./appData/definitions*.
--noGui	Start without the graphical user interface.
(-O | --hasNTPSync)	Enables synchronized ocean between different VR-Vantage applications, such as VR-Vantage Stealth and the VR-Forces front-end. System clocks must be synchronized.
--plugin *filename*	Load the specified plug-in. Can be used multiple times.
--preLoadClouds	Causes VR-Vantage to calculate the location and settings of all clouds in the environment before the terrain or scene is loaded. This increases load time, but eliminates stalls when cloud cover settings change during runtime.
--pseudoFullScreen	Use pseudo full screen for full screen mode. This may improve performance when using dynamic ocean.
(-s |--slave)	Run as a remote display engine.
(-S | --suppressHatIntersectOnAttach)	Suppress the periodic HAT intersection when attached. This setting can improve performance if you need to attach the observer to ground vehicles a lot, but do not need to attach to air vehicles.
--stencilBits *bits*	Sets the number of stencil bits (0 or 8). Default: 0.
--tcpSendTimeout	Sets the timeout (in seconds) for the master display engine to wait for a remote display engine before giving up on a send. Default: 30 seconds.

Table 3-1: VR-Vantage command-line options (Sheet 5 of 5)

Option	Description
`(terrain_MTF_file | scene_MSF_file)`	Specifies a terrain (MTF) or scene (MSF) file to load. The path must be a relative path (from *./bin64*) to the file, for example, *../userData/terrains/MTF_file*.
`--unplug filename`	Do not load the specified plug-in. Can be used multiple times.
`--userDataDir directory`	Specifies the user data directory.
`(-v | --version)`	Display version information and exit.

Windows syntax:
`-W | --withDisplayEngine port@netAddress;displayConfigFilename1 .... n`
Linux syntax:
`-W | --withDisplayEngine port@netAddress:displayConfigFilename1 .... n`
	Connect to a display engine running on another computer. For more information, please see *Loading Multiple Display Engine Configurations from the Command Line* under "Loading a Display Engine Configuration," on page 6-9.

`(-z | --useReverseZBuffer)`	Specifies use of reverse Z buffering, which may reduce Z-fighting when viewing the terrain from a distance. (3D observer modes only.)
`(-Z | --ignoreZoomFor-SpeedtreeLod`	Disable zoom for SpeedTree LOD.

3.1.4. Shutting Down VR-Vantage

➤ To shut down VR-Vantage, choose **File** → **Exit**.

3.2. Introduction to the VR-Vantage Graphical User Interface

The VR-Vantage graphical user interface (GUI) uses features that are common to most windowed applications, such as menus, toolbars, dockable panels, and context-sensitive help. For a brief introduction to these features and descriptions of common icons used in VR-Vantage, please see Chapter 45, *The VR-Vantage User Interface*.

3.3. Opening a Terrain

VR-Vantage includes several terrains. When you open a terrain in VR-Vantage, you open an MTF file; you do not open a file in its native format (such as Open-Flight or DTED). This procedure explains how to open the MAK-supplied terrains or a terrain that you have created and saved in the MTF format.

(If you want to open files directly from their native formats, you do so by loading terrain patches, raster maps, and feature layers. You can then save the resulting terrain to an MTF file. For details, please see "Terrain Agility and Composability," on page 25-2.) For details about how to compose a terrain from source data, please see Chapter 29, *Terrain Performance and Configuration*. For details about loading streaming terrains using an earth file, please see Chapter 26, *Streaming Data Using Earth Files*. For details about opening a geocentric terrain from a terrain server, please see "Connecting to Terrain Servers," on page 26-2.



- When VR-Vantage starts up, it prompts you to load a terrain (unless you have disabled the feature). Thereafter, you would use the procedure described here to load terrains.
- The first time you load a terrain (including the terrains supplied with VR-Vantage), VR-Vantage caches textures in its fast load format. This is a slow process. However, once it is cached, the terrain loads quickly. If you want to avoid runtime caching, you can preprocess the terrain with the medfTool before you load it. If you want to disable runtime caching, you can do so as described in "Configuring File Caching," on page 29-2.

To open a terrain:

1. Choose **File** → **Open Terrain**, or click the Open Terrain button () on the Terrain Toolbar. The Open Terrain dialog box opens.
2. Select the terrain (MTF format) that you want to open.
3. Click Open.

3.3.1. Closing a Terrain

To close a terrain, use one of the following methods:

- Choose **File** → **Close Terrain**, or click the Close Terrain button () on the Terrain Toolbar.
- Choose **File** → **New Terrain**, or click the New Terrain button () on the Terrain Toolbar.
- Choose **File** → **New Scene**, or click the New Scene button () on the Scene Toolbar.
- Open a different terrain.

3.3.2. Reloading a Terrain

You can quickly reload a terrain without using a file selection dialog box.

➤ To reload a terrain, choose **File** → **Reload Terrain**, or click the Reload Terrain button () on the Terrain Toolbar.

3.4. Connecting to a Simulation Network

To view a simulation, you must connect VR-Vantage to a simulation network. VR-Vantage includes connectors for DIS, CIGI, HLA 1.3, and HLA Evolved (IEEE 1516-2010, also referred to as HLA 1516e). You can change the configuration values of the default connectors and you can add connectors. For details about creating and editing connections, please see Chapter 4, *Managing Network Connections*.

When VR-Vantage starts up, it does not automatically connect to a simulation network unless you configure it to do so. The auto connect feature lets you specify that VR-Vantage automatically connect to a network through a connection when it starts up. If you do not use auto connect, you must connect manually. A connector's status is listed in the Status column in the Connection Status page of the Simulation Connections dialog box.



- You can connect to more than one simulation network at the same time.
- You can quickly get connection status, connect to a network, and disconnect on the Simulation Connections Toolbar.

To connect to a simulation network:

1. Choose **Settings** → **Connection**. The Simulation Connections dialog box opens (Figure 3-3).

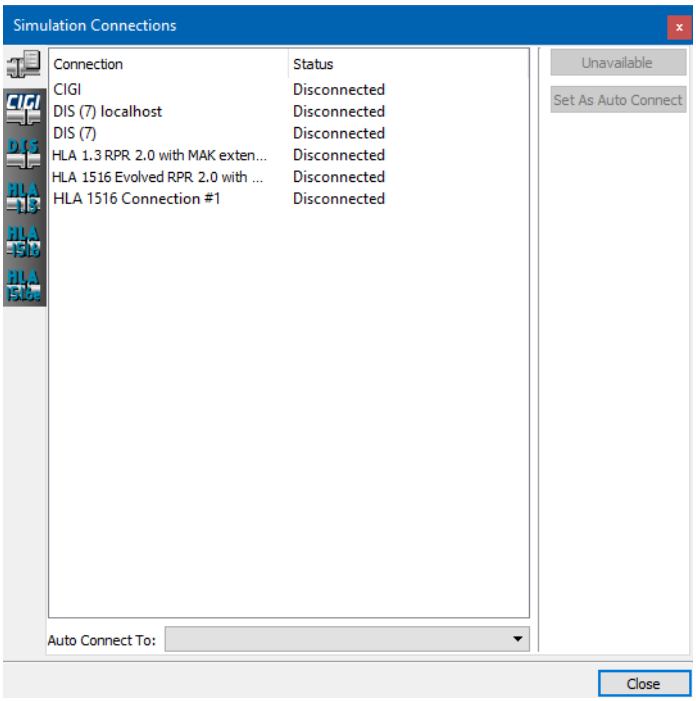


Figure 3-3. Connection Status page

2. Select the page for the connection type that you want to use.
3. Select a connector in the Connections list (Figure 3-4).

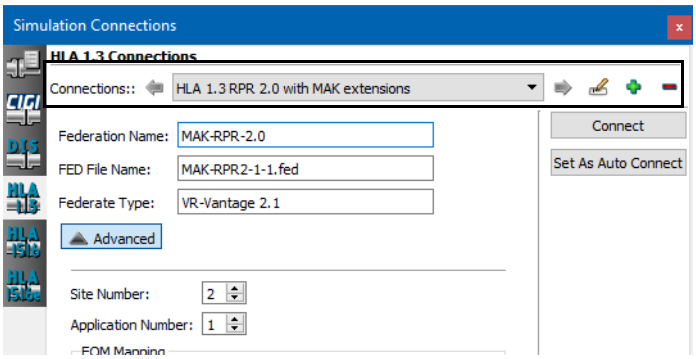


Figure 3-4. HLA 1.3 Connections page

4. Click Connect.

3.4.1. Setting Up Auto Connect

If you specify auto connection, VR-Vantage connects to the specified simulation network when it starts. If a connection is designated as an auto connection, a star is added to its entry in the connections list (Figure 3-5).

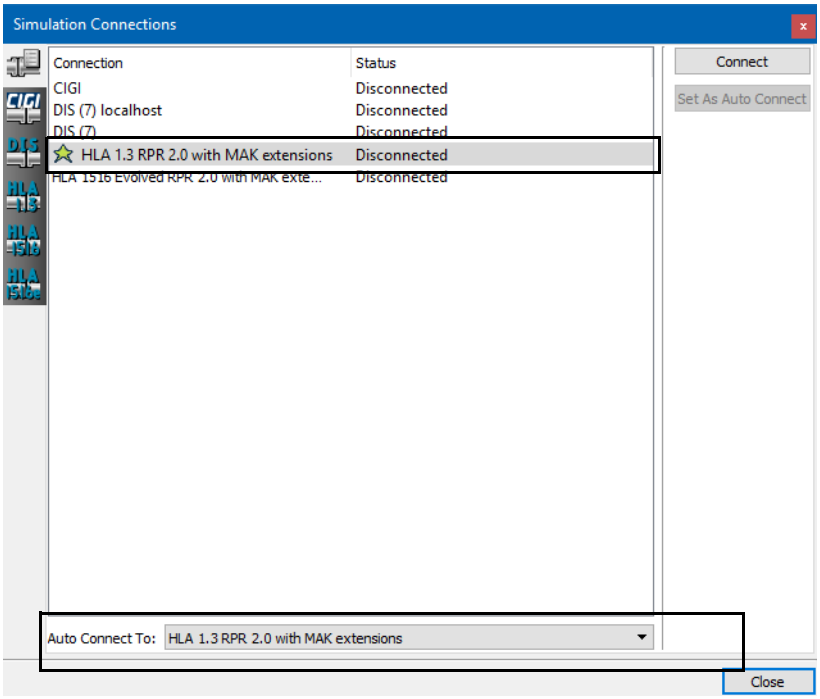


Figure 3-5. Auto connect enabled

To configure auto connection:

1. Choose **Settings** → **Connection**. The Simulation Connections dialog box opens.
2. To configure auto connect on the Connection Status page (Figure 3-5):
   a. Select the Connection Status page.
   b. Do one of the following:
      - Select the desired connection in the Connection list and click Set As Auto Connect.
      - Select a connection from the Auto Connect To list. (To disable auto connection, select the blank line.)

   To configure auto connect on a protocol-specific page:

   a. Select the page for the connection type that you want to use.
   b. Select a connector in the Connections list.
   c. Click Set As Auto Connect.

3.4.2. Disconnecting from the Simulation Network

To disconnect from the simulation network:

1. Choose **Settings** → **Connection**. The Simulation Connections dialog box opens (Figure 3-3).
2. On the Connection Status page or on a protocol-specific page, select the connection that you want to disconnect.
3. Click Disconnect.

3.4.3. Using the Simulation Connections Toolbar

The Simulation Connections Toolbar lets you quickly open the Simulation Connections dialog box to the Connection Status page. It also lets you quickly connect to and disconnect from a specific connection.



Figure 3-6. Simulation Connections Toolbar

➤ To view connection status, click the Connection Status button ( ).

To quickly connect to selected simulation connections:

1. Click the down arrow next to the Connect button ( ). A list of all configured simulation connections is displayed.

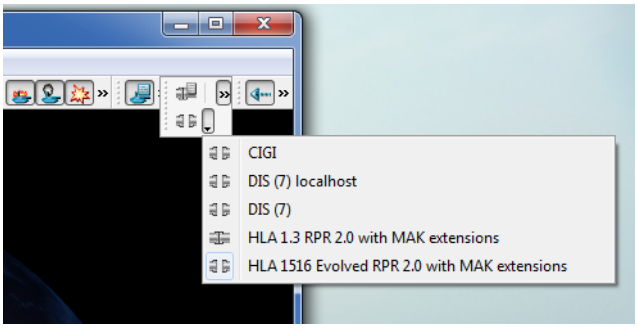


Figure 3-7. Connection list

2. Click the connection that you want to connect to when you use the toolbar.

- You can only select one connection in the list. To connect to multiple simulations, use the Connection Settings dialog box.
- The Connect button toggles between an open and closed state depending on the status of the connection.

---

3. Click the Connect button.

➤ To disconnect all connections, click the Disconnect button ( ).

3.5. Managing VR-Vantage Settings

VR-Vantage has the following types of settings:

- Independent, or display, settings are the individual settings managed on the Settings menu and the pages of the Display Settings dialog box. Changing an independent setting, such as enabling track histories or wireframe mode, does not affect other settings.

  Independent settings are categorized as being global settings or observer-specific settings. Global settings apply regardless of the observer mode. Observer-specific settings can be different for each observer mode. For more information, please see "Global Settings and Observer-Specific Settings," on page 3-17.

- Dependent settings are the schemas, model definitions, element definitions, and object type mappings that collectively determine how simulation objects and objects are modeled in the scene. Changing a dependent setting, such as a model definition may affect settings that depend on it, such as an entity definition.

VR-Vantage manages settings as follows:

- As you make changes to settings during a session, it saves the changes, except for model definitions, element definitions, and simulation object type mappings.
- It remembers the settings in effect when you exit and uses them as the default settings in your next session.
- During any particular session, you can revert to the default settings.
- You can save the current settings to a file.
- You can import saved settings.
- You can restore settings to those that were in effect when VR-Vantage was first installed (factory settings).

Dependent settings have two additional options that independent settings do not have:

- You can save or restore some types of settings by specific setting.
- You can merge settings from a file into the current settings.

The procedures for saving and restoring settings are the same regardless of the types of settings. You start the process by clicking a button on a settings dialog box page. The way that VR-Vantage saves and restores depends on the type of setting, as follows:

- When you initiate a save or restore operation on independent settings, it saves or restores all of the settings on the dialog box page from which you started the save or restore. If you have changed a setting on another page, it is not affected.
- When you initiate a restore operation on dependent settings, related settings may also be restored. VR-Vantage displays a message listing the settings that will be restored.
- When you save dependent settings, you either export all settings of a given type or selected settings. For details, please see relevant chapters in Section VIII, "Modeling Simulation Objects,".

(Figure 3-8) illustrates the common settings management buttons. (The Visual Model Editors dialog box has some additional buttons.)



Figure 3-8. Settings buttons

- Revert to the default settings. Override any changes made to settings during this session.
- Restore the factory settings. Change settings to the settings used when VR-Vantage was installed. (The factory settings and default settings may be different.)
- Import settings from a file. Replace the current settings with those stored in the selected file.
- Merge in settings from a file. Merging works as follows:
  - Saved settings that do not have a match in the current settings are added.
  - Saved settings replace matching settings.
  - Current settings that are not replaced (that is, for which there is no match in the saved settings) remain valid.

  Figure 3-9 illustrates how merged settings work. Setting A in the saved settings has no match in current settings, so it is added to the merged settings. Current settings B_2 and C_2 are replaced with saved settings B_1 and C_1. Current setting D has no match in saved settings, so it is retained in the merged settings.

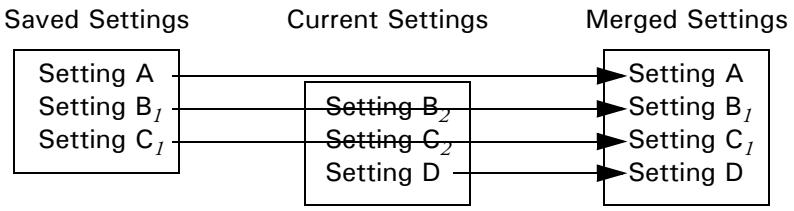


Figure 3-9. Merged settings

- Export all settings to a file. Save the current settings to the selected file.
- Save the selected setting to a file.

- The buttons described in this section are disabled on dialog box pages on which their function is not supported. This includes most of the pages on the Terrain Settings dialog box.
- Each VR-Vantage application saves settings to its own directory (*./appData/settings/stealth*, *./appData/settings/vantageIG*, and so on, and to *./appData/definitions*.)
- If you load an alternative settings file at startup, it is treated like an imported settings file.

3.5.1. Synchronizing Settings Among VR-Vantage Installations

If your simulation is using several VR-Vantage applications or display engines, you may want to use the same settings across all of them, particularly if you have added model definitions and mappings to the factory settings. To synchronize settings, copy the data directories (*./appData*, *./data*, and *./userData*) from the computer on which you have updated them to the computers on which the other VR-Vantage applications and display engines are installed. Then import the settings into any Settings page that you customized on the source VR-Vantage machine.

3.5.2. Global Settings and Observer-Specific Settings

VR-Vantage has two types of independent settings – global settings and observer-specific settings.

Global settings affect the display regardless of which observer mode you are using. Global settings include simulation object labels, tactical graphics, and cockpit displays. The settings on the Settings menu are all global settings.

Observer-specific settings can be different for each observer mode. In fact, given a similar projection and model set, it is the different settings that distinguish one observer mode from another. For example, one difference between Stealth observer mode and Out-the-Window observer mode is that Stealth mode has tactical smoke enabled and Out-the-Window mode has it disabled.



4. Managing Network Connections

This chapter provides details about how to configure network connections.

Managing Simulation Connections 4-2
- Displaying Connections 4-2
- Adding a Simulation Connection 4-3
- Renaming a Simulation Connection 4-4
- Deleting a Simulation Connection 4-4

Configuring Connection Settings 4-4
- HLA Connection Settings 4-5
- DIS Connection Settings 4-6
- CIGI Connection Settings 4-8

4.1. Managing Simulation Connections

VR-Vantage uses connectors to access the various types of input it requires. To connect to a simulation you connect an appropriate simulation connection, for example, an HLA 1.3 connection or a DIS connection. The Simulation Connections dialog box lists connections and lets you manage them. You can:

- Add connections.
- Configure connections.
- Delete connections.
- Connect to the network using a specific connection.
- Disconnect a connection.
- Specify a connection to use automatically at startup.
- Enable or disable publishing of observers.
- Enable or disable acceptance of view control messages.

4.1.1. Displaying Connections

The Connection Status page on the Simulation Connections dialog box displays all of the configured connections and their connection status.

To view the list of connections:

1. Choose **Settings** → **Connection**. The Simulation Connections dialog box opens.
2. Select the Connection Status page (Figure 4-1).

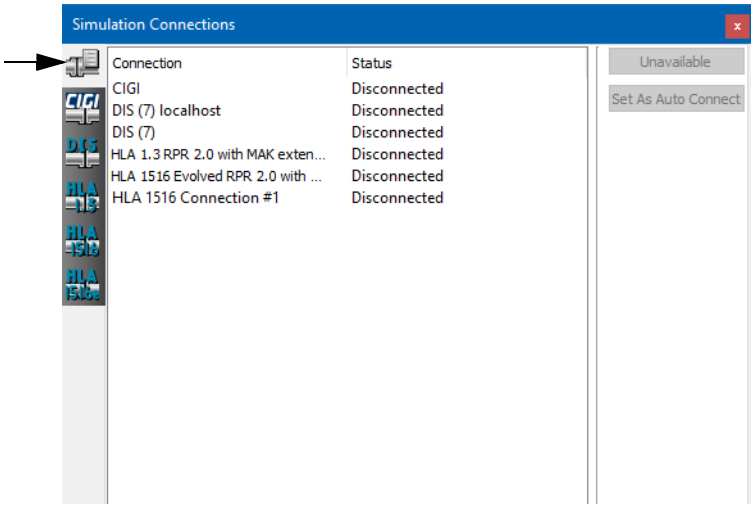


Figure 4-1. Simulation Connections dialog box, Connection Status page

4.1.2. Adding a Simulation Connection

You might want to edit the parameters for an existing connection and add a new connection with the updated parameters. In HLA, you might want to add a connection that supports a different FOM or FOM Mapper.

To add a simulation connection:

1. Choose **Settings** → **Connection**. The Simulation Connections dialog box opens (Figure 4-1).
2. Select the page for the connection type you want to add (Figure 4-2).

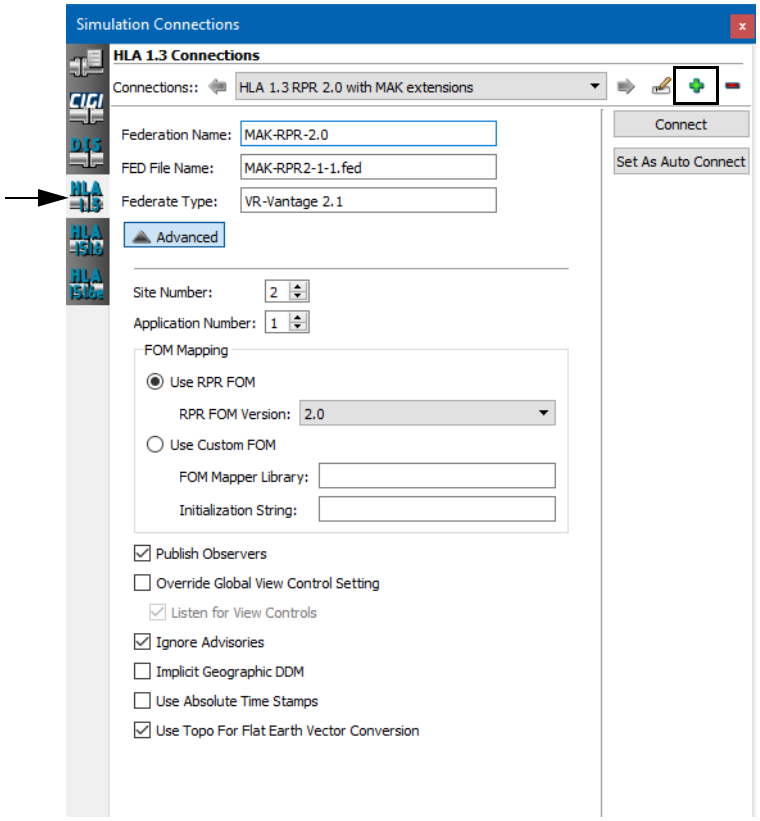


Figure 4-2. Protocol-specific page

3. Click the Add button (+) to the right of the Connections list. A new connection is added to the list. (To change the name of the connection, follow the procedure in "Renaming a Simulation Connection," on page 4-4.)
4. Set the connection parameters to the desired values. (For details about connection parameters, please see "Configuring Connection Settings," on page 4-4.)
5. Optionally, click Set As Auto Connect to have VR-Vantage automatically connect using this connection when it starts up.
6. Click Close.

4.1.3. Renaming a Simulation Connection

To rename a simulation connection:

1. Choose **Settings** → **Connection**. The Simulation Connections dialog box opens (Figure 4-1).
2. Select the page for the connection type that you want to rename.
3. In the Connections list, select the connection that you want to rename.
4. Click the Rename button (✎). The Rename Connection dialog box opens.
5. Type a new name for the connection.
6. Click OK.

4.1.4. Deleting a Simulation Connection

To delete a simulation connection:

1. Choose **Settings** → **Connection**. The Simulation Connections dialog box opens (Figure 4-1).
2. Select the page for the connection type that you want to delete.
3. In the Connections list, select the connection that you want to delete.
4. Click the Delete button (−). You are prompted to confirm the deletion.
5. Click Yes.

4.2. Configuring Connection Settings

The different simulation connection types have different parameters, but the process of configuring a connection setting is essentially the same regardless of the connection type.

HLA 1.3 and HLA 1516 have the same set of connection parameters. HLA Evolved has additional parameters. HLA 1.3 and HLA Evolved have default connections configured for RPR FOM 2.0 with extensions that support changing DI-Guy characters based on messages sent from VR-Forces, MAK's computer generated forces application.

To configure connection settings:

1. Choose **Settings** → **Connection**. The Simulation Connections dialog box opens (Figure 4-1).
2. Select the page for the connection type that you want to configure.
3. In the Connections list, select the connection that you want to configure, or add a new connection, as described in "Adding a Simulation Connection," on page 4-3.
4. Click Advanced to display all of the parameters.
5. Specify values for connection parameters, as described in Table 4-1, Table 4-2, or Table 4-3.

4.2.1. HLA Connection Settings

Table 4-1 describes the connection settings for HLA 1.3, HLA 1516, and HLA 1516e.

Table 4-1: HLA connection settings

Parameter	Description
Federation Name	Specifies the federation execution name.
FED File Name	Specifies the FED file. Default: the federation name with *.fed* appended.
Federate Type	Federate Type can be used to distinguish different categories of federates. Its only practical use is in save-and-restore.
Advanced	
Federate Name	Specifies a unique name that can be assigned to the federate. No two federates within the same federation can have the same name. If no name is specified, the RTI will automatically assign a unique name for the federate. (HLA 1516e only)
Local Settings Designator	Specifies a string that is passed to the RTI during federate connect when using HLA Evolved. It is usually used to provide additional RTI configuration, but every RTI uses it differently. Please consult your RTI documentation for proper usage. (HLA 1516e only)
Site Number	Specifies the site number for the observer. The Site ID and Application Number uniquely identify the observer.
Application Number	Specifies the application number for the observer.
Use RPR FOM	Specifies use of a version of the RPR FOM.
RPR FOM Version	Specifies the RPR FOM version. VR-Vantage has built-in support for RPR FOM versions 0.5, 0.7, 0.8, 1.0, 2.0006, 2.0014, 2.0017, and 2.0020 (RPR FOM 2.0). You must specify the correct FED file for the version you use. If you use a different FOM, you must specify the correct FED file and FOM Mapper.
Use Custom FOM	Specifies use of a FOM other than the RPR FOM.
FOM Mapper Library	Specifies the name of a FOM Mapper library.
Initialize String	Specifies initialization data for the specifiedFOM Mapper, if required. It is your responsibility to know if your FOM Mapper needs initialization data.
FOM Modules	Lets you specify FOM Modules. Consult your RTI documentation for information about FOM Modules. (HLA 1516e only)

Table 4-1: HLA connection settings

Parameter	Description
Publish Observers	Specifies whether or not to publish an entity for the observer that is using this connection.
Override Global View Control Setting	Apply the setting in this connection instead of the setting in the GUI. If this is used, the GUI setting is ignored.
Listen for View Controls	Specifies whether or not to respond to view control messages sent by other applications.
Ignore Advisories	Enables or disables use of HLA advisory messages.
Implicit Geographic DDM	Enables or disables use of implicit DDM. Valid only with the MAK RTI.
Use Absolute Time Stamps	Enables or disables use of absolute time stamps.
Use Topo For Flat Earth Vector Conversion	Use a topocentric system to convert vectors from geocentric to flat earth. This approach is deprecated. Select this option when receiving data from simulations that use this approach, such as older versions of VR-Forces and DI-Guy.

4.2.2. DIS Connection Settings

Table 4-2 describes the connection settings for DIS connections.

Table 4-2: DIS connection settings

Parameter	Description
Port	The DIS port to connect to.
Exercise ID	Specifies the exercise ID for VR-Vantage.
Site Number	Specifies the site number for VR-Vantage.
Application Number	VR-Vantage can send PDUs to the network to communicate the observer's position and orientation. By default, these PDUs are sent to the host machine's broadcast address. You may want to direct VR-Vantage data to a unicast address so it is received by only one application.
DIS Version	Specifies the DIS version (4, 5, 6, or 7).
Network Interface Address	On computers with multiple network cards, the address of the network card to use.
Advanced	
Destination Address	Specifies the destination address for outgoing PDUs. Default: the computer's broadcast address.

Table 4-2: DIS connection settings

Parameter	Description
Multicast Addresses	Specifies one or more multicast groups to subscribe to. Separate multiple addresses with a semi-colon (;). For more information about multi-cast addresses, please see the note that follows this table.
Send Buffer Size	Increasing the send and receive buffer size can reduce dropped packets. If the parameters are set to 0, the system default setting is used.
Receive Buffer Size	Increasing the send and receive buffer size can reduce dropped packets. If the parameters are set to 0, the system default setting is used.
Multicast TTL	Specifies the number of routers that a message can pass through when PDUs are sent to multi-cast addresses.
Enable Timeout Processing	Enables or disables applying a timeout value to discovered objects.
Timeout Interval	The number of seconds to wait without receiving an updated before timing out an object.
Use Asynchronous IO	Enables or disables asynchronous IO. Use of asynchronous IO can reduce dropped packets.
Use Absolute Time Stamps	Enables or disables use of absolute time stamps.
Show Self Published Objects	Broadcast the location of objects that VR-Vantage creates - primarily the observer.
Publish Observers	Specifies whether or not to publish an entity for the observer that is using this connection.
Override Global View Control Setting	Apply the setting in this connection instead of the setting in the GUI. If this is used, the GUI setting is ignored.
Listen for View Controls	Specifies whether or not to respond to view control messages sent by other applications.
Use Topo For Flat Earth Vector Conversion	Use a topocentric system to convert vectors from geocentric to flat earth. This approach is deprecated. Select this option when receiving data from simulations that use this approach, such as older versions of VR-Forces and DI-Guy.



Multicast subscription is a property of the host machine, not of individual applications. If several applications are running on the same machine using the same port, and some of those applications subscribe to multicast groups, all the applications will receive packets for all multicast groups to which any application is subscribed.

Therefore, if VR-Vantage is running on the same machine and using the same port as other applications that subscribe to multicast groups, VR-Vantage may receive traffic intended for the other applications.

4.2.3. CIGI Connection Settings

Table 4-3 describes the connections settings for CIGI connections.

Table 4-3: CIGI connection settings

Parameter	Description
CIGI Version	The version of CIGI being used.
Driver Mode	Synchronous or asynchronous.
CIGI Entity Mapping	The host-specific CIGI mapping to use. These mappings are configured in the CIGI Mapping Settings dialog box.
Driver Frequency	The frequency at which VR-Vantage should try to communicate with the host. There is no guarantee that this frequency can be maintained.
Host Frequency	
Transport Type	The network transport type to use: Best Effort, Reliable Client, or Reliable Server.
Host IP Address	The IP address of the CIGI host.
IG Network Interface Address	The IP address for VR-Vantage.
Listen Port	The port on which to listen for CIGI messages.
Send Port	The port on which to send CIGI messages.
Packet Buffer Size	Specifies the amount of space to allocate for packets.
CIGI Messages: Incoming CIGI Messages: Outgoing	The number and size of CIGI message buffers to maintain. Decreasing the number or size can slow down performance. Increasing the number or size can use up memory.



5. Optimizing Performance

This chapter explains how to monitor performance and improve performance.

Factors that Affect Performance 5-2
Displaying Performance Statistics 5-3
Displaying the VR-Vantage Function Profiler 5-5
Displaying OSG Statistics 5-5
Preloading Models 5-6
Command-Line Options for Preloading Models 5-7
Clearing the Model Instancing Cache 5-7
Configuring VSync 5-8
Configuring Graphics Quality 5-8
Alpha to Coverage 5-9
Using Incremental Compiling with Streamed Data 5-11
Coloring Draw Calls 5-11
DI-Guy Performance Settings 5-12

5.1. Factors that Affect Performance

The performance of a visualization product such as VR-Vantage depends on:

- Processor speed.
- Video card. (Always use the latest drivers for your video card.)
- Available memory.
- Size of the terrain database.
- The number and complexity of models loaded and how well they have been optimized for performance. For information about optimizing models, please see "Best Practices for Creating Models for VR-Vantage," on page 35-27.
- Number of objects in the simulation and how active they are.
- Use of model instancing.

You can display statistics to help you evaluate VR-Vantage's performance. For details, please see "Displaying Performance Statistics," on page 5-3.

The following display features can affect performance:

- Displaying performance statistics.
- Use of ground clamping.
- Displaying track histories.
- Displaying height-above-terrain lines.
- Enabling advanced lighting and various lighting effects.
- Enabling dynamic ocean effects. When you are using Plan View observer mode, enabling wakes, spray effects, and the buoyancy model can seriously degrade performance. These features are disabled by default. If you inadvertently enable them and experience this problem, switching to Stealth mode and back to Plan View mode restores performance.
- Anti-aliasing settings. For details, please see "Configuring Graphics Quality," on page 5-8.

In addition to the sections in this chapter and the sections in Chapter 29, *Terrain Performance and Configuration*, the following sections describe options for optimizing performance:

- "Configuring SpeedTree Trees," on page 35-24.
- "Clearing the Model Instancing Cache," on page 5-7.
- "Setting the Clipping Planes," on page 6-14.
- "Considerations and Limitations for Building Terrains," on page 25-5.
- "Filtering Simulation Objects Using Interest Management," on page 19-16.
- "Setting the Ocean LOD Elevation," on page 25-18.
- "Configuring the Ocean Height Map," on page 25-18.
- "Displaying Wakes and Spray Effects," on page 21-23.
- "Configuring the Skybox Cube Map," on page 13-13.
- "Improving Performance when Attaching to Ground Objects," on page 16-6.
- "Enabling Buoyancy for Surface Entities," on page 21-24.
- Chapter 13, *Lighting Effects*.

5.2. Displaying Performance Statistics

VR-Vantage can display two kinds of performance statistics — OSG statistics (Figure 5-1), and VR-Vantage statistics (Figure 5-2). VR-Vantage also has a function profiler that developers can use to assess the performance of new code.

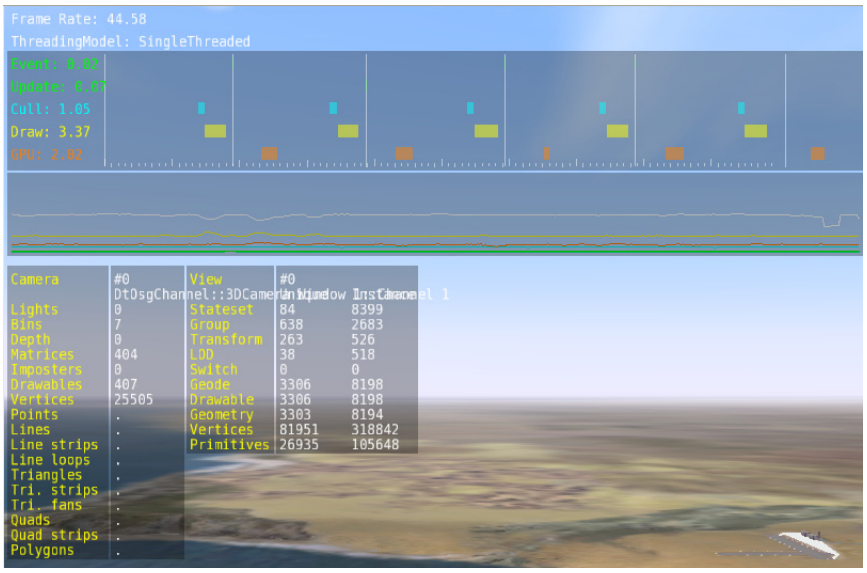


Figure 5-1. OSG performance statistics

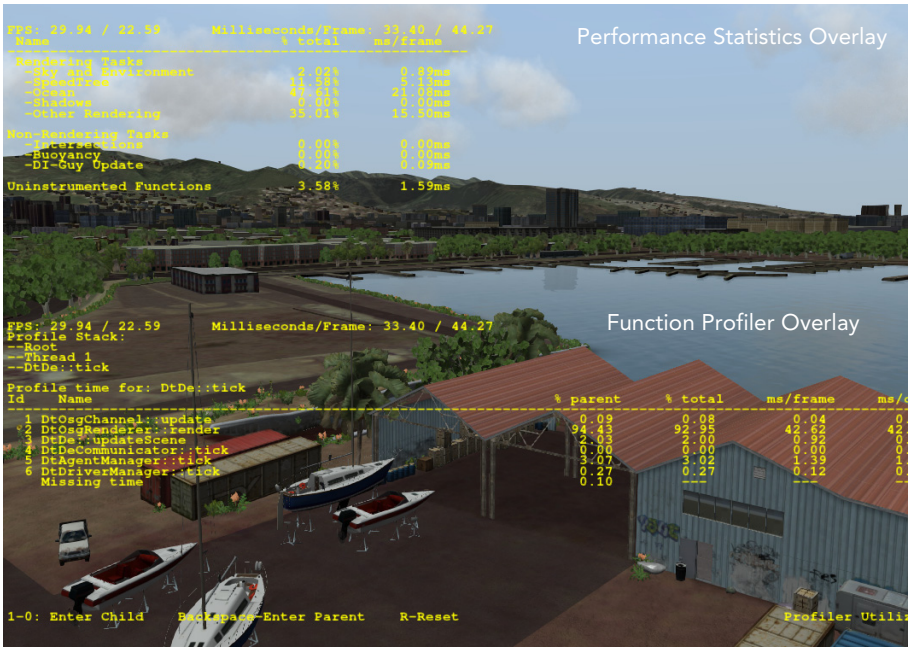


Figure 5-2. VR-Vantage performance statistics and function profiler

Displaying performance statistics may affect performance. Displaying OSG statistics may affect performance more than VR-Vantage statistics.

To display VR-Vantage performance statistics:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 5-3).

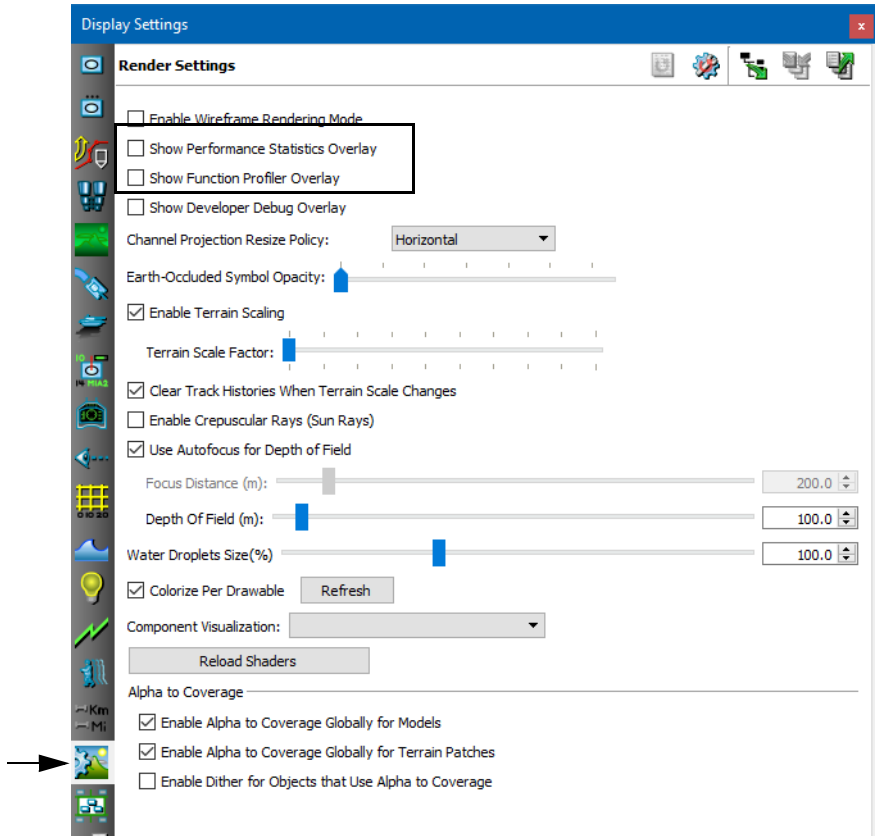


Figure 5-3. Render Settings page

3. Select the Show Performance Statistics Overlay check box. A statistics display is added to the window (Figure 5-2).

5.2.1. Displaying the VR-Vantage Function Profiler

The VR-Vantage Function Profiler is primarily for the use of developers who are debugging VR-Vantage applications. For information about how to integrate the profiler into your code, please see Chapter 11, Using the Function Profiler in VR-Vantage *Developers Guide*.



For information about another debugging utility, please see "Using the Developer Debug Overlay," on page 5-13.

To run the VR-Vantage Profiler:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 5-3).
3. Select the Show Function Profiler Overlay check box. A statistics display is added to the window (Figure 5-2).

5.2.2. Displaying OSG Statistics



If you display OSG statistics and VR-Vantage statistics at the same time, they overlay each other.

To display OSG performance statistics:

1. Press **F2**. The frame rate is displayed.
2. Continue pressing **F2**. Additional statistics are displayed. When all statistics are displayed, pressing **F2** closes the display.

5.3. Preloading Models

When VR-Vantage discovers simulation objects on the network, it must load the models for these objects. This can affect performance. It is particularly noticeable when VR-Vantage first connects to an exercise, but can occur at other times if there is a sudden influx of new objects. You can reduce the likelihood of this problem by loading models before you connect to an exercise.

To preload models, you create a list of the models that you expect to need. VR-Vantage loads these models when it starts up. You can also load them immediately.

While VR-Vantage is participating in an exercise, you can capture a list of models that have not been preloaded. You can then add this list of models to the default preload list or specify the list in a command-line argument.

By default, the list of models to be loaded is saved to *./appData/settings/<application>/default_ModelPreload.mpsx*. You can export the list to a different file name. You can also import and merge lists.

To configure the list of models for preloading:

1. Choose **Settings** → **Application**. The Application Settings dialog box opens.
2. Select the Model Preload Configuration page (Figure 5-4).

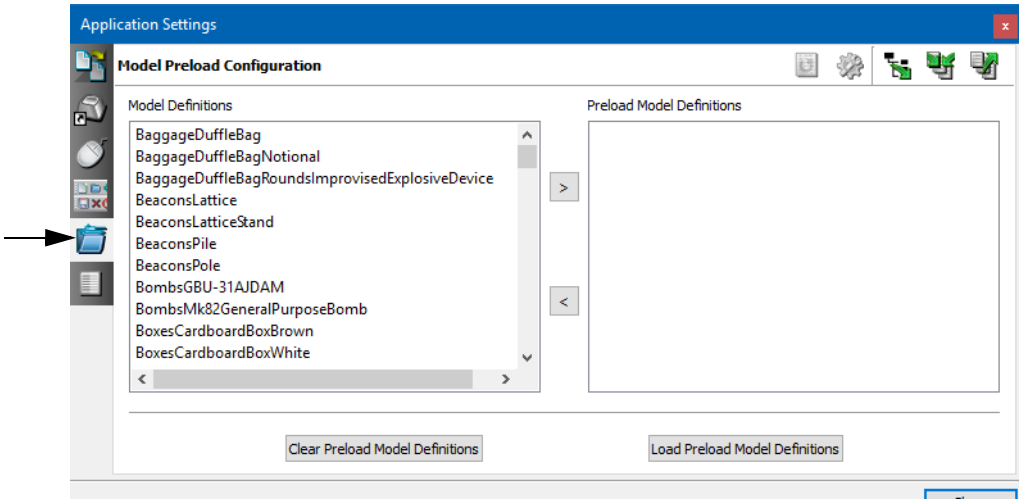


Figure 5-4. Model Preload Configuration page

3. In the Model Definitions list, select the models that you want to preload. Use the standard methods of multiple object selection.
4. Click the right-facing arrow to move the models into the Preload Model Definitions list.
5. To load the models immediately, click Load Preload Model Definitions.

5.3.1. Command-Line Options for Preloading Models

By default, when VR-Vantage starts, it loads the models listed in *default_ModelPreload.mpsx*. You can specify that it load a different model list with the `--modelPreloadFileName` command-line option. The syntax is:

```
--modelPreloadFileName filename
```

where *`filename`* is the name of the file to load.

While VR-Vantage is participating in an exercise, it can capture a list of models that were not preloaded. You can then merge this list with the default preload file or specify it as an alternative to the default file on the command line. To capture a list of models, use the following command:

```
--discoveredModelPreloadFileName filename
```

where *`filename`* is the name of the file to save the list to.

5.4. Clearing the Model Instancing Cache

VR-Vantage has a feature called model instancing that lets models share resources, such as geometry and textures. These shared resources are stored in a cache. Use of model instance caching decreases memory use and improves performance.

You can clear the model instancing cache if you need the disk space. You can clear it automatically whenever you create a new scene, or you can clear it manually.

To clear the model instancing cache after each scene change:

1. Choose **Settings** → **Application**. The Application Settings dialog box opens.
2. Select the File Caching Settings page (Figure 5-5).

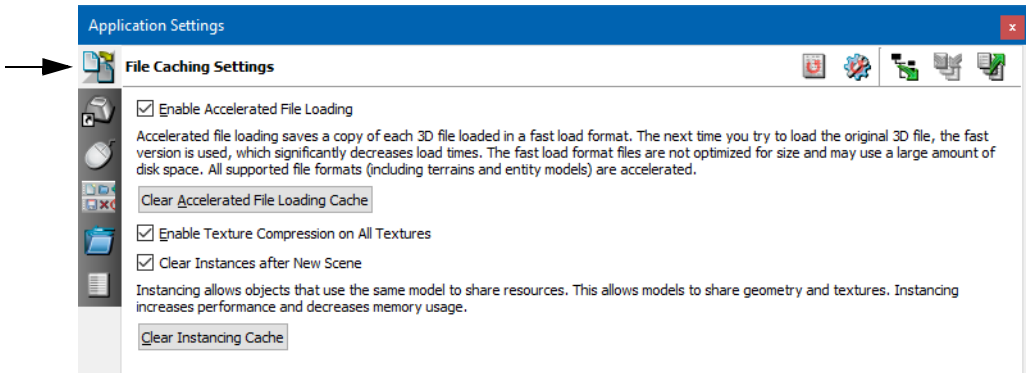


Figure 5-5. File Caching Settings page

3. Select the Clear Instances After New Scene check box.

To clear the model instancing cache manually:

1. Choose **Settings** → **Application**. The Application Settings dialog box opens.
2. Select the File Caching Settings page (Figure 5-5).
3. Click Clear Instancing Cache.

5.5. Configuring VSync

VSync (vertical synchronization) synchronizes VR-Vantage's frame rate with your monitor's refresh rate. This prevents a visual anomaly called tearing, in which parts of two frames of data can be visible at the same time, but it caps the maximum render frame rate based on the display. Typically this cap is 60hz.

In some cases, if the frame rate cannot quite keep up with the refresh rate, VSync can force the frame rate to a noticeably lower rate. To avoid this problem, we recommend that you enable triple buffering for your video card. You must do this in the video card's configuration application; VR-Vantage cannot do this for you.

By default, VSync is disabled. If you want to enable it, start up with the `--enableVSync` command-line option, or you can configure your video card to force it to be on.



For `--enableVSync` to work, your graphics card must be set to "Use application setting".

5.6. Configuring Graphics Quality

VR-Vantage has command-line options that let you configure graphics quality. In particular, depending on the capabilities of your graphics card, you may need to adjust the default settings for anti-aliasing using `--antiAliasing` *`level`*. Anti-aliasing is a technique used to reduce the jagged edges of digital graphics. You can set the level to 0, 2, 4, 8, or 16. Default: 4.

On older graphics cards, you may need to set anti-aliasing to 0 to improve performance.

5.6.1. Alpha to Coverage

Alpha to coverage is a way of handling objects with transparency that improves rendering of objects that have a sharp transition between opaque and transparent. Grass, trees, and fences are a few examples of when alpha to coverage is a good option, as shown in Figure 5-6. In addition to improving visualization of some objects, there is a slight performance improvement when alpha to coverage is enabled.



Off On

Figure 5-6. Alpha to coverage

You can set alpha to coverage for models and terrains. For models, you can enable it globally or in specific model definitions. For terrains, you can enable it globally or for specific terrain patches.

By default, alpha to coverage is enabled globally for models and terrains.

If alpha to coverage is enabled, you can enable a dither shader that modifies the alpha with a noise pattern so that there are no hard edges between different alpha levels. The trade-off for this setting is that it adds a small amount of noise as the object or camera moves.

VR-Vantage uses render bin 1 for the alpha to coverage render bin. You can change the render bin with the VRV_ALPHA_TO_COVERAGE_RENDERBIN environment variable.

Enabling Alpha to Coverage for Models

You can enable alpha to coverage globally for models and you can specify it for specific models. When you enable alpha to coverage globally, it affects all models that do not have it specified specifically in their model definition.

To enable or disable alpha to coverage for a specific model, add the alphaToCoverageEnabled parameter to the model definition and set it to true or false. For information about editing model definitions, please see "Editing a Model Definition," on page 32-8.

To enable or disable alpha to coverage globally for models:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 29-7).
3. Select or clear the Enable Alpha to Coverage Globally for Models check box.

Enabling Alpha to Coverage for Terrains

You can enable alpha to coverage globally for terrains or for specific terrain patches. The global setting affects all terrain patches that do not have an alpha to coverage option specified in their MTF file.

You enable alpha to coverage for a terrain patch when you add it. For details, please see "Adding Elevation Data (Terrain Patches) to a Terrain," on page 25-7.

To enable or disable alpha to coverage globally for terrains:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 29-7).
3. Select or clear the Enable Alpha to Coverage Globally for Terrain Patches check box.

Enabling the Dither Shader for Alpha to Coverage

To enable or disable the dither shader for alpha to coverage:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 29-7).
3. Select or clear the Enable Dither for Objects that Use Alpha to Coverage check box.

5.7. Using Incremental Compiling with Streamed Data

Users of streamed terrains might want to use incremental compiling. This is an OSG option that allows the processing of streamed data to be spread across multiple frames, reducing performance spikes. To use it, create the environment variable OSG_ICO. (It does not need a value. It just needs to exist.)



Using OSG_ICO may cause LOD issues with other paged formats.

5.8. Coloring Draw Calls

One of the factors that affects performance is the number of draw calls it takes to render a scene. VR-Vantage lets you color the draw calls in a scene so that you can see which objects may be creating performance problems.

When you color draw calls, all the drawables in the scene are randomly colorized. The colors themselves do not convey any meaning. The important information is the number of draw calls required for a model. For example, if a vehicle has 25 different colored parts, you know that it requires 25 draw calls and you may want to edit the model to use fewer of them.

Figure 5-7 illustrates a tank with its drawables colorized.

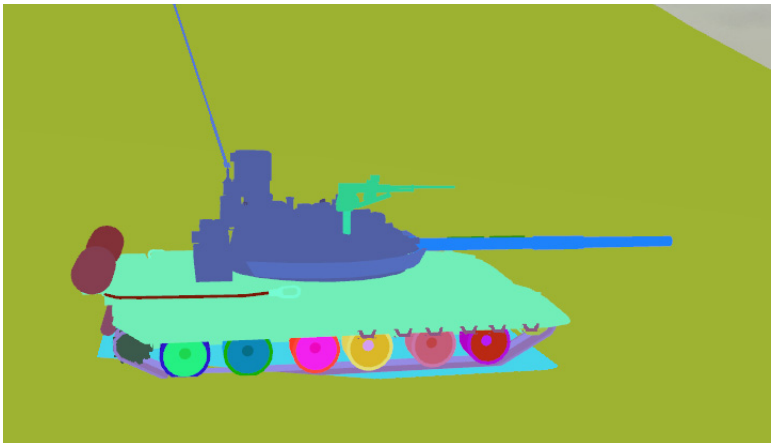


Figure 5-7. Colorized drawables on tank

The colors do not update as new information comes into the scene. Therefore, you can refresh the colorization.

To visualize draw calls:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 5-3).
3. Select the Colorize Per Drawable check box.
4. To refresh the display, click Refresh.

5.9. DI-Guy Performance Settings

The following settings affect performance of DI-Guy human characters:

- Maximum draw distance. VR-Vantage will not render characters that are farther away from the observer than this distance.
- Use multi-threaded update. Developers who are debugging DI-Guy may want to disable multi-threaded update to make it easier to debug.
- Enable instancing. Enabling instancing improves performance by reusing model data for similar characters. You can set the LOD beyond which instancing is used. You can visualize instanced characters.

To specify DI-Guy performance settings:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the DI-Guy Settings page (Figure 5-8).

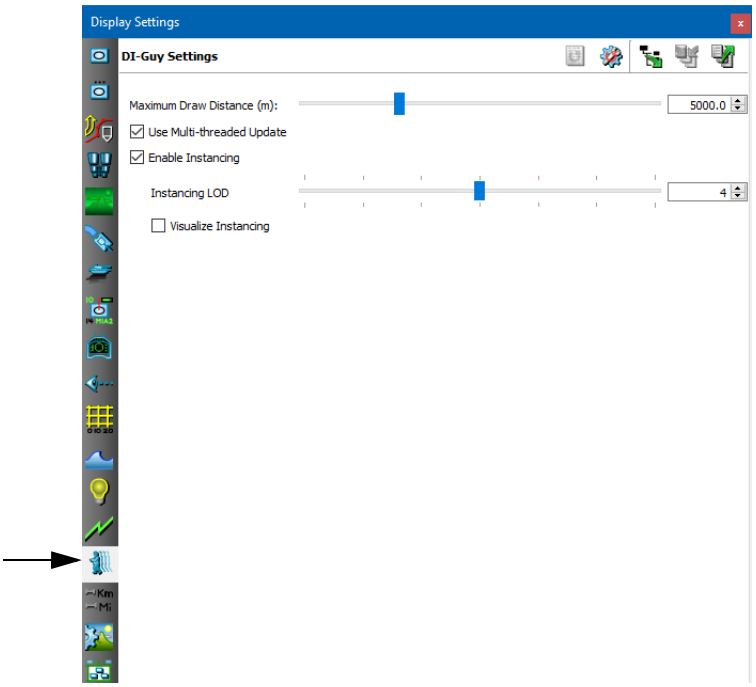


Figure 5-8. DI-Guy Settings page

3. To set the maximum draw distance, adjust the Max Draw Distance slider or change the value in the box.
4. Optionally, select or clear the Use Multi-threaded Update check box.
5. Optionally, select or clear the Enable Instancing check box.
6. If instancing is enabled, adjust the Instancing LOD slider or change the value in the box.
7. Optionally, select or clear the Visualize Instancing check box.

5.10. Using the Developer Debug Overlay

VR-Vantage uses the ImGui API to create an overlay that developers can use to test changes to some of its visual features. VR-Vantage includes an overlay for testing ocean settings, shadow settings, and other visual features. It also includes an ImGui test user interface that demonstrates some of its features.



Figure 5-9. Developer debug overlay



This feature is provided “as is” for the use of interested developers. For complete information about how to use ImGui, please see documentation at https://github.com/ocornut/imgui and other web pages.

To enable the developer debug overlay:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 5-3).
3. Select the Show Developer Debug Overlay check box. The Developer Debug Overlay opens. It has a menu for Ocean Settings.
4. To view shadow settings:
   a. In the Display Settings dialog box, select the Shadow Settings page.
   b. Select the Enable Shadow check box.
5. To view the ImGui Test UI, select the Show Imgui Test UI check box. (This check box is only available if you have enabled Shadow Settings.)



II. Display Engine, Video Streaming, Oculus Rift

This section describes display engines, which render the visual data in VR-Vantage. It also covers other capabilities that involve special hardware and video streaming.



6. Windows, Channels, and Display Engines

This chapter explains how to add and configure windows and channels and how to connect to remote display engines.

The VR-Vantage Display Engine 6-3
- How VR-Vantage Responds to Display Engine Connections 6-4
- The Display Engine's Connection States 6-4
- Windows and Channels 6-4
- Window Types 6-4

Managing Display Engine Configurations 6-5
- Adding a Window 6-5
- Adding a Channel to a Window 6-7
- Removing a Window 6-8
- Removing a Channel 6-8
- Saving a Display Engine Configuration 6-8
- Loading a Display Engine Configuration 6-9

Changing a Window's Attributes 6-11

Changing a Channel's Attributes 6-12
- Setting the Clipping Planes 6-14
- Specifying the Projection Resize Policy Attribute 6-16
- Changing a Channel's Frustum (Field of View) 6-19
- Changing the Viewport 6-20
- Configuring Water Visibility 6-21
- Channel Keywords 6-22

Starting a VR-Vantage Display Engine 6-22

Connecting to a Display Engine 6-23
- Disconnecting from a Display Engine 6-24

Configuring Multichannel Displays 6-24
- Changing the Camera's Position and Orientation Offset 6-26
- Creating a Multichannel Configuration 6-27
- Loading a Multichannel Configuration 6-27

Stereoscopic Displays 6-28
Configuring Anaglyphic Stereo 6-29
Configuring Polarized Stereo 6-30
Taking Screen Captures 6-31
Specifying the Location of Screen Captures 6-31
Display Issues on Linux 6-32

6.1. The VR-Vantage Display Engine

VR-Vantage applications have a component called a display engine. The display engine renders visual data. A display engine can also control the display on remote display engines.

VR-Vantage includes an application called the VR-Vantage Display Engine. A VR-Vantage Display Engine has a very limited GUI. Its primary purpose is to provide windows that can be controlled by a VR-Vantage application's (master) display engine.

The master display engine can connect to multiple VR-Vantage Display Engines (typically running on other computers) and control what is displayed in their windows.



For the remainder of this manual, we will continue to use VR-Vantage as an inclusive term for referring to VR-Vantage applications, will assume that you remember that VR-Vantage has a display engine that can control VR-Vantage Display Engine applications, and will refer to VR-Vantage Display Engine applications as display engines or external display engines (to distinguish them from the display engine in VR-Vantage).

VR-Vantage drives all of the display engines it is connected to, as illustrated in Figure 6-1.

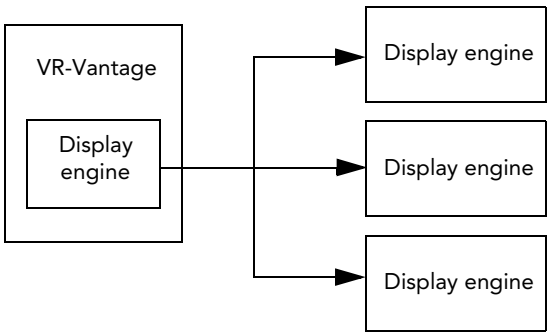


Figure 6-1. VR-Vantage and display engines



Display engines must have the same terrains, models, and other visualization data available as VR-Vantage. To replicate settings from a VR-Vantage application to a display engine, copy the data directories (*./appData*, *./data*, and *./userData*) from the computer on which VR-Vantage is installed to the computer on which the display engine is installed. Then import the settings into any Settings page that you customized on the VR-Vantage machine.

6.1.1. How VR-Vantage Responds to Display Engine Connections

VR-Vantage reloads the current scene each time a display engine is connected. This ensures that the display engine has an exact duplicate of the current scene. VR-Vantage does not perform any special actions when a display engine disconnects.

6.1.2. The Display Engine's Connection States

An external display engine is initially in an unconnected state. While it is unconnected, it cannot display a scene. When VR-Vantage connects to a display engine, the display engine creates an identical copy of the scene regardless of whether or not it can render it. The display engine is automatically sent a default display engine configuration that has a single channel on a single window. However, you can change this at runtime or programmatically. You could also load a saved configuration. If a display engine loses its connection to VR-Vantage, it clears its local scene and returns to an unconnected state.

6.1.3. Windows and Channels

A display engine (including the master display engine) can display multiple windows, each of which can have zero or more channels. The windows are all configured in VR-Vantage as part of a display engine configuration. You can save a display engine configuration and load it at a later time.

6.1.4. Window Types

To view a scene and simulation data, a display engine must have a window. VR-Vantage supports the following types of windows:

- Full screen. A full screen window uses the entire area of a monitor. It does not have any menus, panels, or other GUI controls.
- Resizable. A resizable window can be moved, resized, maximized, and minimized. It does not have any GUI controls. An inset view is an example of a resizable window. (For information about inset views, please see "Inset Views," on page 17-2.)
- Embedded. An embedded window is part of an application. The main display window in VR-Vantage is an embedded window. The application provides a variety of controls that affect the view in the window. You could, if you wanted to, change VR-Vantage's window to a different type, in which case it would no longer be embedded in the main application window.

A window can have zero or more channels. Each channel defines a viewable area on the screen. You must have at least one channel to see a rendered image in the window. Each channel can have its own observer.

6.2. Managing Display Engine Configurations

A VR-Vantage application has one display engine. You can add windows to the display engine and you can add channels to the windows. The windows and channels make up a display engine configuration. You can save a display engine configuration and load a saved configuration. Chapter 7, *Display Engine Tutorials*, shows how to edit a channel's attributes and how to save and load display engine configurations.

6.2.1. Adding a Window

You can add as many windows as you want. New windows are named for the window type, for example Stealth Inset 1, Stealth Inset 2, Plan View Inset 1, and so on. Each window is created with one channel. You can add new windows from the File menu or in the Display Engine Configuration Editor.

➤ To add a window from the File menu, choose **File** → **New Stealth Window** (or **New Plan View Window**). (The new window commands that are available on the File menu depend on which VR-Vantage application you are running.) A new window opens (Figure 6-2).

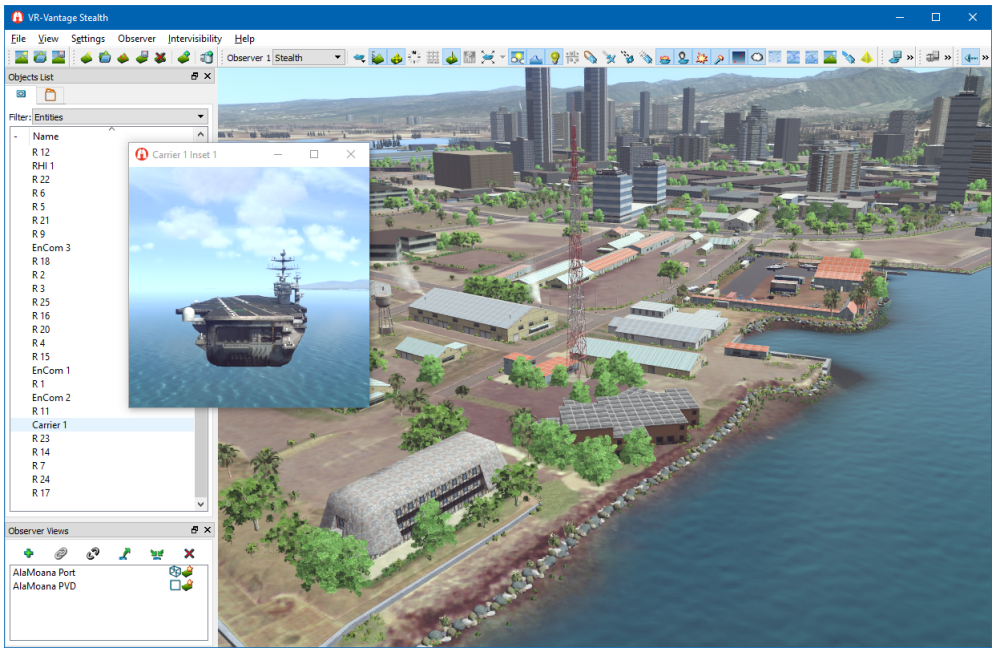


Figure 6-2. New window

To add a window from the Display Engine Configuration Editor:

1. Choose **View** → **Display Engine Configuration Editor Panel**. The Display Engine Configuration Editor Panel is added to the VR-Vantage window (Figure 6-3).

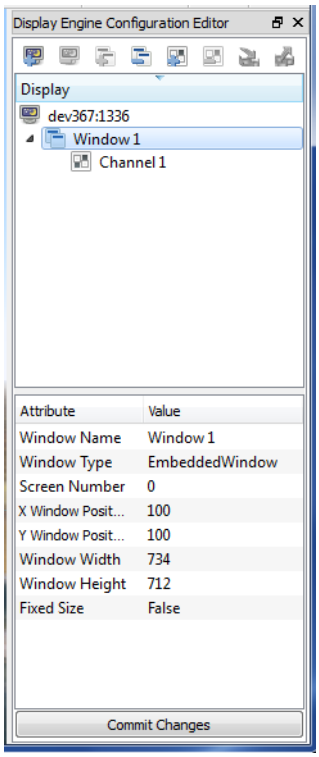


Figure 6-3. Display Engine Configuration Editor Panel



The Display Engine Configuration Editor Panel is available only in a VR-Vantage application, not on remote display engines.

2. Select the display engine at the top of the Display tree.
3. Click the Add a Window button (), or right-click the display engine name and choose **Add a Window** on the context sensitive menu. A new window opens and is listed in the Display Engine Configuration Editor Panel (Figure 6-4).

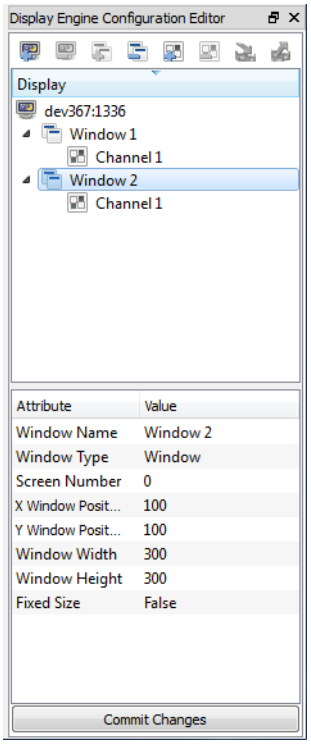


Figure 6-4. New window

6.2.2. Adding a Channel to a Window

Adding a channel to a window lets you configure multiple views within that window.

To add a channel to a window:

1. In the Display Engine Configuration Editor Panel, select the window to which you want to add a channel.
2. Click the Add a Channel button (), or right-click the window name and choose **Add a Channel** from the menu.



To see a difference between the channels, change the attributes of the new channel (Figure 6-11). For details about editing channel attributes, please see "Changing a Channel's Attributes," on page 6-12.

6.2.3. Removing a Window

To remove a window:

1. In the Display Engine Configuration Editor Panel, select the window that you want to remove.
2. Click the Remove Window button (), or right-click the window name and choose **Remove Window** from the menu.

6.2.4. Removing a Channel

To remove a channel:

1. In the Display Engine Configuration Editor Panel, select the channel that you want to remove.
2. Click the Remove Channel button (), or right-click the window name and choose **Remove Channel** from the menu.

6.2.5. Saving a Display Engine Configuration

You can save a display engine configuration so that you can easily replicate the configuration at a later time.



If you are running two or more display engines, you cannot save the configurations for all of them at once. You have to save the configuration for each display engine individually.

To save a display engine configuration:

1. In the Display Engine Configuration Editor Panel, select the display engine whose configuration you want to save.
2. Click the Save Display Engine Configuration button (), or right-click the name of the display engine and choose **Save Display Engine Configuration** on the menu. The Save Display Engine Configuration dialog box opens.
3. Type a name for the display engine configuration.
4. Click Save.

6.2.6. Loading a Display Engine Configuration



You cannot load multiple display engine configurations at the same time in the Display Engine Configuration Editor Panel. You can only load one configuration at a time. However, you can load multiple configurations from the command line. For details, please see "Loading Multiple Display Engine Configurations from the Command Line," on page 6-9.

To load a display engine configuration:

1. In the Display Engine Configuration Editor Panel, select the display engine at the top of the window.
2. Click the Load Display Engine Configuration button (), or right-click the name of the display engine and choose **Load Display Engine Configuration** on the menu. The Load Display Engine Configuration dialog box opens.
3. Select the display engine configuration you want to load.
4. Click Open.

Loading Multiple Display Engine Configurations from the Command Line

VR-Vantage users often set up multichannel configurations that they wish to reuse. (For details, please see "Configuring Multichannel Displays," on page 6-24.) You can load these configurations (or any set of configurations) in the Display Engine Configuration Editor Panel, as described in "Loading a Display Engine Configuration," on page 6-9. However, you cannot load them as a group. You must load each one individually. To automate the process of loading multiple display engine configurations, you can load them from the command line.

You load multiple display engine configurations from the command line with the `--dispSetting` command line argument and the `-W` or `--withDisplayEngine` command-line argument. The syntax on Windows is:

```
vrvIG.exe --dispSetting masterdisplayConfigFile
   -W port@netAddress;remotedisplayConfigFile1 ....
   -W port@netAddress;remotedisplayConfigFileN
```

The syntax for Linux uses a colon instead of a semi-colon:

```
vrvIG.exe --dispSetting masterdisplayConfigFile
   -W port@netAddress:remotedisplayConfigFile1 ....
   -W port@netAddress:remotedisplayConfigFileN
```

The `--dispSetting` argument specifies the display engine configuration file for the VR-Vantage application. The `-W` argument specifies the display engine configuration files for the VR-Vantage display engines. The display engine configuration file name can be an absolute path or a path relative to the executable. For example, suppose that you have saved display engine configurations for a master display engine and two remote display engines. You save the configuration for each display engine as follows:

- Master display engine — *../appData/settings/stealth/machine1.dcx*.
- First remote display engine — *../appData/settings/stealth/machine2.dcx*.
- Second remote display engine — *../appData/settings/stealth/machine3.dcx*.

To load these three configurations:

1. Run the remote display engines on the two remote machines.
2. Start VR-Vantage. On Windows use the following command line:

```
vrvIG.exe
  --dispSetting ../appData/settings/stealth/machine1.dcx
  -W 22563@machine2;
  ../appData/settings/stealth/machine2.dcx
  -W 22563@machine3;
  ../appData/settings/stealth/machine3.dcx
```

On Linux use:

```
vrvIG.exe
  --dispSetting ../appData/settings/stealth/machine1.dcx
  -W 22563@machine2:
  ../appData/settings/stealth/machine2.dcx
  -W 22563@machine3:
  ../appData/settings/stealth/machine3.dcx
```

6.3. Changing a Window's Attributes

You can change the following attributes of a window:

- Window Name.
- Window Type (for details about window types, please see "Window Types," on page 6-4).
- Screen Number.
- Position (X, Y coordinates of upper left corner, in screen pixels).
- Width and height, in pixels.
- Fixed size.

To change a window's attributes:

1. Select the window in the Display Engine Configuration Editor Panel. A list of attributes is displayed at the bottom of the editor (Figure 6-3).
2. Click the value of the attribute that you want to change.
3. To change a numeric value, type or select a value.

   To change the Window Type or Fixed Size, select a value from the list.

   To change the name, type a new name.
4. Click Commit Changes. The window is updated.



You can also change a window's position by dragging it to a new location. You can change its size by resizing the window directly. The new values are shown in the attributes list the next time you select the window in the Display Engine Configuration Editor Panel.

6.4. Changing a Channel's Attributes

You can change the following attributes of a channel:

- Channel Name.
- Observer Name.
- Sensor. Use the sensor configured for the current observer, or override the observer and use one of the listed sensors.
- Projection Units – field of view, in angles, or database, in meters.
- Projection Resize Policy – fixed, vertical, or horizontal.
- Frustum (Left, Right, Top, and Bottom) (field of view).
- Z Near and Z Far (clipping planes).
- Near Far Clip Policy.
- Reduce Z Fighting Ratio.
- Dynamic Near Clip Attached Policy.
- Attached Near Clip Altitude.
- Fixed Near Clip When Attached.
- Dynamic Water Visibility Enabled.
- Dynamic Water Visibility Maximum.
- Camera Position Offset (X, Y, Z).
- Camera Orientation Offset (Psi (heading), Theta (pitch), Phi (roll)).
- Viewport (Left, Right, Bottom, Top).
- Keywords – keywords to associate with the channel, for example, "showCockpits". (For a list of channel keywords, please see "Channel Keywords," on page 6-22.)

To change a channel's attributes:

1. Select the channel in the Display Engine Configuration Editor Panel. A list of attributes is displayed at the bottom of the editor (Figure 6-5).

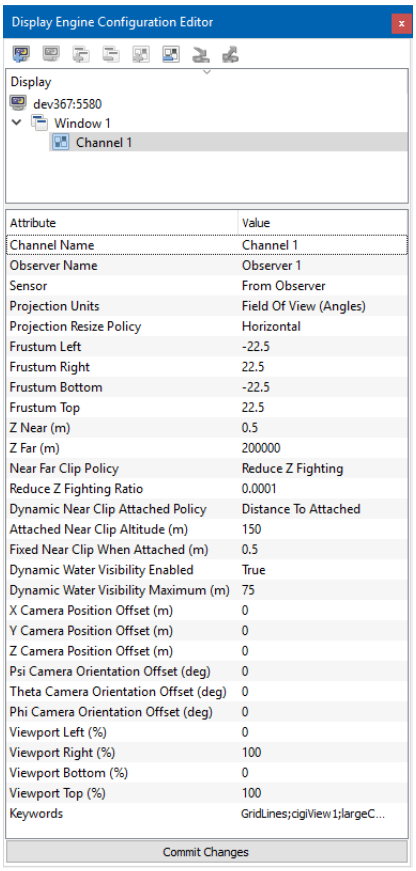


Figure 6-5. Channel attributes

2. Click the value of the attribute that you want to change.
3. To change a numeric value, type or select a new value.

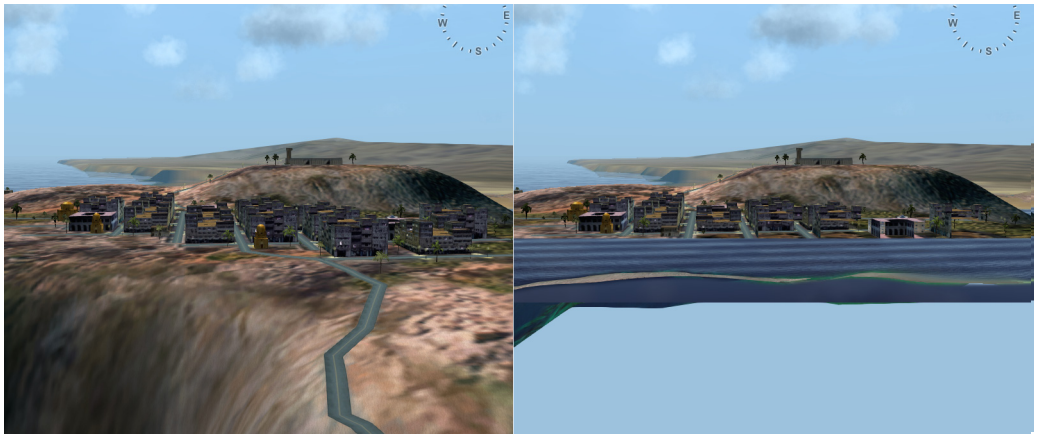
   To change the Observer Name, Sensor, Project Units, Projections Resize Policy, or Parent Channel, select a value from the list.

   To change the Channel Name, type a new name. (For details about some of these attributes, please see the sections that follow this one.)
4. Click Commit Changes. The window is updated.

6.4.1. Setting the Clipping Planes

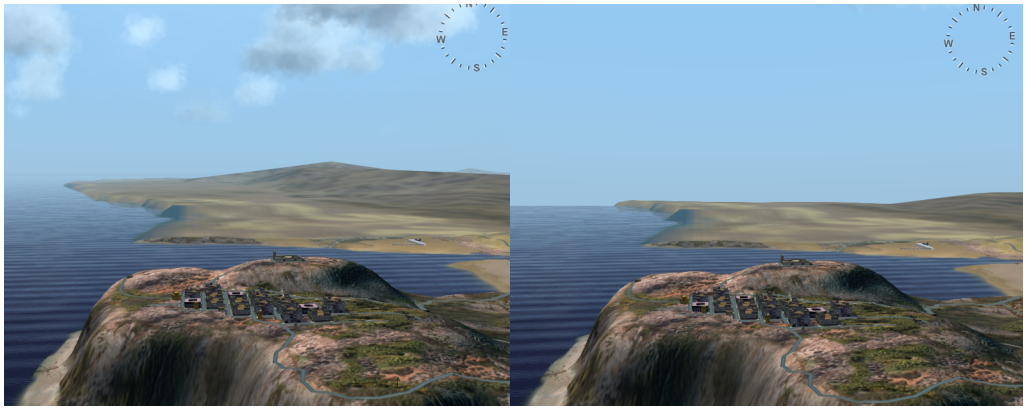
To improve performance, the graphics engine does not render anything that falls outside of a specified range of distances from the observer. This range is determined by the near and far clipping planes. Clipping planes are set on a per-channel basis.

Figure 6-6 illustrates the effect of changing the near clipping plane (the Z Near attribute). Figure 6-7 illustrates the effect of changing the far clipping plane (the Z Far attributes).



Z Near = 1 Z Near = 500

Figure 6-6. Near clipping



Z Far = 10000000 Z Far = 10000

Figure 6-7. Far clipping

The ratio between the far clipping plane and the near clipping plane determines the precision with which the renderer can determine differences in depth. If this ratio becomes too small, then terrain and objects that are distant may not sort correctly, resulting in what is referred to as Z-fighting. VR-Vantage provides different policies for managing the way near and far clipping planes are determined.

VR-Vantage sets the clipping planes using one of the following policies (the Near Far Clip Policy attribute), plus their modifiers:

- Fixed. Clipping planes are set explicitly using the Z Near and Z Far attributes.
- Reduce Near Clip. Clipping planes are adjusted dynamically based on the observer's altitude. The near clipping plane is increased slightly as the observer rises to 20,000 meters, but stays under 1.5 meters. When the observer rises above 20,000 meters, the near and far clipping planes gradually increase. The Reduce Near Clip policy can be modified by the Dynamic Near Clip Attached Policy.
- Reduce Z Fighting. Clipping planes are set based on a ratio of the near clipping plane to the far clipping plane (the Reduce Z Fighting Ratio attribute). The near clipping plane can get as large as needed to maintain the ratio. This policy is the default clipping plane policy. The Reduce Z Fighting policy can be modified by the Dynamic Near Clip Attached Policy.

The Reduce Z Fighting clipping plane policy greatly reduces Z-fighting when you are using dynamic ocean. However, if the observer is attached to a simulation object and its altitude is moderately high, it may be closer to the entity than the near clipping plane and the entity may be clipped away. You can adjust the ratio to provide the best results. If you are not using dynamic ocean or you are frequently attaching to entities, you may want to use one of the other clipping plane policies.

If you set the Near Far Clip Policy to Reduce Z Fighting or Reduce Near Clip and the observer is attached to a simulation object, VR-Vantage can dynamically change the near clipping plane based on the Dynamic Near Clip Attached Policy, as follows:

- Ignore Attached. Do not take entity attachment into account when setting the near clipping plane.
- Fixed Near Clip. Set the near clipping plane to the value of the Fixed Near Clip When Attached attribute if the observer altitude is >= the value of the Attached Near Clip Altitude attribute.
- Distance To Attached. Use the distance between the observer and the attached entity to determine the near clipping plane if the observer altitude is >= the value of the Attached Near Clip Altitude attribute. When this policy is selected, VR-Vantage compares the near clipping value without this modifier to the value with it and uses the smaller of the two values. This is the default policy.



If the settings described in this section do not reduce Z-fighting, you can try using reverse Z buffering. To enable reverse Z buffering, start VR-Vantage with the `--useReverseZBuffer` command-line option. However, the Oculus Rift HMD, simulation object labels, shadows, HUDs, and lightpoints do not work with reverse Z buffering. Reverse Z buffering is not supported in 2D observer modes.

6.4.2. Specifying the Projection Resize Policy Attribute

The Projection Resize Policy determines how a channel behaves when you resize it. The options are:

- Fixed. Keep the aspect ratio the same regardless of how the window is resized.
- Horizontal. Maintain the horizontal aspect ratio when the window is resized.
- Vertical. Maintain the vertical aspect ratio when the window is resized.

Figure 6-8 illustrates the effect of different resize policies, as follows:

- In the window with the fixed resize policy, all of the visual data in the initial view is still in the resized view, although the vertical dimensions are distorted.
- In the window with the horizontal resize policy, the field of view is widened to maintain the correct aspect ratio of the visual data.
- In the window with the vertical resize policy, the field of view is shortened.

You can change the projection resize policy for individual channels by editing the Projection Resize Policy attribute. You can also change the projection resize policy for all channels.

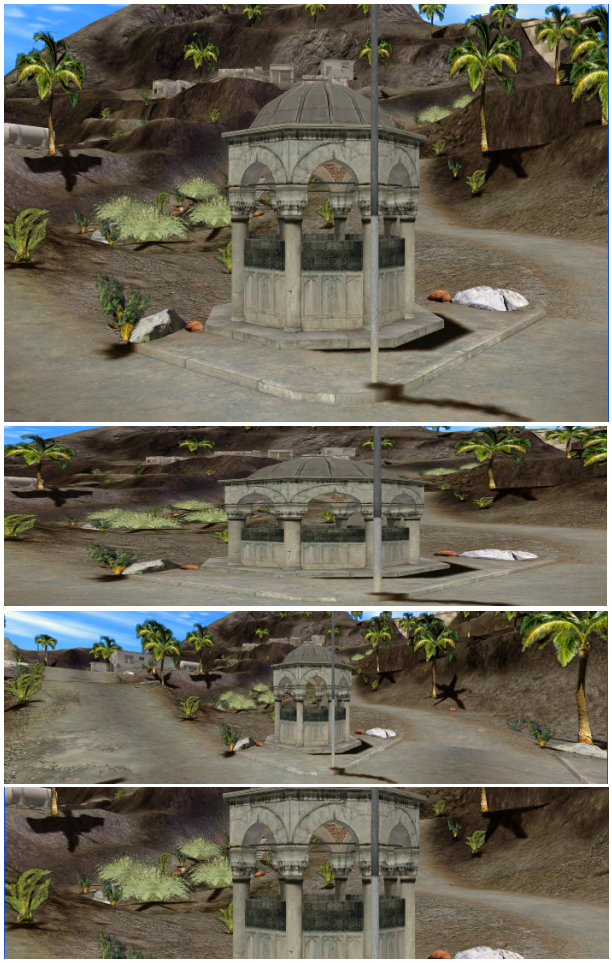


Figure 6-8. Effect of different projection resize policies

To change the projection resize policy for all channels:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 6-9).

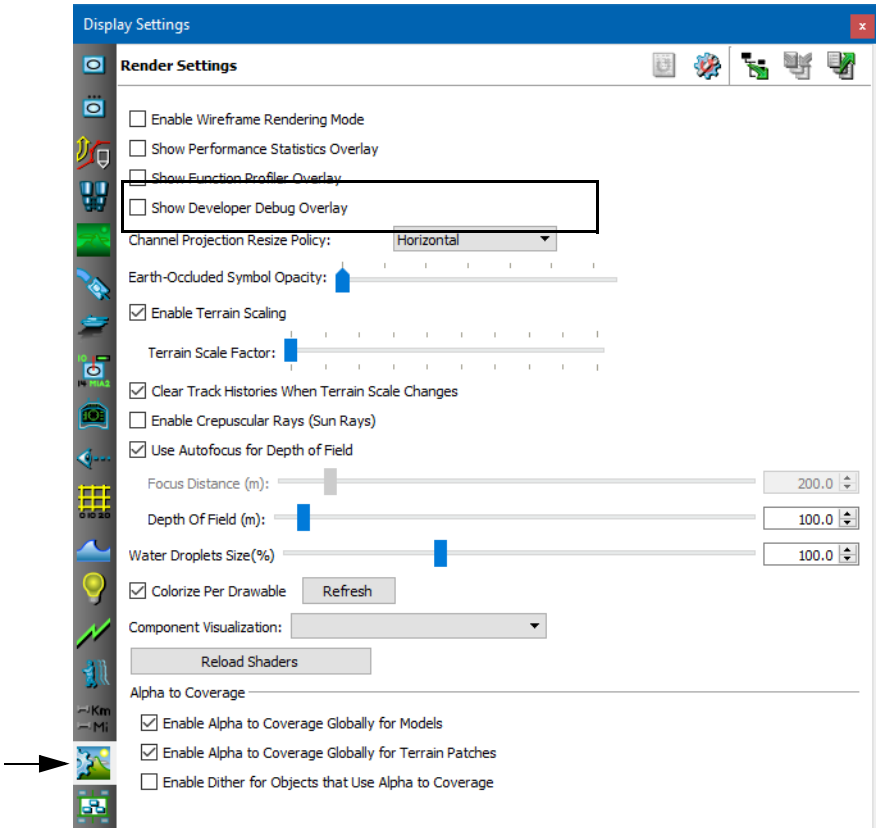


Figure 6-9. Display Settings, Render Settings page

3. Select an option on the Channel Projection Resize Policy list.

6.4.3. Changing a Channel's Frustum (Field of View)

The field of view setting affects perspective in a manner similar to a camera lens.

A wide field of view creates an effect like that of a wide-angle lens. Objects appear smaller and farther away from the observer, since the observer coverage spans a wider area. Depth distances between objects become exaggerated.

A narrow field of view creates an effect like that of a telephoto lens. Objects appear larger and closer to the observer, and the overall scene depth appears flattened. The distances between objects appears compressed.

Figure 6-10 illustrates different field of view settings taken from the same observer location. The view on the left has frustum values -22.5, 22.5, -22.5, 22.5. The view on the right has the frustum values -10, 10, -10, 10.

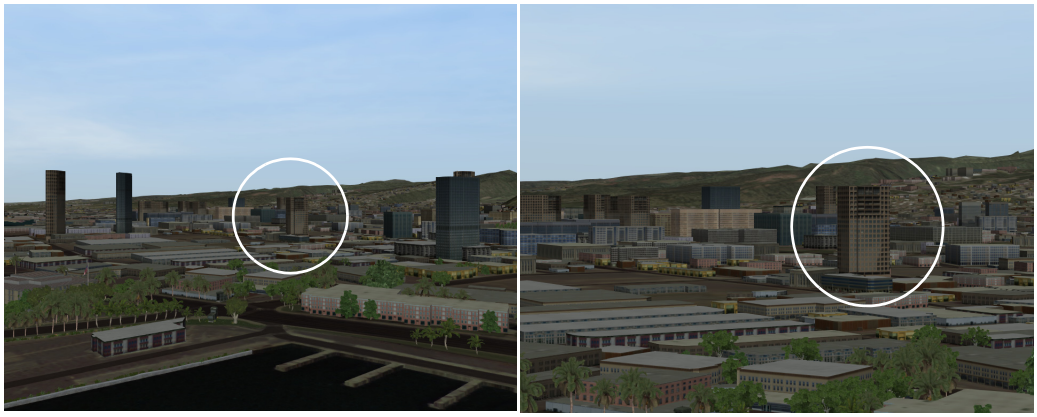


Figure 6-10. Field of view

➤ To change a channel's field of view, edit the frustum values in its attributes list.

6.4.4. Changing the Viewport

When you change the viewport of a channel, you change the portion of the window in which the channel is displayed. This does not change the extents of the scene. A narrow viewport compresses the scene width. A short viewport compresses the height. Figure 6-11 shows a window that has three channels, each with a different viewport. (The arrows show which portion of the window corresponds to each channel.) Notice that each channel has the same field of view.

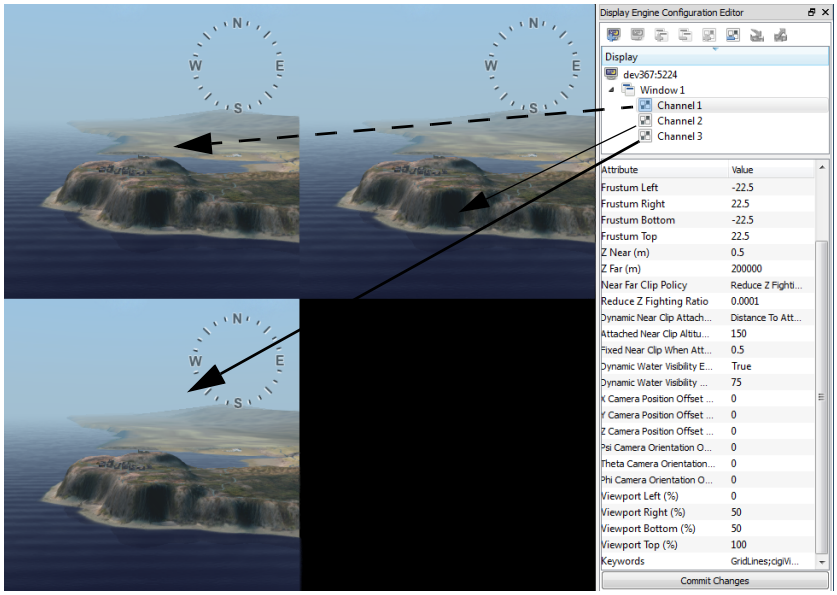


Figure 6-11. Window with three channels and three viewports

Viewports are specified as a percentage of the window from left to right and top to bottom. Figure 6-12 shows the percentage allocation for the viewports in (Figure 6-11).

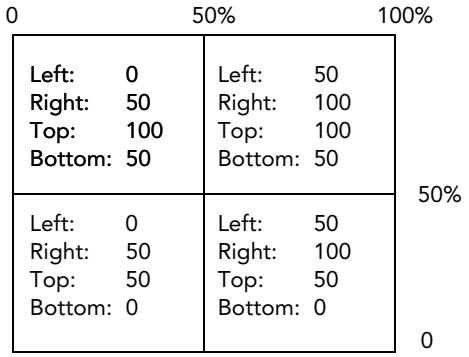


Figure 6-12. Viewport configuration



When you change viewports, the window might not completely refresh. To see the changes, resize the window.

6.4.5. Configuring Water Visibility

The water visibility attributes (along with the surface transparency setting on the Scene Settings dialog box, Environment Conditions Settings page) are designed to reduce Z fighting. Z-fighting occurs when the altitude of the ocean surface and the depth of the ocean floor are very close to each other, usually around coastlines that have bathymetry data. The likelihood of Z-fighting decreases as the difference between the ocean surface and ocean floor increases, but it increases as the observer's altitude increases (because the ocean surface and ocean floor are proportionally closer to each other). Therefore there is a dynamic relationship as these factors change.

Adding transparency to the water reduces Z-fighting. The Dynamic Water Visibility Enabled attribute increases transparency automatically as the observer's altitude increases. Dynamic Water Visibility Maximum specifies the maximum depth to which transparency is added (Figure 6-13).

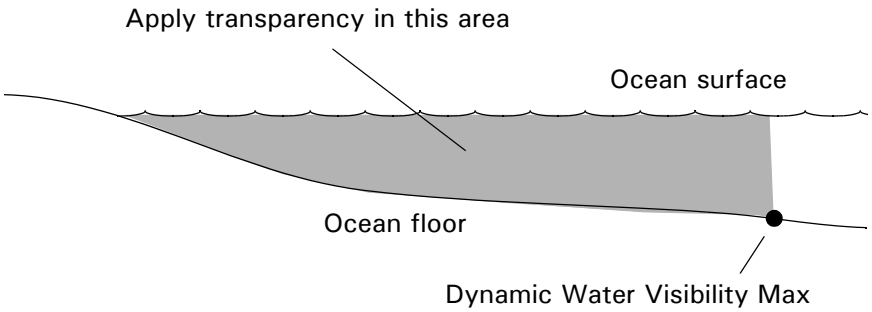


Figure 6-13. Dynamic water visibility

Dynamic water visibility is enabled by default. When Dynamic Water Visibility Enabled is True, VR-Vantage increases the water visibility depth from the value Surface Transparency setting (at 0 meters altitude), to the maximum value specified at the current Ocean LOD Elevation. The default depth is 75 meters. This value provides reasonable performance given the default Ocean LOD Elevation (5000 meters), the altitude above which VR-Vantage stops displaying dynamic ocean. If you change the Ocean LOD Elevation so that dynamic ocean is displayed when the observer is higher than 5000 meters, Z-fighting may increase and you may need to increase the Dynamic Water Visibility Maximum value. (For details about setting the Ocean LOD Elevation, please see "Setting the Ocean LOD Elevation," on page 25-18.)

There is no performance cost to enabling this attribute when you are not using dynamic ocean. If there are no underwater terrain polygons, these attributes have no effect.

6.4.6. Channel Keywords

VR-Vantage supports the following channel keywords:

- largeCompass. If the compass is enabled, this channel shows the large compass. Default: Included in configuration of all channels.
- smallCompass. If the compass is enabled, this channel shows the small compass. Default: Not included in the configuration of any channels.
- GridLines. If grid lines are enabled, this channel shows the grid lines. Default: Included in configuration of all channels.
- showRemoteGraphics. If remote graphics are received this channel displays them. Default: Included for all channels except inset views.
- showCockpits. If GL Studio cockpits are enabled, this channel shows them. Default: Included in configuration of all channels.
- NoExternalInput. This keyword disables all mouse/keyboard/touch events for the channel. Default: Only included in the configuration of Sensor Gimbal View channels (controlled by remote VR-Forces).
- cockpitOnly. If this keyword is true, the scene will not be rendered in the channel. Default: Included in the configuration of instrument panels only.
- cigiView1. CIGI provides for the definition of views and view groups, by integer ID. Subsequent CIGI view controls that address a given ID will only be applied to the channel or channels with the cigiViewX keyword, where X is the view ID. Default: cigiView1 is included for all channels except inset views.

6.5. Starting a VR-Vantage Display Engine

On Windows, you can start a display engine from the Start menu or by running a batch file. On Linux, you start a display engine from a script.

➤ To start a display engine on the Windows Start menu, choose **All Programs** → **MAK Technologies** → **VR-Vantage 2.3** → **VR-Vantage Display Engine**.

➤ To start a display engine from the Windows console or on Linux, run:

`./bin64/displayEngine.`*`extension`*

where *`extension`* is the appropriate script extension for the platform.

6.6. Connecting to a Display Engine

If you want to control a display engine, you must first connect to it.

Connections are made using TCP networking. All computers running display engines together must be on the same network. The display engines listen for VR-Vantage on a TCP socket. A display engine can only be connected to a single master application.

! If you have trouble connecting to a remote display engine, make sure that your firewall is not blocking the port you are using to connect.

---

i It is possible to run multiple display engines on the same machine with different ports. However, this is not a recommended configuration.

To connect to a display engine:

1. On one computer (the master), run VR-Vantage.
2. On another computer, run the VR-Vantage Display Engine application, as described in "Starting a VR-Vantage Display Engine," on page 6-22.
3. On the master computer, choose **View** → **Display Engine Configuration Editor**. The Display Engine Configuration Editor Panel opens (Figure 6-3).
4. In the Display Engine Configuration Editor Panel, click the Connect a Display Engine button (), or right-click the main display engine icon and choose **Connect a Display Engine** on the popup menu. The Connect To Display Engine dialog box opens (Figure 6-14).

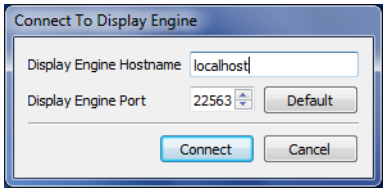


Figure 6-14. Connect to Display Engine dialog box

5. In the Display Engine Hostname box, type the host name or IP address of the computer on which the display engine is running.
6. Optionally, in the Display Engine Port box, specify a port for the display engine. The default is 22563. In most cases the default value will work well.
7. Click Connect. The display engine is added to the Display Engine list. A window is added to the display engine. It shows the same view as Window 1 on VR-Vantage.

6.6.1. Disconnecting from a Display Engine

To disconnect from a display engine:

1. In the Display Engine Configuration Editor Panel, select the display engine that you want to disconnect.
2. Click the Disconnect a Display Engine button (), or right-click the display engine name and choose **Disconnect a Display Engine** on the menu.

> You cannot disconnect the master display engine.

6.7. Configuring Multichannel Displays

People running simulations often want to set up a multichannel configuration using multiple monitors (on one or more computers) to display exercises. Typical configurations are a horizontal alignment, which widens the view, and the angled alignment, which surrounds the observer to simulate an out-the-window view (Figure 6-15). Since VR-Vantage just sends out its view, if you do not configure the displays for multichannel, the subordinate monitors display the exact same view as the master (Figure 6-17, default window views). Therefore, for multiple views, you must configure the channels to calculate an offset from the master view.



Figure 6-15. Multi-channel displays (top-down view)

The distribution of the view across multiple channels is controlled by the frustum. Figure 6-16 illustrates the default frustum for Channel 1. It has a 60° field of view, from -30° to +30°.

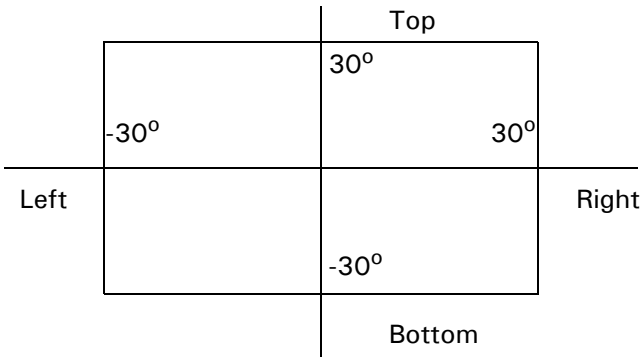


Figure 6-16. Default frustum for Channel 1

If you wanted to split the view into, for example, three horizontal channels, you would edit the left and right frustum values for each channel, as shown in Figure 6-17. The field of view is the same: -30° to +30°. However, now each channel shows a 20° segment of the total field of view.

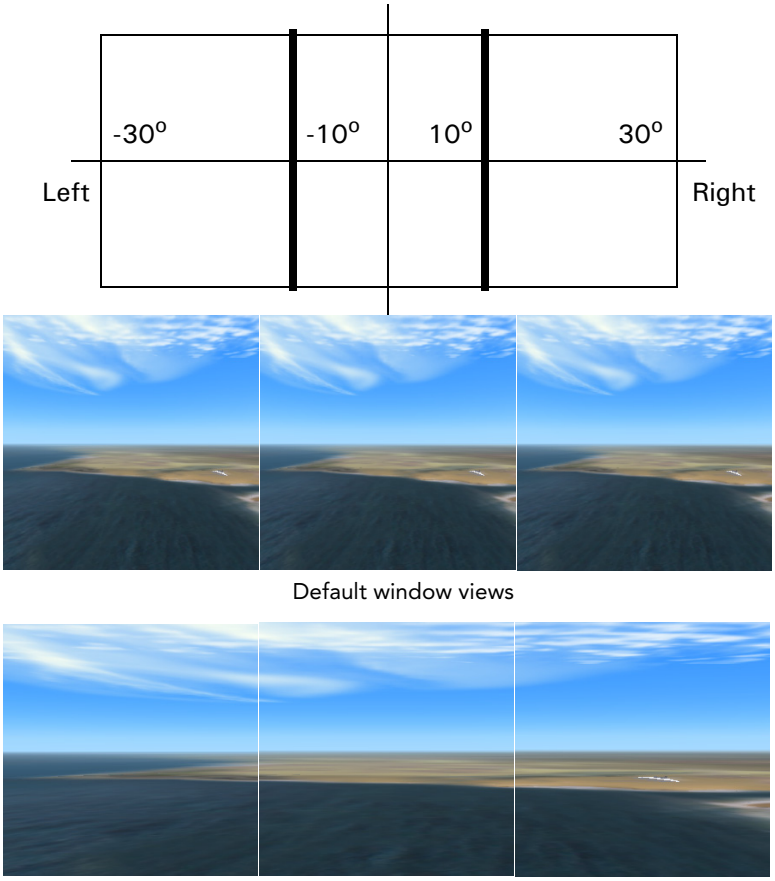


Figure 6-17. Frustum values for three channels



For a multi-monitor configuration to work correctly, use fixed size windows whose size matches the aspect ratio specified by the horizontal and vertical fileds of view and set the projection resize policy to fixed. (For information about the projection resize policy, please see "Specifying the Projection Resize Policy Attribute," on page 6-16.)

6.7.1. Changing the Camera's Position and Orientation Offset

Changing the frustum affects how much of the database is in the view. Changing the camera's position and orientation affects the point from which you look at the view. For example, imagine you are simulating the view from a vehicle, as in vehicle 1 in Figure 6-18. (The camera location is represented by the binoculars.)

If you wanted to show the view as experienced by a gunner riding in the back of the vehicle, or in a turret, you would change the X, Y, and Z camera position offsets in the channel's attribute list (vehicle 2).

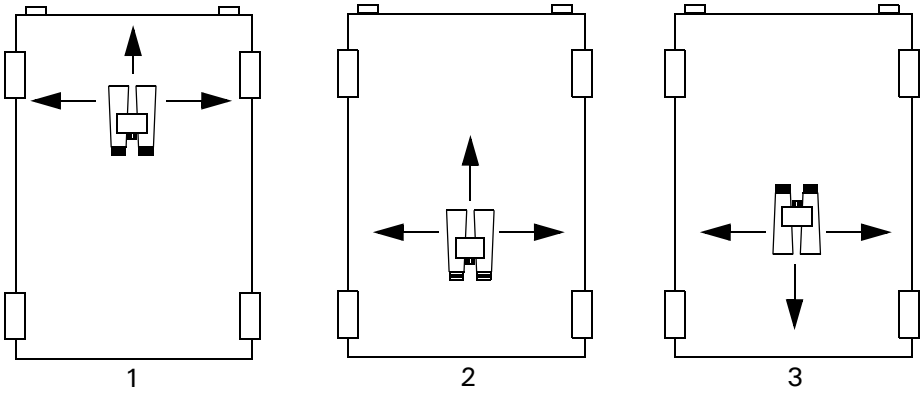


Figure 6-18. Position and orientation offsets

If you wanted the gunner to look out the rear of the vehicle, you would change its orientation, with the Psi (heading), Theta (pitch), and Phi (roll) Camera Orientation Offsets (vehicle 3).

6.7.2. Creating a Multichannel Configuration

A multichannel configuration is a set of display engine configurations. You must save each configuration individually. Chapter 7, *Display Engine Tutorials* shows how to create, save, and load a multichannel display engine configuration.

To save a multichannel configuration:

1. Start VR-Vantage and one or more VR-Vantage Display Engines.
2. In the Display Engine Configuration Editor Panel, connect the display engines.
3. Configure the display engines with the windows and channels that you want them to have.
4. For each display engine in the Display Engine Configuration Editor Panel:
   a. Select the display engine.
   b. Click the Save Display Engine Configuration button ( ).
   c. Type a name for the configuration.
   d. Click Save.

6.7.3. Loading a Multichannel Configuration

You can load a multichannel configuration in the Display Engine Configuration Editor Panel or from the command line. If you load it in the editor, you must load each configuration individually. For details about loading multiple configurations from the command line, please see "Loading Multiple Display Engine Configurations from the Command Line," on page 6-9.

To load a multichannel configuration:

1. Start VR-Vantage and one or more VR-Vantage Display Engines.
2. In the Display Engine Configuration Editor Panel, connect the display engines.
3. For each display engine in the Display Engine Configuration Editor Panel:
   a. Select the display engine.
   b. Click the Load Display Engine Configuration button ( ).
   c. Select the configuration you want to load.
   d. Click Load.

6.8. Stereoscopic Displays

Stereoscopic display produces realistic three dimensional imagery by showing each of the viewer's eyes a different image, rendered from that eye's point of view. A stereoscopic display requires a system that can render two channels, one for each eye, and a method of displaying the correct channel to the correct eye of the viewer. Three common display methods are active stereo, polarized display, and anaglyphic stereo. OpenSceneGraph can support all three of these methods, and since VR-Vantage uses OpenSceneGraph for most of its rendering, all three should be possible in VR-Vantage.

- Active Stereo: In an active stereo setup, frames are alternated on the display for the left eye and the right eye, and a pair of shutter glasses blocks light to either the left or right eye in synchronization with the display. This method requires hardware support to synchronize the glasses to the display, along with a graphics card capable of quad buffering, and a display that can refresh at twice the desired frame rate. Active stereo should be possible with VR-Vantage, but has not been tested.
- Anaglyphic Stereo: Anaglyphic stereo is the cheapest and simplest method to use. A typical setup is a display showing a red channel and a cyan channel simultaneously, along with a pair of glasses with one red lens and one cyan lens for the viewer. This will work with no special hardware aside from the inexpensive glasses. It is also the simplest method to set up in VR-Vantage, requiring only that some OpenSceneGraph environment variables be set.
- Polarized Stereo: A polarized display shows the frames for both eyes simultaneously, but with their light polarized in different directions. The viewer wears glasses that have lenses polarized to match the two different frames, allowing each eye to see only the frame intended for it. Polarized stereoscopy requires some specialized equipment to display the images with the correct polarization. One simple method for producing this kind of display is to use two projectors with polarizing filters, projecting onto a rear projection screen. Monitors are available that can polarize the two frames. It is easy to configure a polarized display with VR-Vantage.

6.8.1. Configuring Anaglyphic Stereo

You can configure OpenSceneGraph for anaglyphic stereo rendering through environment variables. To use VR-Vantage in anaglyphic mode, run it with the environment variables described in Table 6-1.

Table 6-1: Environment variables for anaglyphic stereo

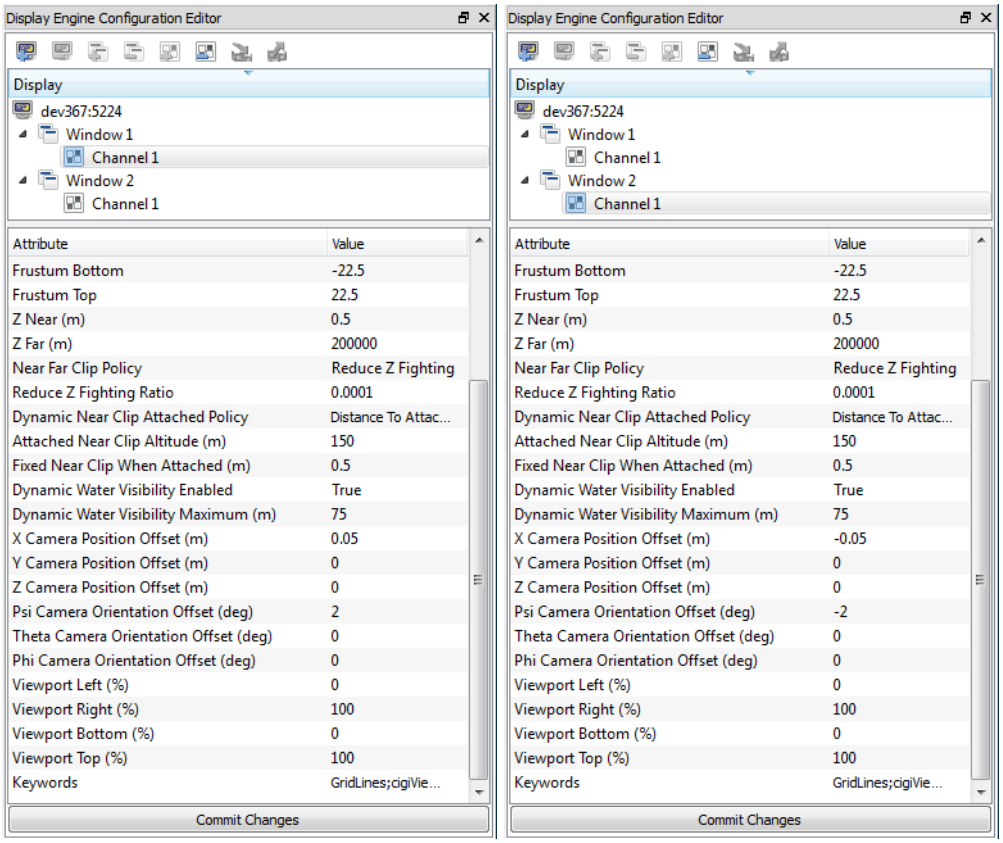
Environment Variable	Description	Default
OSG_STEREO	Turn stereo on.	ON
OSG_STEREO_MODE	Use anaglyphic stereo when in stereo.	ANAGLYPHIC
OSG_SCREEN_DISTANCE	Specifies the distance, in meters, that the viewer is from the screen.	0.50
OSG_SCREEN_HEIGHT	Specifies the height of the image, in meters, on the screen.	0.26
OSG_SCREEN_WIDTH	Specifies the width of the image, in meters, on the screen.	0.325
OSG_EYE_SEPARATION	Specifies the eye separation (interocular distance).	0.06

For more information, pleases see the OpenSceneGraph documentation at http://www.openscenegraph.org/index.php/documentation/guides/user-guides/62-stereo-settings.

6.8.2. Configuring Polarized Stereo

You can configure polarized stereo by taking advantage of VR-Vantage's ability to produce multichannel displays. You configure one channel per eye. Each channel renders with a camera that is at a slight offset from the other channel's camera to represent the spacing between the viewer's eyes.

Varying the eye spacing and focal length can produce different effects, and the best values may depend on what the viewer is looking at. Figure 6-19 shows an example configuration.



Left channel Right channel

Figure 6-19. Polarized stereo channel configurations

To compute the correct channel configuration values:

1. Pick a focal length, `L`. Objects at this distance will appear to be in the plane of the display. Objects that are further away will appear behind the display, and objects closer will appear in front of it.
2. Pick an eye spacing, `E`, to represent the distance between the viewer's eyes. Human eyes are typically about 0.063m apart[1].
3. Set the X Camera Position values to `-E/2` meters for the left channel and `E/2` meters for the right channel.
4. Set the Psi Camera Orientation values to `-arctan(E/2)/L` degrees for the left channel and `arctan(E/2)/L` degrees for the right channel.

6.9. Taking Screen Captures

You can take screen captures of the VR-Vantage window. By default, screen captures are saved to *./userData/screenshots*. You can change the default location. You can change the key used for screen captures. For details about changing key mappings, please see Chapter 46, *Creating and Editing Key Mappings*.

➤ To take a screen capture, press **F5**.

6.9.1. Specifying the Location of Screen Captures

To specify the location where screen captures are saved:

1. Choose **Settings** → **Application**. The Application Settings dialog box opens.
2. Select the General Application Settings page (Figure 6-20).

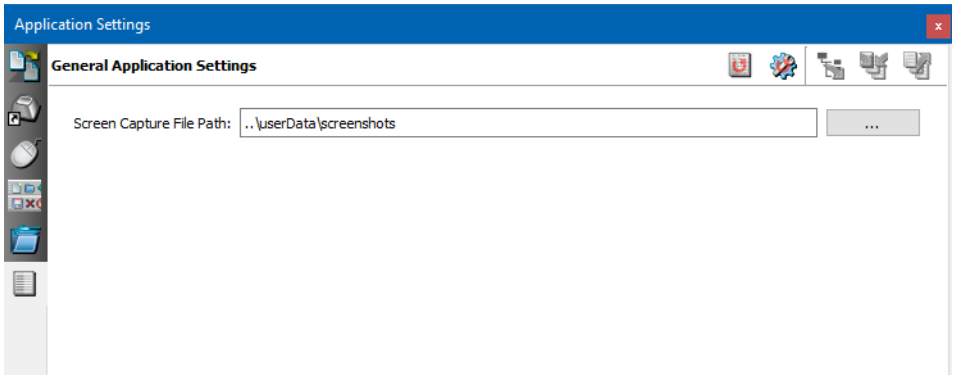


Figure 6-20. General Application Settings page

3. In the Screen Capture File Path box, specify the location where you want to save screen captures.

1. ""Variation and extrema of human interpupillary distance", N. A. Dodgson, Proc SPIE 5291. Select publications link and scroll to 2004 at http://www.cl.cam.ac.uk/~nad10/pubs/EI5291A-05.pdf.

6.10. Display Issues on Linux

VR-Vantage does not support two separate X screens. The reason is that when you create two separate X screens and try to create two OpenGL windows, they may not share OpenGL contexts. A number of VR-Vantage components require sharing contexts for our rendering to work. These include: SilverLining, SpeedTree, and our 2D overlays (possibly also GL Studio).

The work-arounds to support two screens include:

- Use the Nvidia driver's TwinView mode. This behavior is identical to the Window multi-monitor behavior and VR-Vantage works correctly with this setting.
- Use Xinerama, which is an X extension that behaves similar to TwinView.
- Run VR-Vantage on one X Screen and run a Remote Display Engine on the other. This will require two installs to ensure that the *./appData* directories are not compromised.



7. Display Engine Tutorials

This chapter has a tutorial that shows how to connect to a remote display engine, edit display engine configurations, and apply display engine configurations to display engines.

Saving and Loading Display Engine Configurations 7-2
Connect a VR-Vantage Display Engine 7-2
Save Display Engine Configurations 7-5
Load the Display Engine Configurations in the GUI 7-7
Load Display Engine Configurations from the Command Line 7-8

7.1. Saving and Loading Display Engine Configurations

This tutorial demonstrates how to:

- Connect a VR-Vantage application to a VR-Vantage Display Engine.
- Change a display engine configuration.
- Save display engine configurations.
- Load display engine configurations in the Display Engine Configuration Editor Panel.
- Load multiple display engine configurations from the command line.

7.1.1. Connect a VR-Vantage Display Engine



This tutorial assumes that you have two computers, one for your VR-Vantage application, and one for the remote display engine. You could run this tutorial on one computer if you have two monitors.

To create the display engine configurations:

1. On computer 1, start VR-Vantage.
2. Load *MAK Earth (online).mtf*.
3. Choose **View** → **Display Engine Configuration Editor Panel**. The Display Engine Configuration Editor Panel opens. It shows one window and one channel for the VR-Vantage window on the local computer (Figure 7-1).

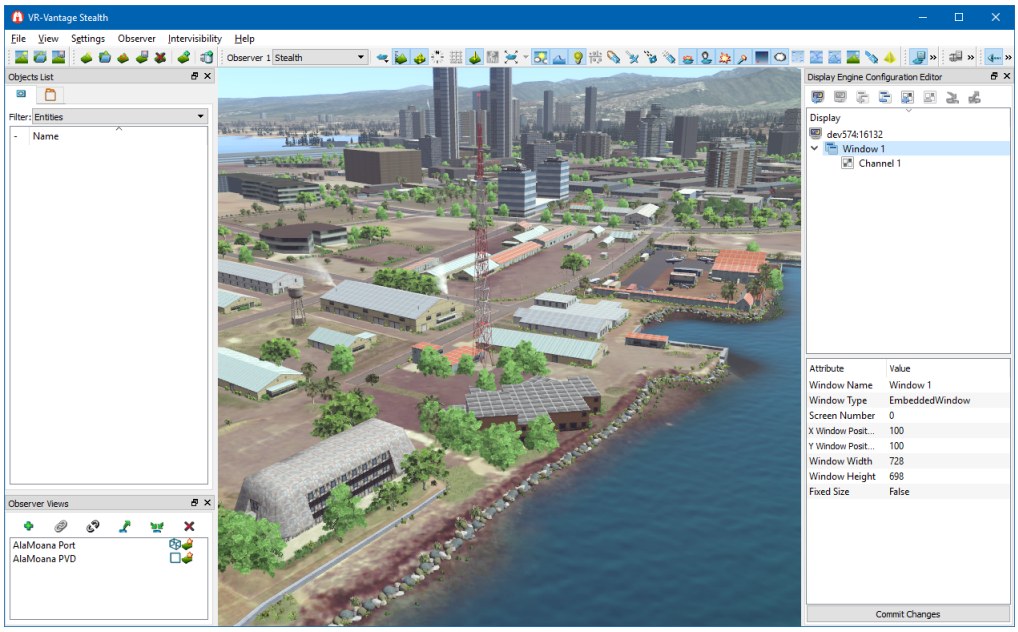


Figure 7-1. Display Engine Configuration Editor Panel and default window

4. On computer 2, on Windows, on the Start menu, choose **MAK Technologies** → **VR-Vantage 2.3** → **VR-Vantage Display Engine**.

   On Linux, run:

   `./bin64/displayEngine.`*`extension`*

   where *`extension`* is the appropriate script extension.

   A VR-Vantage display engine opens. It does not show a default window.

5. On computer 1, in the Display Engine Configuration Editor Panel, click the Connect a Display Engine button ( ). The Connect to Display Engine dialog box opens.

6. In the Display Engine Hostname box, replace "localhost" with the name of computer 2. (If you are using one computer with two monitors, leave the text as localhost.)

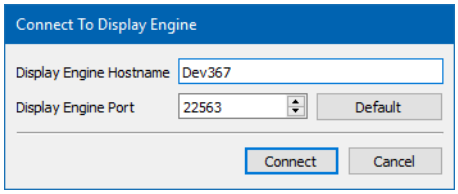


Figure 7-2. Connect to Display Engine dialog box

7. Click Connect. An entry for the remote display engine is added to the Display Engine Configuration Editor Panel (Figure 7-3). The remote display engine now displays the MAK Earth terrain with the same view as the VR-Vantage application.

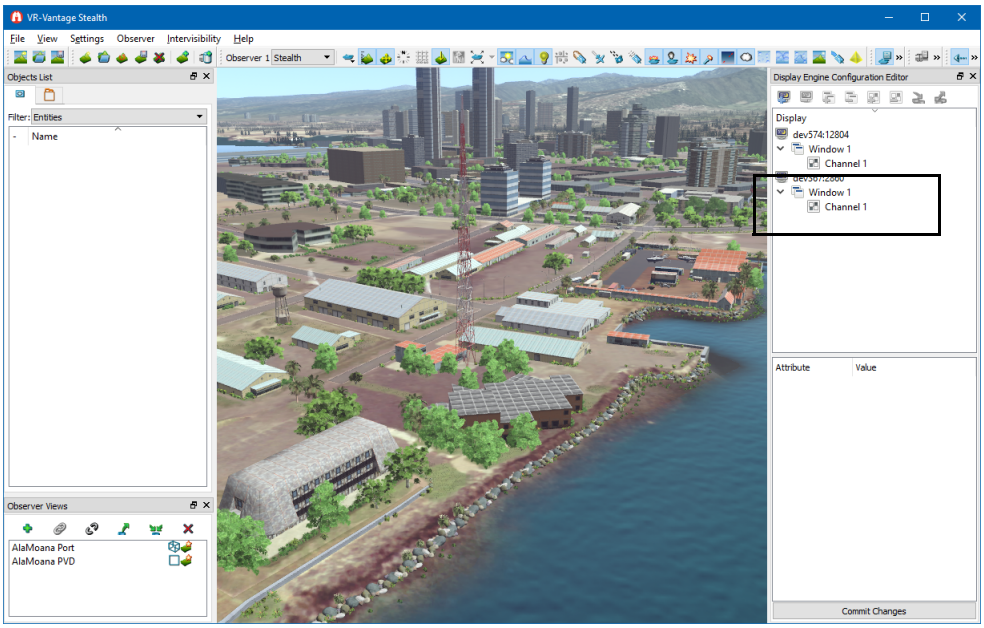


Figure 7-3. Remote display engine connected

7.1.2. Save Display Engine Configurations

By default, the views in the VR-Vantage window and the remote display engine window look exactly the same. So we will change the view in one of them so that when we load the display engine configurations later on, we can tell that each is different.

To edit the display engine configurations and save them:

1. Select Channel 1 of the remote display engine. Its attributes are displayed.
2. Change the value for the Viewport Right and Viewport Top attributes to 50 (Figure 7-4).

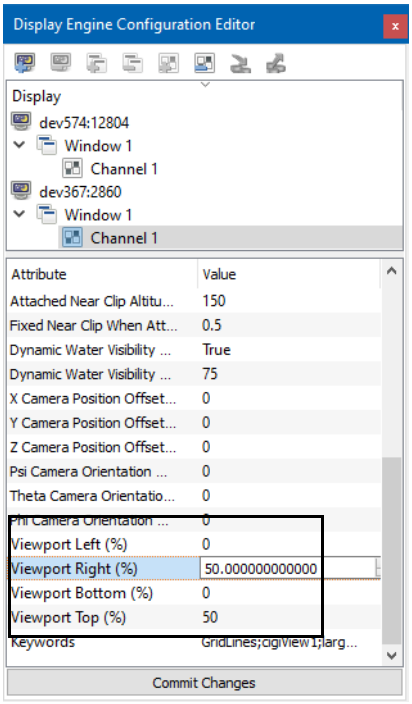


Figure 7-4. Channel with changed viewports

3. Click Commit Changes. The view of the terrain is now in the lower left quadrant of the display engine (Figure 7-5). (The full screen view of the terrain may also be visible until such time as the display engine refreshes.)



Figure 7-5. Remote display engine view after changing configuration

4. Select the display engine on the VR-Vantage application (in this example, dev367).
5. Click the Save Display Engine Configuration button ( ). A save dialog box opens.
6. Call this configuration *mainwindow.dcx*.
7. Click Save.
8. Select the display engine on the remote display engine (in this example, Dev25).
9. Click the Save Display Configuration button. A save dialog box opens.
10. Call this configuration *remotedisplay.dcx*.
11. Click Save.

7.1.3. Load the Display Engine Configurations in the GUI

We now have two visually distinct display engine configurations. In this part of the tutorial, we simply load them into the two display engines to see that loading a display engine configuration changes the view in the window. This part of the tutorial also demonstrates that display engine configurations are not unique to a particular display engine and are not tied to one either. Any particular configuration can be loaded into any display engine.

To load the display engine configurations in the GUI:

1. Select the display engine on the VR-Vantage application.
2. Click the Load Display Engine Configuration Editor button (). The Load Display Configuration dialog box opens.
3. Select *remotedisplay.dcx*.
4. Click Open. The window on the VR-Vantage application now shows the terrain in the lower left quadrant. (You may need to press **`Space Bar`** to reset the observer.)
5. Reload the *mainwindow.dcx* configuration in the main window.
6. Try loading the two different configurations in the remote display engine.

7.1.4. Load Display Engine Configurations from the Command Line

As you have seen in the previous part of this tutorial, you can load display engine configurations for individual display engines through the GUI. However, you cannot load more than one at a time. If you have a complex set of display engines that you need to configure, loading them one at a time can be tedious. To load multiple display engine configurations at one time, you must start VR-Vantage from the command line.

You load multiple display engine configurations from the command line with the `--dispSetting` command line argument and the `-W` or `--withDisplayEngine` command-line argument. The syntax for Windows is:

```
vrvApplication.exe --dispSetting masterdisplayConfigFile
  -W port@netAddress;remotedisplayConfigFile1 ....
  -W port@netAddress;remotedisplayConfigFileN
```

The syntax for Linux is:

```
vrvApplication.exe --dispSetting masterdisplayConfigFile
  -W port@netAddress:remotedisplayConfigFile1 ....
  -W port@netAddress:remotedisplayConfigFileN
```

In this tutorial, we use the default port (22563). The netaddress for the remote display engine is the computer name (Dev25). The display engine configuration files are *mainwindow.dcx* and *remotedisplay.dcx*.

So the command line to load the two display engine configurations that we created would be:

```
vrvIG.exe --dispSetting ../appData/settings/vantageIG/-
  mainwindow.dcx -W 22563@Dev25;
  ../appData/-settings/vantageIG/remotedisplay.dcx
```

To load the tutorial configurations from the command line:

1. Open a command window.
2. Change directory to *./bin64*.
3. On Windows, enter the command:

   ```
   vrvIG.exe --dispSetting ../appData/settings/-
     vantageIG/mainwindow.dcx -W 22563@Dev25;
     ../appData/settings/vantageIG/remotedisplay.dcx
   ```

   On Linux, enter the command:

   ```
   vrvIG.exe --dispSetting ../appData/settings/-
     vantageIG/mainwindow.dcx -W 22563@Dev25:
     ../appData/settings/vantageIG/remotedisplay.dcx
   ```

   VR-Vantage starts. The display engine configurations are visible in the main window and the remote display engine window.

4. Choose **View** → **Display Engine Configuration Editor Panel**. In the Display Engine Configuration Editor Panel, confirm that VR-Vantage is connected to the remote display engine.



8. Using Streaming Video

VR-Vantage can send the view in the display window to a video stream. It supports several different video formats.

Streaming Video 8-2
Starting a Video Stream 8-3
Setting Up Auto Start 8-4
Configuring Video Streams 8-4
Configuring the Number of Capture Buffers 8-8
Viewing Streaming Video 8-9
Viewing Streaming Video Using VLC 8-10
Recording the VR-Vantage Display 8-14

8.1. Streaming Video

VR-Vantage can capture the view in the display windows and send it as a video stream. Streaming video can be used for embedded training to stimulate devices that expect live video with simulated video. Video streams can be encoded and transported in several different ways using common open standards. VR-Vantage can encode metadata about the video image into the video stream following MISB (Motion Imagery Standards Board) metadata standards for UAV systems.

You can also stream video to a file. In other words, you can record the VR-Vantage display. To do so, use the GStreamer MPEG2 - File Out video stream, as described in "Configuring Video Streams," on page 8-4.

! Streaming video is not supported on Linux.

VR-Vantage processes captured video with image tasks. Some image tasks transform the image and some transport the image. You can mix together multiple image tasks to accomplish your goal. You can replace or add your own image tasks through the API. The image tasks provided are:

- Shared Memory. VR-Vantage compresses the video and places it in shared memory. We provide an example application that shows how to read the shared memory data for your application.
- TCP. VR-Vantage acts as a server and connected clients are able to connect over TCP and get raw or compressed images through the connection.
- Web socket. VR-Vantage acts as a server and web browsers can connect to it and display the video stream.
- GStreamer. GStreamer is a multimedia library that uses configuration files to build a pipeline. Pipelines allow for different encoding, multiplexing, and transport options. This is how VR-Vantage encodes video with MPEG-2 and H.264 encoding schemes and transports them over a network with embedded metadata. (GStreamer is a complex SDK. If our implementation does not meet your needs for video streaming, please contact MAK support. We will work with you to help you achieve your goals.)
- Image Compression. VR-Vantage compresses the video frame using JPG or PNG compression. This is typically used with video sinks that use image frames such as the WebSocket, shared memory, and TCP video sinks.
- Transform. VR-Vantage transforms the video frame by scaling, flipping, or converting the bit depth of the image. This is typically used with video sinks that use image frames such as the WebSocket, shared memory, and TCP video sinks. GStreamer can also transform the image and you may get better results using GStreamer alone than using a Transform operation and a GStreamer operation.

- VR-Vantage does not include metadata plug-ins. They are distributed as bonus material. For details, please see your MAK sales representative.
- Streaming video can reduce performance.

8.2. Starting a Video Stream

You can start a video stream manually or specify that a video stream start automatically when VR-Vantage starts up.

To start a video stream:

1. Choose **Settings** → **Video Streams**. The Video Streams dialog box opens. Select the Video Streams page (Figure 8-1).

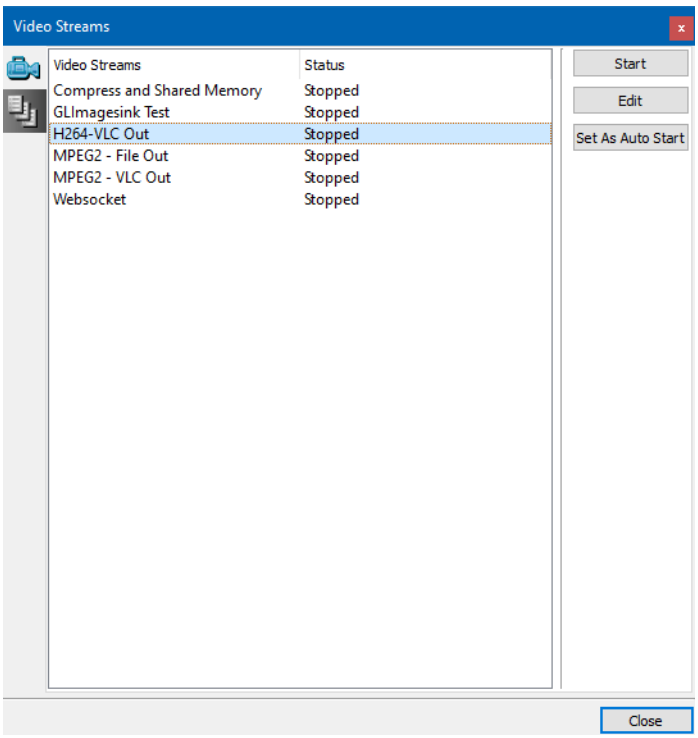


Figure 8-1. Video Streams page

2. Select the video stream that you want to start. If it is available, the Unavailable button changes to Start.
3. Click Start.
4. View the video in your viewing application.
5. Click Stop to stop the video stream.

8.2.1. Setting Up Auto Start

You can specify that a video stream start when VR-Vantage starts up.

To specify auto start:

1. Choose **Settings** → **Video Streams**. The Video Streams dialog box opens. Select the Video Streams page (Figure 8-1).
2. Select the video stream that you want to start automatically.
3. Click Set As Auto Start. VR-Vantage displays a star next to the video stream name.

8.3. Configuring Video Streams

You can configure the different streaming video streams to work with your playback environment. A video stream configuration has:

- A set of parameters that specify how VR-Vantage should capture the video.
- A pipeline. VR-Vantage sends video through a pipeline. The pipeline processes the video based on image tasks that you specify. You can specify one or more image tasks and can set the order in which they are executed. The video stream configurations included with VR-Vantage are examples of how to configure different types of video streams.

To configure a video stream:

1. Choose **Settings** → **Video Streams**. The Video Streams dialog box opens. Select the Video Streams Configuration page (Figure 8-2).

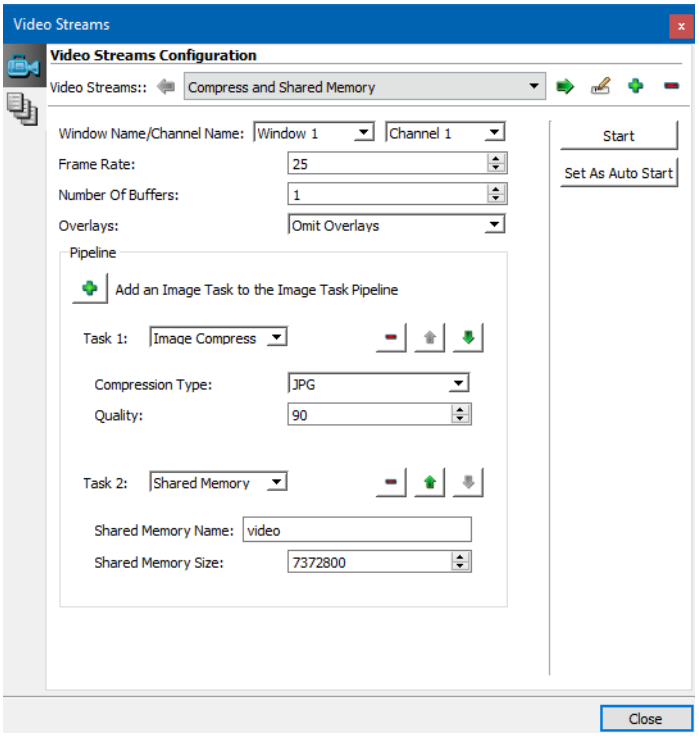


Figure 8-2. Video stream configuration page

2. In the Video Streams list, select the video stream that you want to configure.

3. Specify values for the capture parameters. Table 8-1 describes the parameters

Table 8-1: Video stream configuration capture parameters

Parameter	Description
Window Name	The window from which to stream video.
Channel Name	The channel, in the window specified by Window Name, from which to stream video.
Frame Rate	The frame rate at which to stream video.
Number of Buffers	The number of buffers affects how well VR-Vantage captures video from the rendering process. Depending on your hardware, more buffers may help rendering and capture performance at the cost of overall latency and video memory usage.
Overlays	Omit or include 2D overlays, including 2D icons, the compass, and tactical graphics.

4. To add a task, in the Pipeline group box, click the Add button (+). A task is added to the group box.
5. For the new task or any existing tasks, select an image task in the Task drop down list.
6. Select values for the parameters that are appropriate to the task. The supported tasks and their parameters are as follows:
   - TCP. Send the video over TCP.
     - Port. The port to which the video stream should be sent.
   - GStreamer.
     - Pipeline Configuration. This is a complex string that varies depending on what you want done with the video. It defines the GStreamer nodes that the video frames will be processed through. You can resize, transform, compress, and multiplex the video. Then you can send the results to disk as a file or to the network as a stream. VR-Vantage has examples that show how to encode video, multiplex in metadata about the video stream, save to file. and stream to a network. For information about GStreamer and how to build a configuration description, please go to http://gstreamer.freedesktop.org/.
   - Transform.
     - Flip Image. Optionally, flip the image.
     - Scale Image. Optionally, scale the image to the specified X, Y values.
     - Convert Bit Depth. Optionally, convert the image to the specified bit depth.

- Shared Memory.
  - Shared Memory Name. The name of the shared memory space.
  - Shared Memory Size. The amount of shared memory. This must be large enough to hold at least three frames of data as specified in the capture parameters.
- Image Compress.
  - Compression Type. Select an option from the list.
  - Quality. Specifies the level of compression as a percentage. The higher the percentage, the less the compression and therefore, the better the quality.
- Web Socket.
  - Connection Port. The port on which VR-Vantage will listen for a browser connection.

7. Optionally, add additional image tasks.
8. Optionally, change the order of the tasks.

8.3.1. Configuring GStreamer H.264 VLC Out

This section explains how to set up VR-Vantage to stream to H.264 VLC Out. Assume that your PC's IP address is 123.456.7.89.

To configure H.264 VLC Out:

1. Use the following GStreamer pipeline:

   ```
   appsrc name=myVideoAppSrc ! queue ! videoflip method=5 !
   ffmpegcolorspace ! x264enc byte-stream=true bitrate=1000
   threads=4 tune=zerolatency ! queue ! mpegtsmux ! udpsink
   host=123.456.7.255 auto-multicast=true port=5000
   ```

   Note that the host is based on your PCs IP address.
2. Create a display engine configuration in which the window size dimensions are multiples of 16, for example, 1280x720 and set Fixed Size to true.
3. If you are using Windows 10, you may need to disable the firewall or write new rules to send or receive data on the ports you are using. This applies to the receiving PC as well.
4. On the receiving PC, set up VLC as described in "Viewing Streaming Video Using VLC," on page 8-10.

8.3.2. Configuring the Number of Capture Buffers

When a frame is captured from the frame buffer in OpenGL, it is copied to an offscreen buffer on the GPU. Then the buffer is copied across the bus from the GPU to the CPU, where it can be pushed into the encoding system. With some video cards, this process can cause all rendering to stop until all the memory copying is completed between buffers and between GPU and system memory. Moving data across the bus can be a slow process. This is called stalling out the rendering pipeline and usually results in a significant drop in rendering performance.

To see if the rendering pipeline is stalling:

1. Press **F2** to get the on screen frames-per-second (FPS) display.
2. Start streaming video.
3. See if the FPS value drops.

To see if the number of buffers affects performance:

1. Start VR-Vantage Stealth from the command line with the `--showConsole` option.
2. Watch the timing values that are displayed in the console. If the readback and copy times are in the 10-15ms range, then the rendering pipeline is probably stalling. If they are in the 0-2 ms time range, then they are not adversely affecting performance.

Using one buffer instead of multiple buffers also affects the latency in the video stream. If a system can render, capture, and read back the data using only one buffer, then the data in the buffer is from the current frame and has no additional latency. If your system needs to use more than one buffer (two is typical) to prevent stalling the rendering pipeline, the buffer that is copied from GPU memory to CPU memory is from the previous capture and has one capture frame of latency. If frame capture is occurring at 25fps, then this means the frame that is pushed into the encoding process already has 40ms of latency in it.

8.4. Viewing Streaming Video

The method used to view a video stream depends on its configuration. This section describes viewing the video that results from the default configurations included with VR-Vantage.

- Compress and Shared Memory. This video stream configuration is an example of a pipeline that compresses and moves frames into a shared memory segment. To use this configuration you must provide an application that reads the frames from shared memory and displays them using whatever method it likes. The exampleReadVideoShm example shows how to access frames from shared memory.
- GLImagesink Test. This video stream configuration uses a GStreamer pipeline task to stream video directly to an OpenGL viewer. If this configuration is started, a new OpenGL window opens containing the video stream.



The GStreamer pipeline task can only be used once per application run. To start another video stream using GStreamer, you must stop VR-Vantage and start it again.

- MPEG2 - VLC Out. The MPEG - VLC Out video stream configuration uses a GStreamer pipeline task to compress video using MPEG-2 and then send the video to a UDP port. The port used is part of the pipeline configuration string (and is a parameter to the udpsink GStreamer element). There are commercial and freeware video viewers that can connect to and display MPEG video from a UDP stream. One such viewer is VLC. VLC is a commonly used application to show received network video streams. You can download VLC from http://www.videolan.org/vlc/index.html. For more information, please see "Viewing Streaming Video Using VLC," on page 8-10.
- MPEG2 - File Out. This video stream configuration uses a GStreamer pipeline task to output MPEG-2 video to a file. The file that the video is written to is specified in the pipeline configuration, as a parameter to the filesink GStreamer element. If no path is included in the filename, the file is written to *./bin64*.
- WebSocket. The WebSocket video stream configuration streams capture frames to a web browser. When the video stream is started, it listens on the Connection Port specified for connections from a web browser. Specify the URL as `hostname:port` (use localhost if connecting from the same machine). Typical usage is to put an Image Compress task element prior to this task element to send compressed images to the connected web clients. For more information, please see "Streaming Video to a Web Browser," on page 8-13.



- Some older versions of browsers may not support WebSockets.
- If you stream using WebSockets on Windows 10, you might need to disable the Windows firewall.

8.4.1. Viewing Streaming Video Using VLC

For information about reducing latency in VLC, please see "Reducing Latency in VLC," on page 8-10.

To view video in VLC streamed from VR-Vantage:

1. Start the MPEG - VLC Out stream.
2. Start VLC.
3. Choose **Media** → **Open Network Stream**.
4. For the network URL, put "udp://@:5000" (if the parameter to udpsink is port=5000).

Reducing Latency in VLC

The default setting for viewing network streams in VLC includes a lot of buffering to make sure the video plays smoothly. This introduces a lot of latency on the receiving side. This section explains how to reduce that latency. It assumes you are using VLC 2.2.6 Umbrella.



Reducing latency may lead to jitter. However, when working on a LAN, MAK engineers have rarely had problems.

To reduce latency when using VLC:

1. Start VLC.
2. Choose **Tools** → **Preferences**. The Preferences dialog box opens.
3. Select Input & Codecs (Figure 8-3).
4. In the Codecs group box, in the Hardware-accelerated Decoding list, select Automatic.
5. In the x264 Preset and Tuning Selection list, select ultrafast.
6. In the Network group box, in the Default Caching Policy list, select Lowest latency.
7. Click Save.

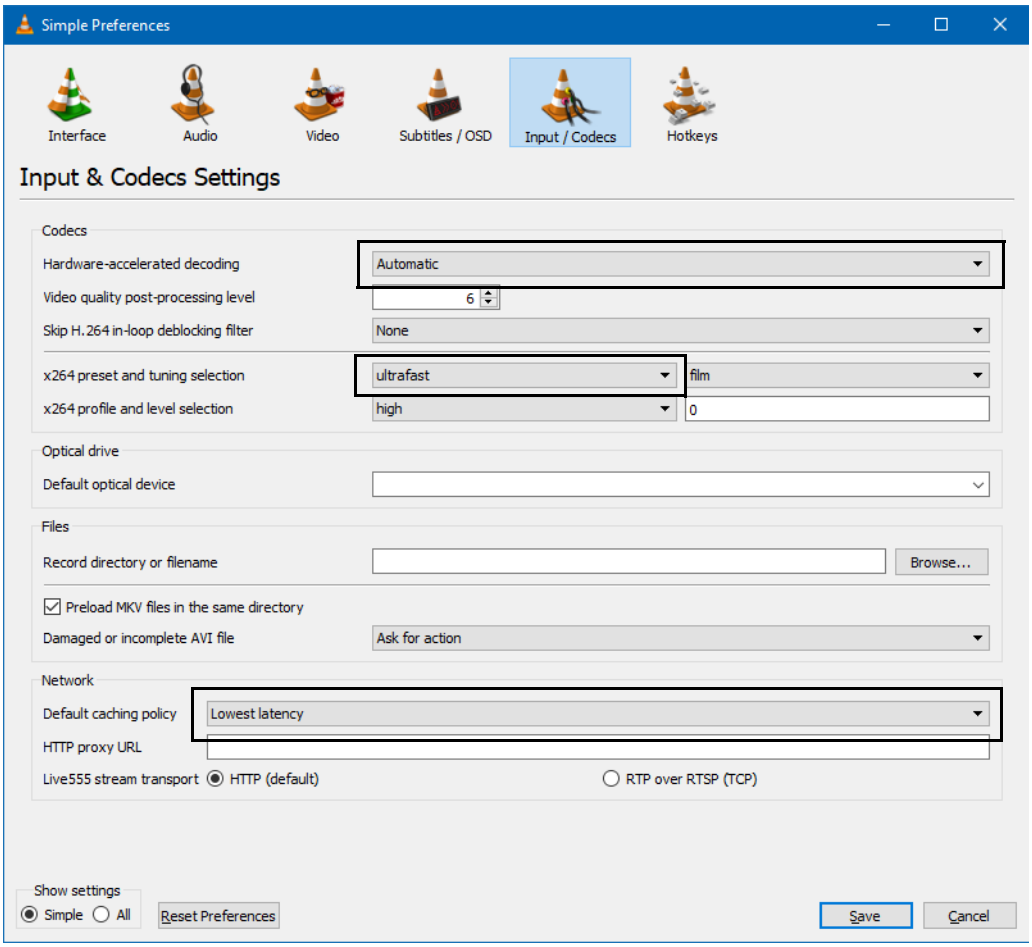


Figure 8-3. VLC Preferences, Inputs & Codecs

8. Choose **Media** → **Open Network Stream**. The Open Media dialog box opens.
9. Select the Network tab (Figure 8-4).
10. Enter the URL as udp://@:5000 (or whatever port you have your system set up for).
11. Select the Show More Options check box. The dialog box expands with more options.
12. Set Caching to 1ms. (Do not set caching to zero. VLC will crash.)

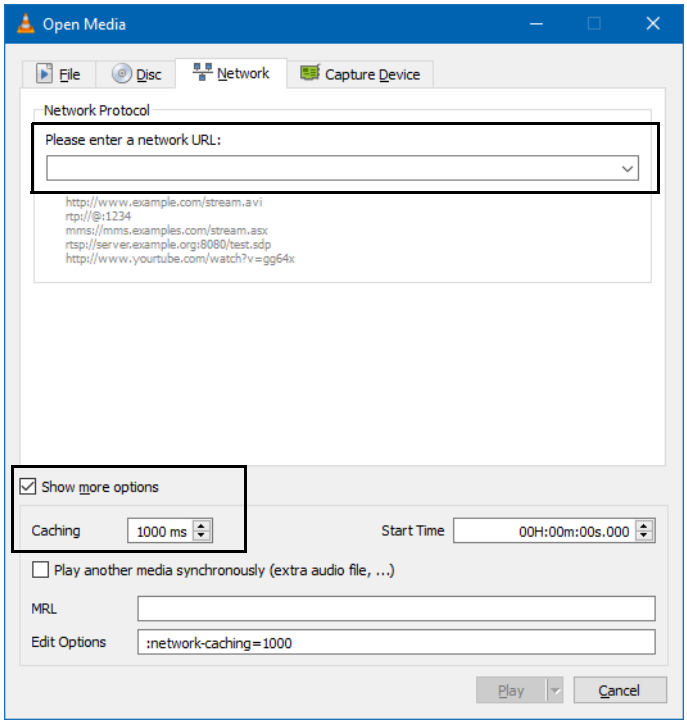


Figure 8-4. Open Media dialog box, Network tab

13. Click Play.

8.4.2. Streaming Video to a Web Browser

To stream video to a web browser, you just need to know the hostname of the computer on which you are running the browser. If you run a browser on the same machine as you are using to run VR-Vantage, you can specify the hostname as localhost (Figure 8-5).

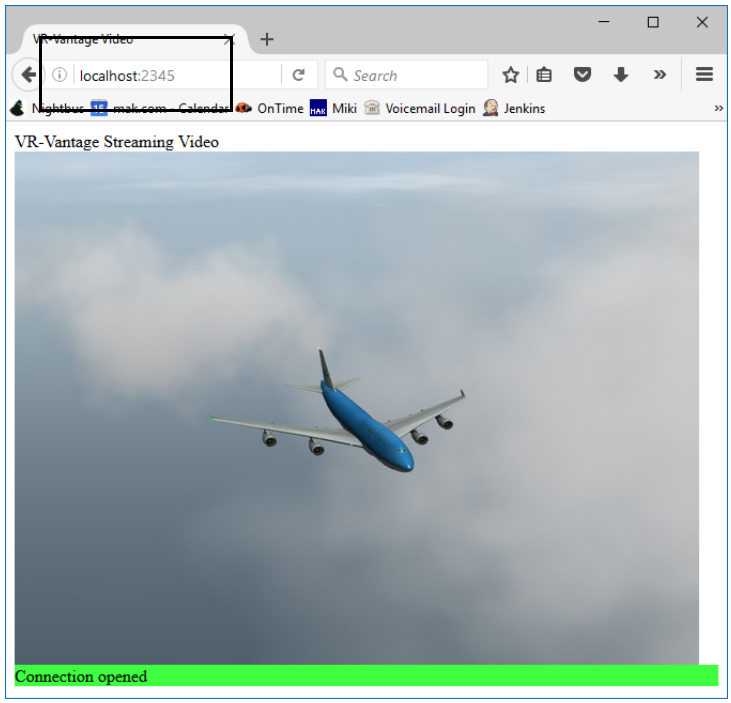


Figure 8-5. Streaming video to a browser

To stream video to your web browser:

1. Start VR-Vantage.
2. Choose the **Settings** → **Video Streams**. The Video Streams dialog box opens.
3. Select the Video Streams Configuration page (Figure 8-2).
4. In the Video Streams list, select Websocket (Figure 8-6).

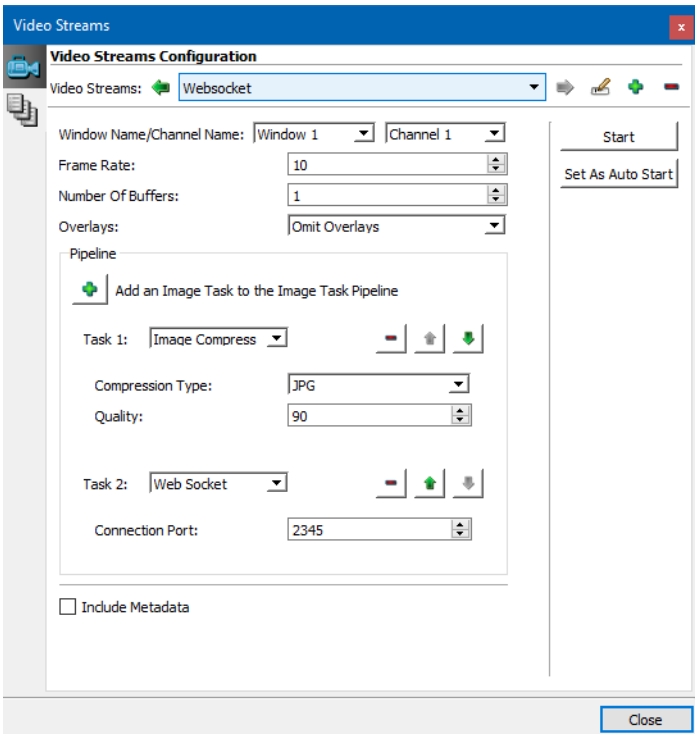


Figure 8-6. Websocket configuration

5. In the Pipeline group box, accept the default value for the Connection Port (2345) or specify a different value.
6. On the Video Streams Configuration page, click Start.
7. If you are not already connected to an exercise, connect VR-Vantage to the exercise you want to view.
8. In your browser type the URL `localhost:2345`. (If you are not connecting to the same machine, use hostname:2345. If you specified a different port, use the port that you specified.) You will see the video stream.

8.5. Recording the VR-Vantage Display

VR-Vantage users often want to record the VR-Vantage display. You can do this with commercial video capture tools. You can also send a video stream to disk. The MPEG2 - File Out video stream configuration provided with VR-Vantage sends the output to a file. The configuration is described in "Viewing Streaming Video," on page 8-9.



9. Using Head Mounted Displays for Virtual Reality

This chapter explains how to use a head mounted display (HMD) such as the Oculus Rift, HTC Vive, or Microsoft Mixed Reality Headset with VR-Vantage.

Introduction to Virtual Reality 9-2
Requirements 9-2
Resetting the Tracking Sensor 9-2
Running VR-Vantage with an HMD 9-3
Enabling and Disabling Virtual Reality Rendering 9-4
Measuring Performance 9-5
Configuration Options for Virtual Reality 9-6

9.1. Introduction to Virtual Reality

VR-Vantage is integrated with several virtual reality head mounted displays (HMDs). VR-Vantage can render stereo views, which are presented inside the HMD. The head position and orientation tracking of the HMD are reflected in the observer as the wearer of the HMD moves their head.

VR-Vantage uses the OpenVR library to integrate with virtual reality systems. Any system that is capable of using OpenVR should work with VR-Vantage. VR-Vantage is tested with the following systems:

- Oculus Rift.
- HTC Vive.
- Windows Mixed Reality headsets.

9.1.1. Requirements

To use an HMD, you need the following software and VR-Vantage setup:

- SteamVR, which is installed through the Steam digital distribution platform (http://store.steampowered.com/).
- If you are using an Oculus Rift, the Oculus Runtime must be installed.
- If you are using Microsoft Mixed Reality Devices, the "Windows Mixed Reality for SteamVR" add-on must be installed through Steam.

9.1.2. Resetting the Tracking Sensor

For user comfort, you can reset the origin of the positional tracking performed by the HMD. This allows you to restart the head tracking once you have put on the HMD and assumed a comfortable position. You can reset the sensor at any time by pressing the **`F8`** key. You can change the keyboard assignment by changing the "Reset Virtual Reality HMD" action in the VR-Vantage Key Mapping Editor.

9.2. Running VR-Vantage with an HMD

To use an HMD, you must load the virtual reality plug-in when you start VR-Vantage. VR-Vantage includes a batch file with the startup commands to load the plugins. This batch file is also included in the start menu. If you want to enable Virtual Reality every time you start VR-Vantage, copy *vrvVirtualRealityPlugin.dll* and *vrvVirtualRealityQtPlugin.dll* from *./bin64* to *,/plugins64/release*.

To use an HMD with VR-Vantage:

1. Start SteamVR. SteamVR Home must be visible in the HMD.
2. Start VR-Vantage Stealth or IG using the **VR-Vantage with Virtual Reality** shortcut on the Start menu.
3. Load a terrain.
4. Connect to a simulation.
5. Position the observer using the mouse and keyboard or attach to an entity in Mimic mode.
6. Disable cockpit displays (Display Settings dialog box, Cockpit Displays page).
7. Position the observer to the location of the eyes of the entity or operator of the vehicle.
8. Put on the HMD.
9. When you are comfortable, reset the tracking sensor as described in "Resetting the Tracking Sensor," on page 9-2.

9.2.1. Enabling and Disabling Virtual Reality Rendering

By default, when you start VR-Vantage with Virtual Reality, rendering to the HMD is enabled at startup. You can enable and disable rendering to the HMD at runtime.

To enable or disable virtual reality rendering:

1. Choose **Settings** → **Application**. The Application Settings dialog box opens.
2. Select the Virtual Reality Settings page (Figure 9-1).

> The Virtual Reality Settings page is available only if the virtual reality plug-in is loaded and an HMD is detected.

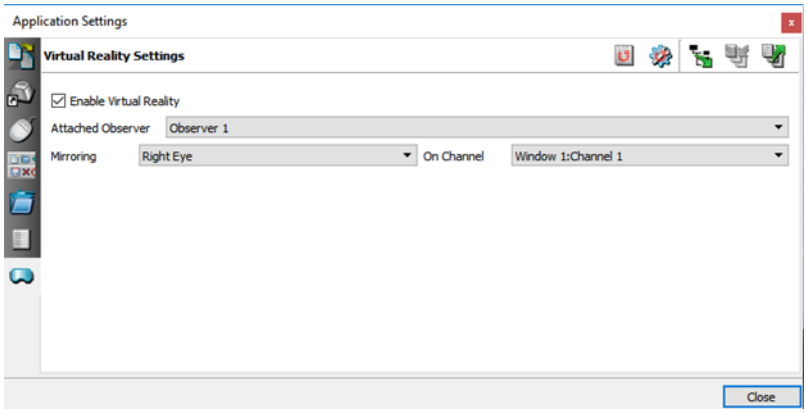


Figure 9-1. Virtual Reality Settings page

3. Select or clear the Enable Virtual Reality check box.

> You can also enable or disable virtual reality by choosing **Settings** → **Enable Virtual Reality**, or clicking the Virtual Reality button ( ) on the Display Settings toolbar.

9.3. Measuring Performance

Virtual reality requires a high frame rate for a comfortable experience. 90hz is the recommended frame rate. Lower frame rates can cause discomfort and nausea in some individuals. Using the frame rate statistics in VR-Vantage can help you tune VR-Vantage for optimal performance. You can also see frame rate statistics in the SteamVR Performance Settings dialog box.

Figure 9-2 illustrates SteamVR performance statistics.

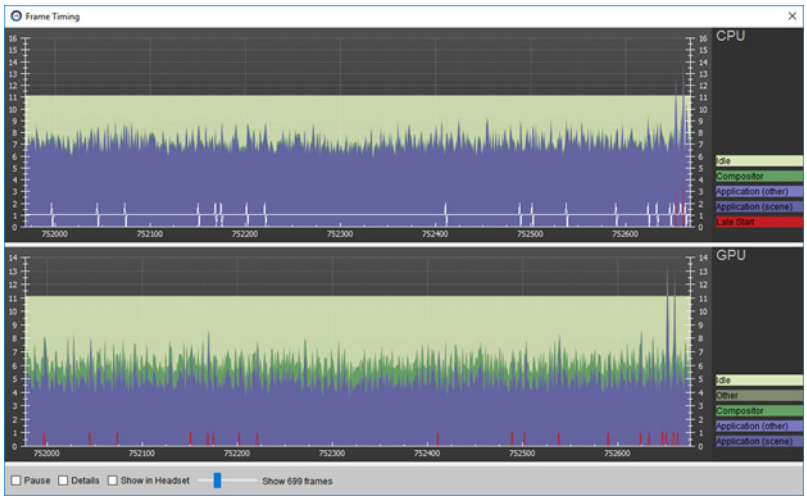


Figure 9-2. SteamVR performance statistics

9.4. Configuration Options for Virtual Reality

You can set the following configuration options for the virtual reality:

- Attached Observer. This is the observer that the HMD will attach to and render the perspective from. The default is to attach to Observer 1.
- Mirroring. This copies the image rendered for a given eye to the selected channel in the main display. The default is to render the right eye to the main channel. Mirroring disable the normal render pass of the attached channel to help with performance.



If mirroring is enabled and the HMD is attached to the observer that is driving a displayed channel, then the main view will be rendered from the HMD perspective. This will affect performance, because the scene will be rendered a third time.

To set virtual reality configuration settings:

1. Choose **Settings** → **Application**. The Application Settings dialog box opens.
2. Select the Virtual Reality Settings page (Figure 9-1).



The Virtual Reality Settings page is available only if the virtual reality plug-in is loaded and an HMD is detected.

3. Change the settings as desired.



10. Using Multiple Projectors with VR-Vantage

This chapter describes how to set up two PC's and two projectors to project a large, seamless, edge blended display of VR-Vantage.

Using Multiple Projectors 10-2
Hardware Requirements 10-2
Installing the Scalable Display Manager 10-2
Configuring the Master Computer 10-3
Running VR-Vantage with the Scalable Plug-in 10-10

10.1. Using Multiple Projectors

VR-Vantage can support displays made up from multiple projectors using the Scalable Display Manager (SDM) from Scalable Display Technologies (http://www.scalabledisplay.com). This software automates the process of combining and edge blending the projectors together into one seamless display. Each computer driving a projector from VR-Vantage is set up either as the master or as a Scalable Display Client (SDC).

This chapter covers the integration of SDM with VR-Vantage.

The VR-Vantage SDM plug-in is only supported on 64-bit Windows.

10.2. Hardware Requirements

The procedure in this chapter assumes you have two computers, two projectors and a camera. The Scalable Display Manager software requires a camera. It must be one of the following cameras:

- Logitech C910, C920.
- Canon Rebel XSi, T1i, T2i, T3, T3i.

Set up the two computers with graphics cards that will be synchronized together. One computer is the master; the other computer is the client.

Set up the camera with its correct drivers on the master computer.

Set up the projector displays so that they overlap at least 15%.

10.3. Installing the Scalable Display Manager

1. Download the Scalable Display Manager (SDM) and associated documentation. Make sure to also download the User Manual.
2. Run the Scalable Display client on the client machines. The client software will run in the background (no GUI will be seen).
3. On the master computer, right click the Scalable Display Manager installer and choose **run as admin**. The Scalable Display Manager is installed.
4. Request a license from Scalable Display Technologies.

10.3.1. Configuring the Master Computer

1. Open the Scalable Display Manager.
2. In the Components Setup Panel, select Display Clients. Make sure you have the master and display clients in the Assigned Display Clients list (Figure 10-1).

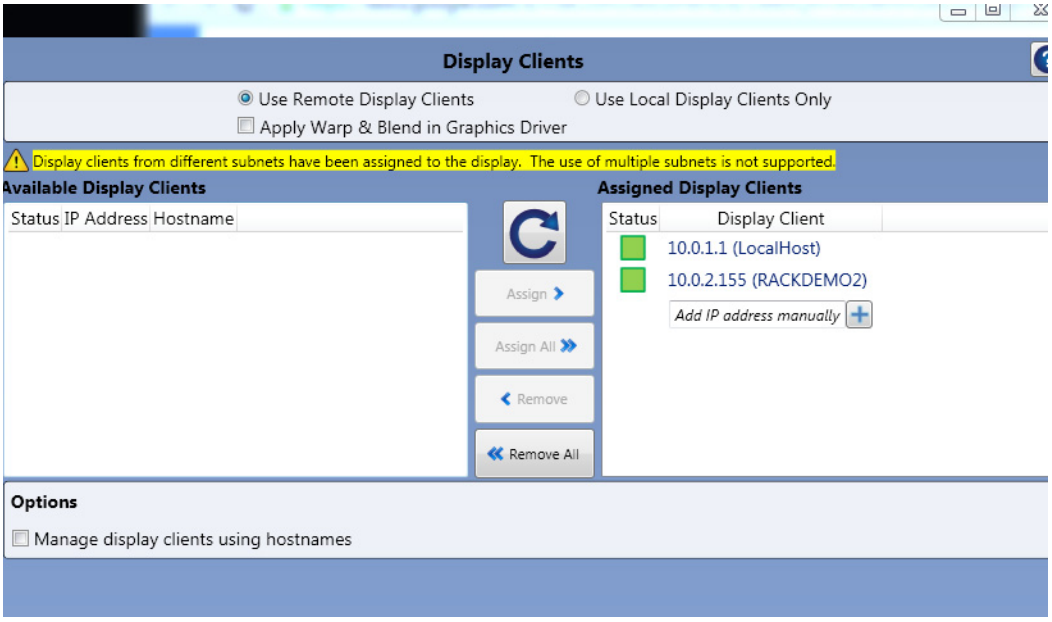


Figure 10-1. Display Clients list

3. In the Components Setup Panel, select Projectors.
4. Specify the arrangement of the projectors (Figure 10-2).

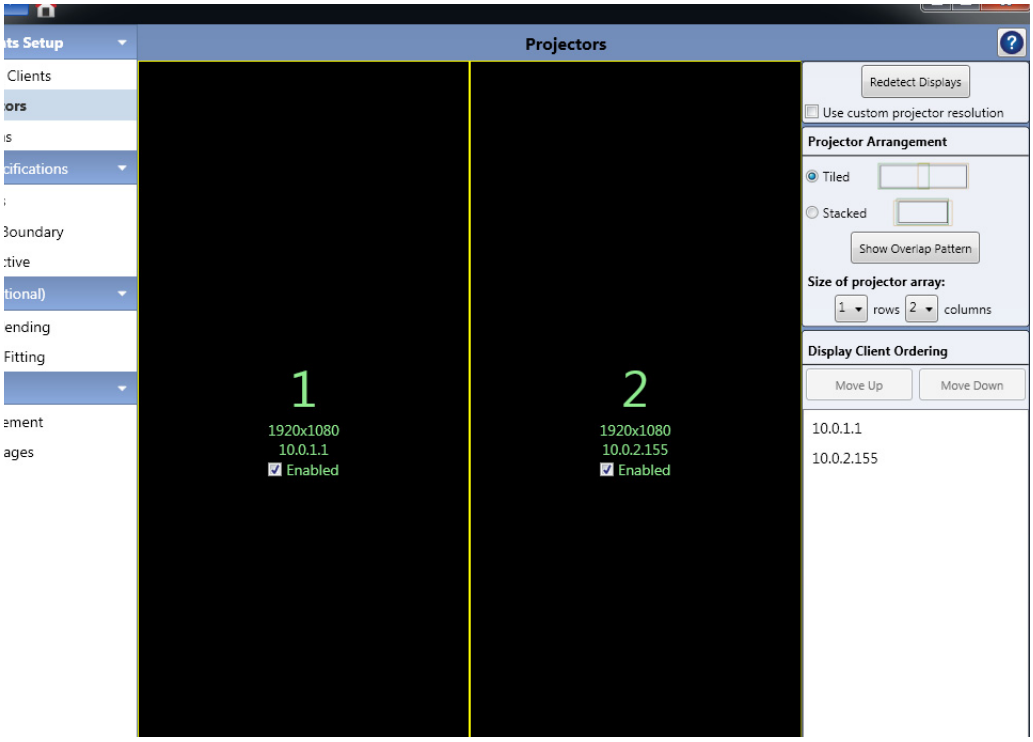


Figure 10-2. Projectors

5. In the Components Setup Panel, select Cameras (Figure 10-3).

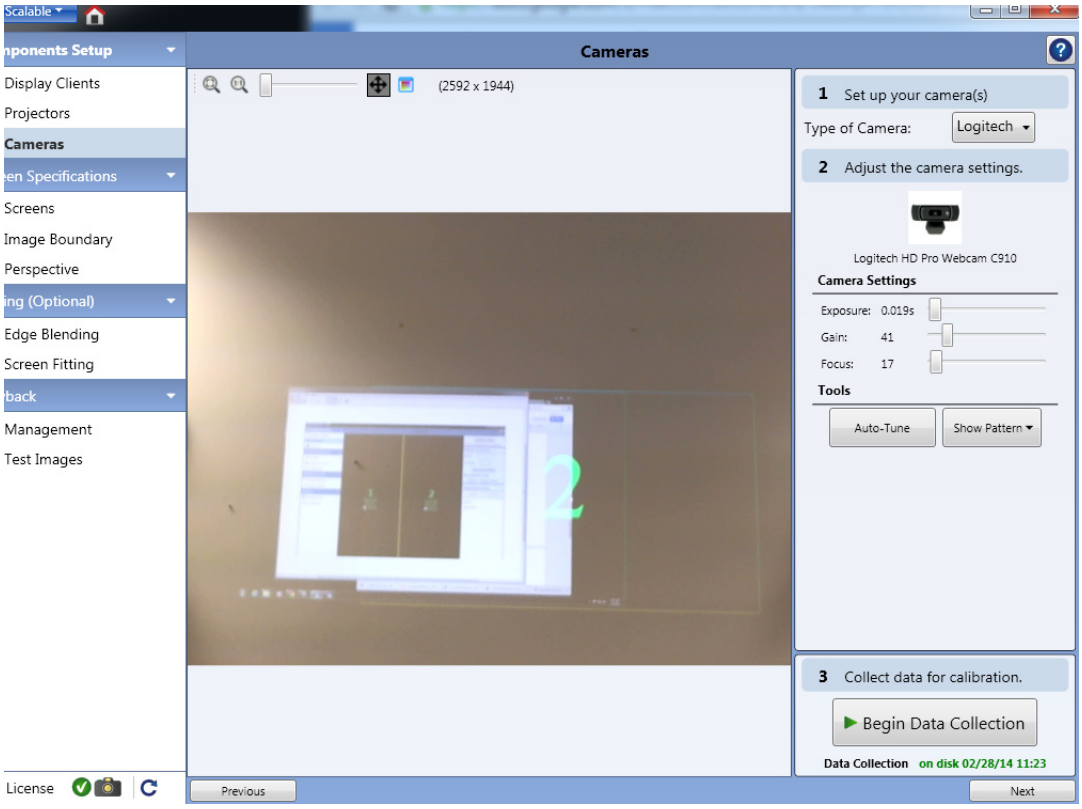


Figure 10-3. Cameras

6. In the Type of Camera list, select your camera.
7. Click the Auto-Tune button to have it configure the right camera parameters.
8. Click Begin Data Collection.
9. On the Screen Specification Panel, select Screens (Figure 10-4).

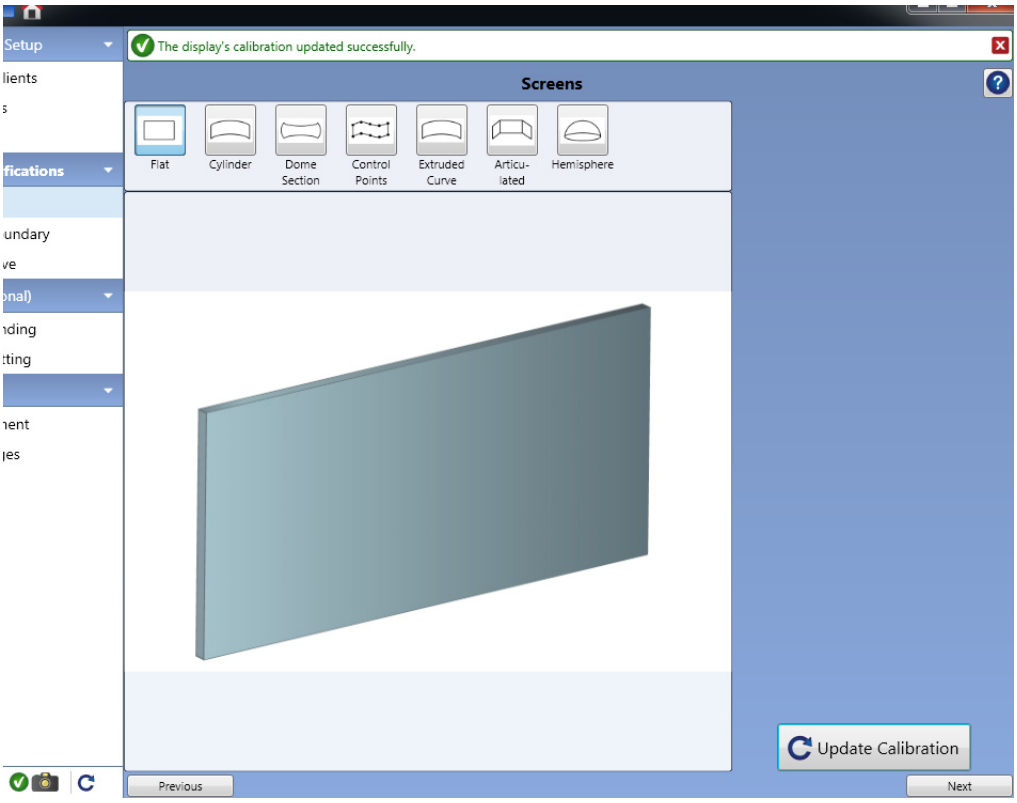


Figure 10-4. Screens

10. Select Flat. (This asumes you are projecting on a wall. If not, select the screen you are setting up. The set up that follows is setting up flat screens.)
11. Click the Update Calibration button.
12. On the Screen Specification Panel, select Image Boundary (Figure 10-4).

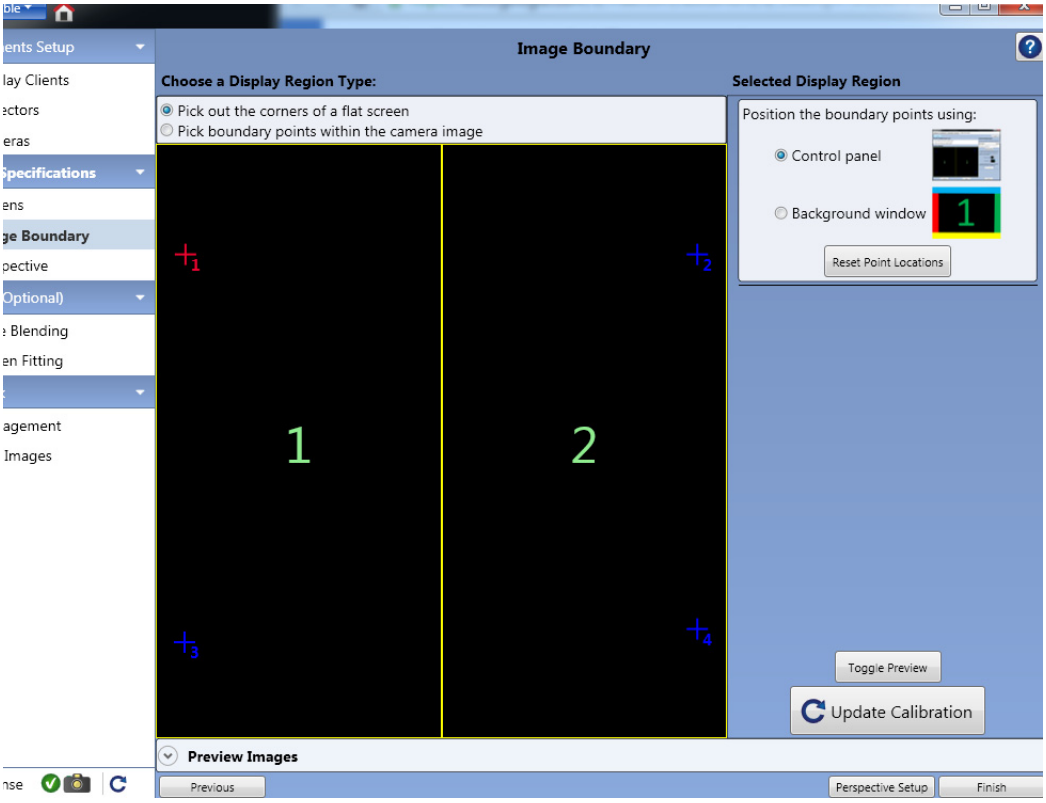


Figure 10-5. Image Boundary

13. Move the four corners to make a rectangle on the display wall.
14. Click Update Calibration.
15. On the Screen Specification Panel, select Perspective (Figure 10-4).

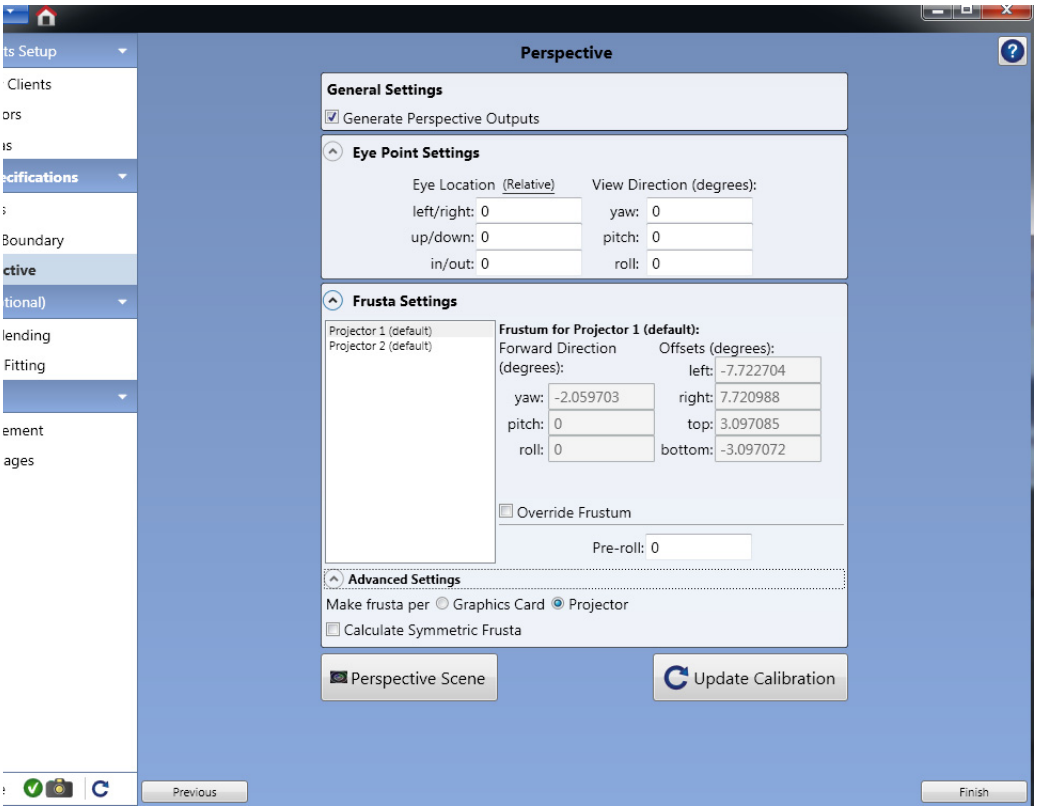


Figure 10-6. Perspective

16. Click the Perspective Scene button. Spherical cones are displayed on the perspective grid on the wall. You should see circles and not ovals. If you see ovals, you have to do the following:

    a. On the Scalable menu, choose **Support** → **Advanced Options**.

    b. Specify the following values (in meters):

    - FlatScreenWidth, double
    - FlatScreenHeight, double

    This defines the width and height of a flat screen for SDK programs. The default is 1.0, 1.0, which will probably stretch the image horizontally because screens rarely have a 1 to 1 aspect ratio. It is used in conjunction with the standard option "Flat Screen Distance" (FlatScreenZ).

c. Set Flat Screen Distance (FlatScreenZ). This is usually measured as the distance from the wall or screen and the eyepoint. When computing the perspective calibration data for the EasyBlend SDK, this is the distance away from (0,0,0), down the -Z axis, of the screen. It is measured as a multiplier of the width of the screen, and defaults to 1 width away. Decreasing this value will increase the perspective field-of-view generated for the SDK. Increasing this value will shrink the perspective field-of-view. This value should always be positive.

   For example, if the camera is 114" from the wall, the chosen eye point is at 80 inches and the display is 95 inches wide (W) by 29 inches high (H) the following values can be converted to meters: FlatScreenWidth, FlatScreenHeight and eye point distance are (2.4 (W), 0.74 (H) and 2.0 (eye point to screen distance) respectively). With these values the FlatScreenZ can be calculated:

```
FlatScreenZ = EyePointDistance/FlatScreenWidth = 2.0/2.4 =
   0.83 meters.
```

d. Click Finish.

e. Shut down the SDM.

f. Restart the SDM.

g. On the Screen Specification Panel, select Perspective (Figure 10-4).

h. Click Update Calibration. If you see circles on the wall, you are finished.

17. Click Finish. The mesh will be written out to *C:\Program Files\Scalable Display\DEI\LocalCalibration\ScalableData.ol*. This is the default that the plug-in looks for. Use the environment variable MAK_SCALABLEDATA if you want to use a different mesh. If you move the projectors you must reconfigure the mesh.

10.4. Running VR-Vantage with the Scalable Plug-in

1. Turn on the projectors.

   To use the Scalable Display Technologies plug-in, start the VR-Vantage applications with the `--plugin` command-line option, as follows:

   ```
   vrvapplication --plugin vrvScalableDisplayPlugin.dll
   ```

2. On the master computer, run the *vantageIGScalableDisplay.bat*.
3. On the client computers, run the *displayEngineScalableDisplay.bat*.
4. On the master, choose **View → Display Engine Configuration Editor**.
5. Click the Add Display Engine button.
6. Use the computer's name as the display engine hostname. You will not have to configure any channel or window parameters.
7. On the master, load a terrain (such as VR-Village). When it is loaded, press **`Ctrl+Enter`** to go into Fullscreen mode. You should now see a big, edge blended display! The picture will look distorted unless viewed from somewhere near the camera (unless you configured it to be viewed from some other point).



III. Scenes and Environmental Effects

The chapters in this section describe scenes and environmental effects such as weather, marine conditions, lighting, and sensor effects.



11. Scenes and Environmental Effects

This chapter describes scenes and environment conditions.

Working with Scenes 11-2
Creating a Scene 11-2
Saving a Scene 11-2
Opening a Scene 11-2
Setting the Date and Time of Day 11-3
Setting the Time Scale 11-4
Specifying the Environment Conditions 11-5
Setting the Default Environment Condition 11-7
Adding an Environment Condition 11-7
Editing an Environment Condition 11-8
Deleting an Environment Condition 11-8
Configuring the Weather 11-9
Setting the Wind Speed and Direction 11-9
Setting Visibility (Fog) 11-10
Specifying Precipitation Type and Intensity 11-11
Specifying Cloud Cover 11-11
Configuring Marine Conditions 11-12
Enabling Marine Effects 11-13
Configuring Marine Conditions 11-14
Enabling Tidal Stream Wakes 11-16
Setting Ocean Quality 11-17
Configuring Breaking Waves 11-17
Displaying Water Droplets and Splash Effects 11-18
Enabling Sea Spray 11-20
Setting the Size of Water Droplets 11-21
Enabling High Quality Rotor Wash 11-21
Displaying Remote Environment Settings 11-22
Disabling Lighting and Clouds 11-22

11.1. Working with Scenes

A scene contains an environment, the time of day, and a terrain filename. If particular environment settings are important to the simulations you are viewing, use of scenes can be an important convenience feature.

11.1.1. Creating a Scene

To create a scene:

1. Optionally, choose **File** → **New Scene**, or click the New Scene button () on the Scene Toolbar. If a terrain is currently loaded, it is removed from the scene.
2. Load the terrain that you want to include in the scene. For details, please see "Opening a Terrain," on page 3-10 and "Terrain Agility and Composability," on page 25-2.
3. Optionally, set the date and time of day.
4. Optionally, specify environment settings, as described in this chapter.

11.1.2. Saving a Scene

To save a scene:

1. Choose **File** → **Save Scene**, or click the Save Scene button () on the Scene Toolbar. The Save Scene dialog box opens. If you did not load a terrain when you created the scene, or if you built a terrain, but did not save it, you are prompted to specify a terrain for the scene.
2. Type a name for the scene.
3. Click Save.

11.1.3. Opening a Scene

To open a scene:

1. Choose **File** → **Open Scene**, or click the Open Scene button () on the Scene Toolbar. The Open Scene dialog box opens.
2. Select the scene you want to open.
3. Click Open.

11.2. Setting the Date and Time of Day

VR-Vantage changes the ambient, diffuse, and specular light based on the simulated time of day and the date. Figure 11-1 illustrates different time of day settings.

When you save a scene, VR-Vantage saves the time of day as part of the scene.



5:15 A.M., June 1 11:15 A.M., June 1

Figure 11-1. Time of Day

> You can synchronize the time of day with VR-Forces. For details, please see "Displaying Remote Environment Settings," on page 11-22.

To set the date and time of day:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Date and Time page (Figure 11-2).

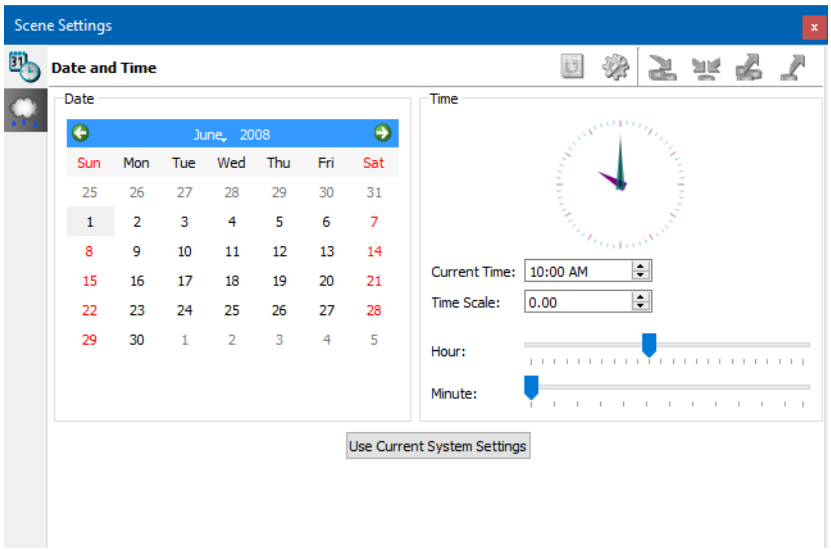


Figure 11-2. Date and Time page

3. To change the month, click the right or left arrows, or click the month and select a month from the list.
4. To change the year, do one of the following:
   – Click the right or left arrows to cycle through the months to a new year.
   – Click the year and type in a year.
   – Click the year and use the spin controls to select a year.
5. To specify the date, select a date in the calendar.



If you change the month, you must explicitly select a date for it to be set.

6. To specify the time, in the Time group box, in the Current Time box, type or select a time value, in HH:MM:SS format, or drag the Hour and Minute sliders. The analog clock display changes to reflect the current simulated time.
7. To set the date and time of day to the system settings, click Use Current System Settings.

11.2.1. Setting the Time Scale

VR-Vantage advances the date and time based on a time scale. By default the time scale is 0.00 and VR-Vantage does not advance time. A time scale of 1.00 means VR-Vantage advances time at real time. The time advance affects VR-Vantage's display of light, sun, star, and moon positions. It has no relationship to or effect on the simulation that VR-Vantage is displaying.

To set the time scale:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Date and Time page (Figure 11-2).
3. In the Time group box, type or select a value in the Time Scale box.

11.3. Specifying the Environment Conditions

VR-Vantage has a set of preconfigured environment conditions. Each environment condition has settings for:

- Weather:
  - Wind speed and direction.
  - Cloud cover.
  - Visibility.
  - Precipitation type and intensity.
  - Fog height and color.
- Marine:
  - Sea swell.
  - Sea state.
  - Surface transparency.
  - Underwater visibility.
  - Choppiness.
  - Surge depth.

Conditions range from clear to stormy and let you quickly set the weather without needing to adjust all the various weather parameters. However, if you want to, you can change any of the individual weather parameters. Changing a weather or marine setting does not affect the saved settings unless you specifically overwrite them.

You can save and import environment settings. When you save a scene, environment settings are saved as part of the scene. Environment settings support the standard VR-Vantage settings behaviors, as described in "Managing VR-Vantage Settings," on page 3-15.

You can add your own environment conditions. You can also set an environment condition as the default condition to use when VR-Vantage starts up. If you do not set a default condition, VR-Vantage saves the settings that were in effect when you last closed it.



The settings on the Environment Conditions Settings page always reflect the current settings for environment conditions. They do not necessarily match any of the preconfigured environment conditions as listed in the Apply Environment Conditions list except immediately after a preconfigured environment condition is applied. In other words, the Apply Environment Conditions list lets you apply environment conditions, but is not intended to summarize the status of the current environment conditions.

To specify an environment condition:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Environment Conditions Settings page (Figure 11-3).

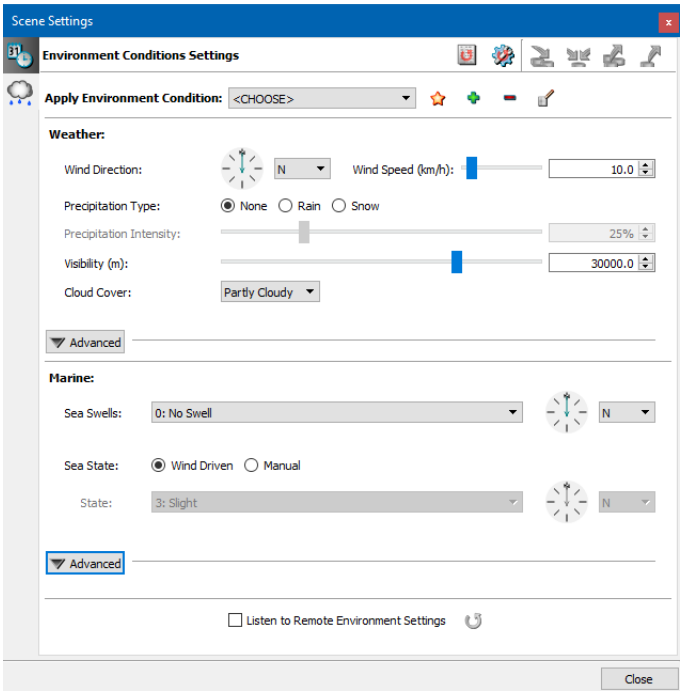


Figure 11-3. Environment Conditions Settings page

3. In the Apply Environment Condition list, select the condition you want.

> After you select a preconfigured environment condition, the list does not show the option you selected. It applies the conditions and the settings in the rest of the page change to reflect those conditions.

11.3.1. Setting the Default Environment Condition

The default environment condition is the condition that is in effect when you start VR-Vantage.

To set the default environment condition:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Environment Conditions Settings page (Figure 11-3).
3. Click the Star button (). The Select an Environment Condition dialog box opens.
4. In the environment condition list, select the condition you want to be the default.
5. Click OK. The selected environment condition now has a star next to it in the Apply Environment Condition list.

11.3.2. Adding an Environment Condition

If you add an environment condition, it gets added to the Apply Environment Condition list.

To add an environment condition:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Environment Conditions Settings page (Figure 11-3).
3. Change the weather settings to the ones you want for the new condition.
4. Click the Add button (). The Name an Environment Condition dialog box opens.
5. Type a name for the new condition.
6. Click OK.

11.3.3. Editing an Environment Condition

You can change the settings for an environment condition. Editing a condition essentially means overwriting a condition with the current set of weather and marine conditions.

To edit an environment condition:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Environment Conditions Settings page (Figure 11-3).
3. Optionally, in the Apply Environment Condition list, select a condition you want to use as your starting point.
4. Change whatever conditions you want.
5. Click the Edit button (). The Select an Environment Condition dialog box opens.
6. In the Select Name list, select the condition you want to overwrite.
7. Click OK.

11.3.4. Deleting an Environment Condition

If you delete one of the built-in environment conditions, you can restore it from the factory settings.

To delete an environment condition:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Environment Conditions Settings page (Figure 11-3).
3. Click the Delete button (). The Select an Environment Condition dialog box opens.
4. In the Select Name list, select the condition you want to delete.
5. Click OK.

11.4. Configuring the Weather

This section describes the individual weather conditions that you can set.

11.4.1. Setting the Wind Speed and Direction

You can set the wind speed and direction. If you do not specify the sea state manually, it is set based on the wind conditions. The maximum wind speed that you can set is 360 km/h.) The wind direction affects the angle at which precipitation falls. Wind speed and direction affect particle systems (dust clouds, smoke, and so on).

To set wind speed and direction:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Environment Conditions Settings page (Figure 11-4).

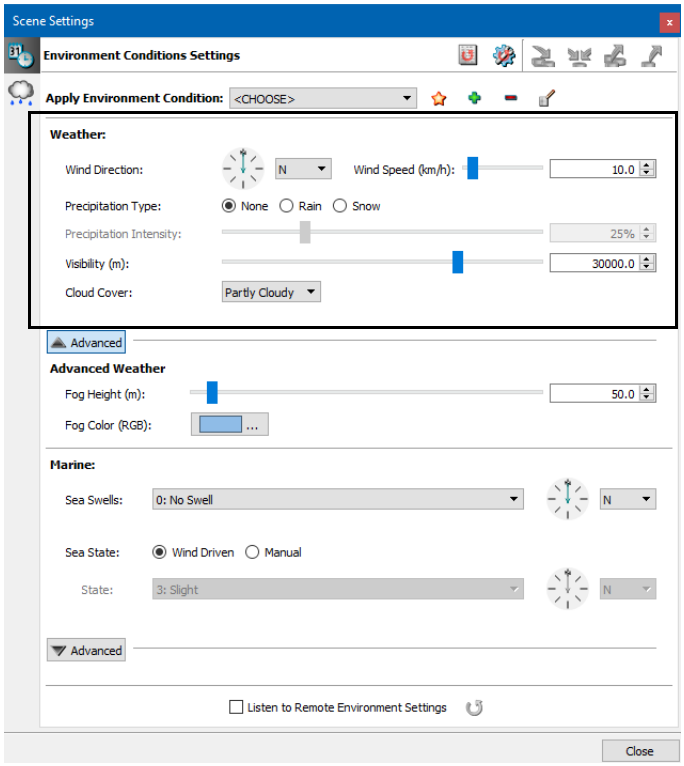


Figure 11-4. Environment Conditions Settings page

3. Drag the Wind Direction indicator to the desired compass point or select an option from the list.
4. Drag the Wind Speed slider or enter a value in the box.

11.4.2. Setting Visibility (Fog)

The visibility setting determines the denseness of the fog effect, which determines how far you can see.

To set visibility:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Environment Conditions Settings page (Figure 11-4).
3. Adjust the Visibility slider or enter a value in the box.

Configuring Fog

You can set the fog height and color. The fog height affects how much sky the observer can see when looking up from ground level. For example, if you set the fog height to 50m and look up, you will see some amount of sky above you, and to the sides it will be very gray. If you then set it to 10m and look up, you will see mostly blue sky above you, but still quite a bit of gray to the sides.

To configure fog:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Environment Conditions Settings page (Figure 11-4).
3. In the Weather section of the page, click Advanced. The dialog box expands to show options for fog.
4. To set the fog height, adjust the slider or enter a value in the box.
5. To set the fog color:
   a. Click the color swatch. A color picker dialog box opens.
   b. Select the color you want to apply to the fog. It is applied immediately, so you can easily see the effect of the new color and change it if you want to.
   c. Click OK.

11.4.3. Specifying Precipitation Type and Intensity

VR-Vantage supports display of rain and snow. You can configure the intensity of precipitation.



Precipitation has no effect on the simulation. It is only a visual effect. Snow does not accumulate.

To specify the precipitation type and intensity:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Environment Conditions Settings page (Figure 11-4).
3. Select a Precipitation Type. If you select rain or snow, the Precipitation Intensity setting becomes editable.
4. Adjust the Precipitation Intensity slider or enter a value in the box. The range is 0% through 100%.

11.4.4. Specifying Cloud Cover

VR-Vantage simulates varying amounts and types of cloud cover, from clear, to cloudy, to thunderstorm. When you choose an environment condition, it includes an appropriate degree of cloud cover. You can also change the cloud cover at any time.

To specify the cloud cover:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Environment Conditions Settings page (Figure 11-4).
3. Select an option from the Cloud Cover list.

Calculating Cloud Positions Before You Load a Terrain

When you change the cloud cover, VR-Vantage can stall while it calculates the size and location of clouds. To prevent this problem, VR-Vantage can calculate cloud locations when it loads a terrain or scene. To calculate cloud positions when you load a terrain or scene, start VR-Vantage with the `--preLoadClouds` command line argument.

Preloading clouds increases the amount of time it takes to load a terrain or scene.

11.5. Configuring Marine Conditions

VR-Vantage supports a variety of dynamic ocean effects (Figure 11-5), including:

- Douglas Sea State. This includes wind waves (wind sea), swell character, and the directions of each. VR-Vantage allows each to be separately configured.
- Wave chop.
- Surface transparency. The ability to see through the water from above sea level.
- Underwater visibility. The ability to see underwater.
- Swell.
- Surge depth. When bathymetry data is present, this is the depth of the water at which full storm effects take place. As the water gets shallower, it gets calmer.
- Tidal current speed and direction. If tidal stream wakes are enabled, they are automatically applied to streamed buoys and beacons.



Figure 11-5. Fishing boat

Enabling dynamic ocean effects can affect performance.

11.5.1. Enabling Marine Effects

Marine effects can reduce performance.

To enable marine effects (dynamic ocean):

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 11-6).

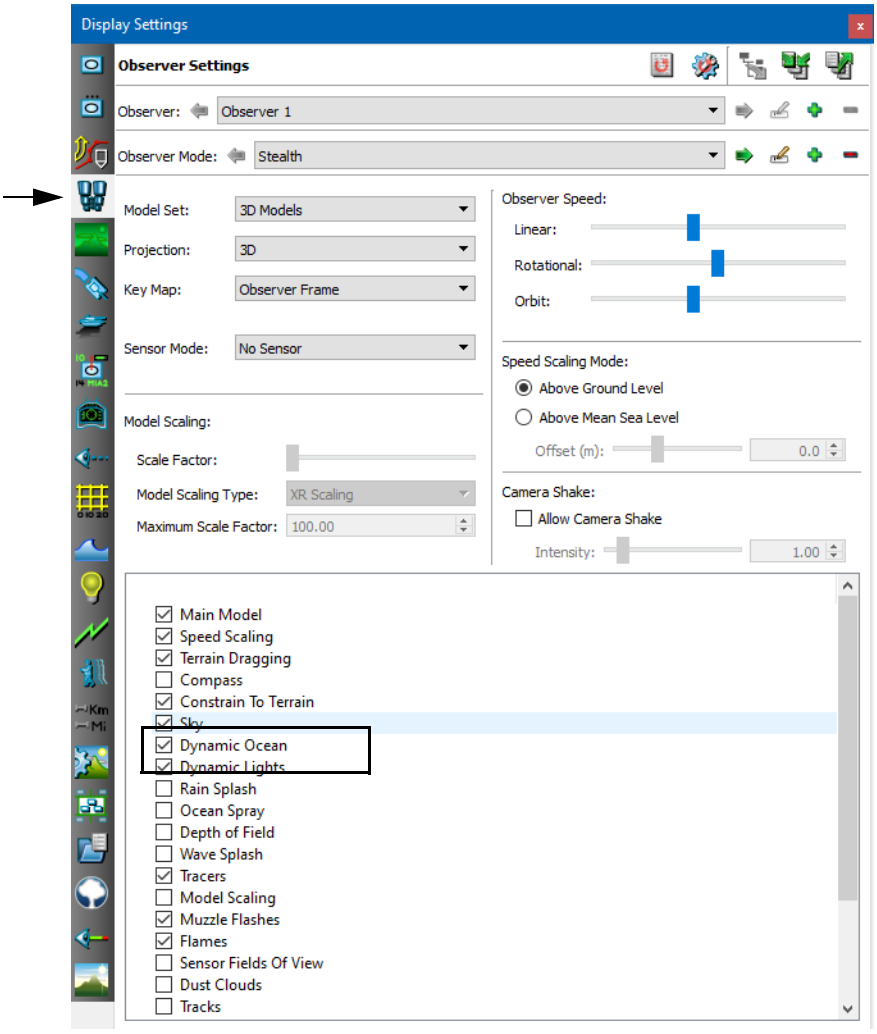


Figure 11-6. Observer Settings page

3. Select or clear the Dynamic Ocean check box.

> You can also enable or disable dynamic ocean by choosing **Observer** → **Dynamic Ocean**, or clicking the Dynamic Ocean button ( ) on the Observer Settings toolbar.

11.5.2. Configuring Marine Conditions

The preconfigured environment conditions include marine conditions. You can also customize the marine settings at any time.

To configure marine effects:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Environment Conditions Settings page (Figure 11-7).

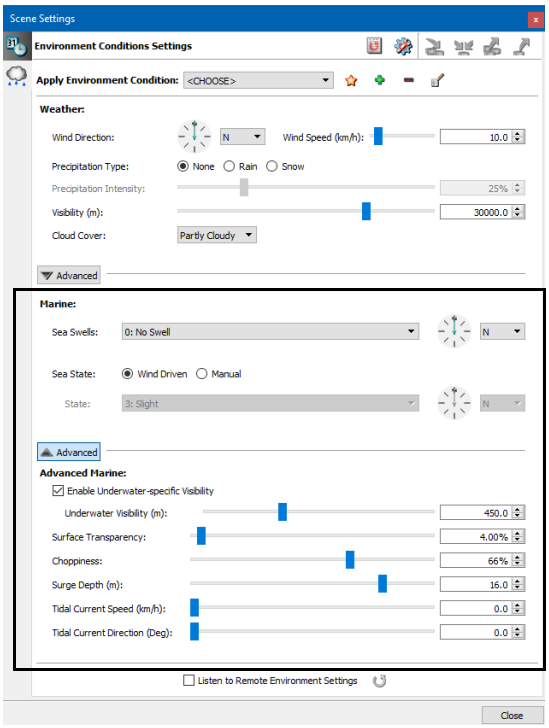


Figure 11-7. Marine Settings

3. Change settings as desired. The settings are described in Table 11-1.

Table 11-1: Advanced marine settings

Parameter	Description
Sea Swells	The amount of rise and fall of waves independent of wind. Choose from 10 fixed swell states. You can also set the compass direction from which swells originate.
Sea State	Determined by the wind direction and speed or set manually. If manual, one of the swell states for the Douglas sea state scale. You can also set the compass direction from which sea state originates.

Table 11-1: Advanced marine settings

Parameter	Description
Advanced Parameters	
Surface Transparency	The depth at which VR-Vantage begins to apply transparency to the water surface if the terrain has bathymetry or underwater polygons. This setting helps reduce Z-fighting along the shoreline. It does not mean that you can see some distance into the water when the observer is over deep water. The maximum value represents transparency to 500 meters.
Underwater visibility	The distance you can see underwater if the observer is underwater.
Choppiness	The percentage of turbulence on the water.
Surge depth	When bathymetry data is present, this is the depth of the water at which full storm effects take place. As the water gets shallower, it gets calmer.
Tidal Current Speed	The speed of the tidal current. This affects the display of tidal wakes on buoys and beacons. Default: 0.
Tidal Current Direction	The direction of the tidal current flow. This affects the display of tidal wakes on buoys and beacons.

11.5.3. Enabling Tidal Stream Wakes

Tidal stream wakes create wake effects around buoys and beacons based on the movement of the tides. Tidal stream wakes are supported as stand-alone models. They are automatically attached to streamed buoys and beacons. VR-Vantage includes wake models for each of the supported buoy and beacon models. They are named for the buoy or beacon type they support, for example, BuoysPillarWake or BeaconsLatticeWake. You may want to edit the **wakeLength** parameter to provide the best visuals for your terrain.

Tidal stream wakes are disabled by default.



Even if tidal stream wakes are enabled, you will not see any wakes unless you also specify a tidal current, which is 0 by default. For details about setting tidal current, please see "Configuring Marine Conditions," on page 11-14.

To enable or disable tidal stream wakes:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Ocean Settings page (Figure 11-8).

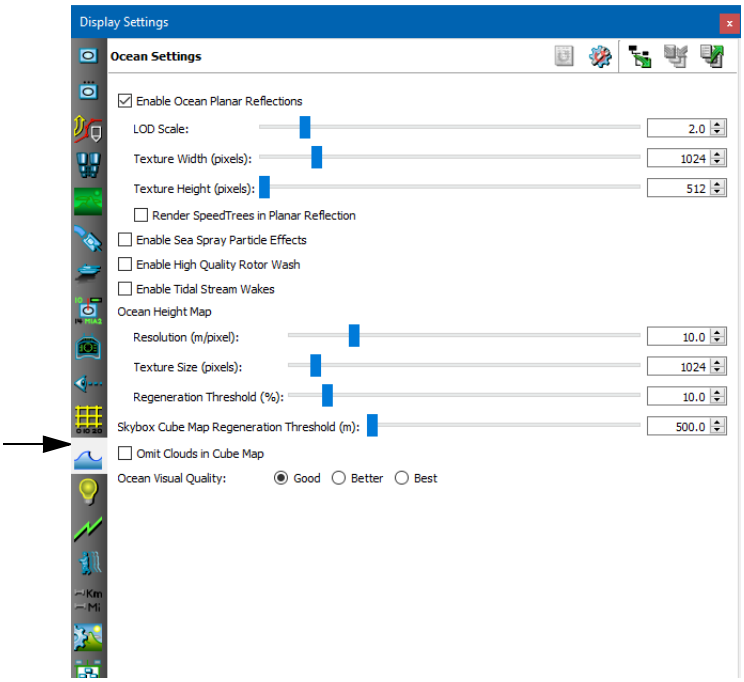


Figure 11-8. Ocean Settings

3. Select or clear Enable Tidal Stream Wakes.

11.5.4. Setting Ocean Quality

You can adjust the overall quality of visualization of the ocean. Increasing ocean quality decreases performance. Best quality may affect overall performance of your computer. Therefore this setting is primarily for use in demonstrations rather than large scenarios.

To set ocean quality:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Ocean Settings page (Figure 11-8).
3. Select one of the Ocean Visual Quality options.

11.5.5. Configuring Breaking Waves

VR-Vantage can simulate the effect of waves breaking along the shore line. This feature uses shapefile data that defines the direction of the breaking waves. The *MAK Earth (online).mtf* terrain includes sample data for breaking waves. To see it, view the island of Oahu, Hawaii.

You can set the amplitude of breaking waves. By default, the amplitude is 0.



This is considered an experimental feature. It may be useful, but may affect performance.

To configure breaking waves:

1. Choose **Settings** → **Experimental Features**. The Experimental Features dialog box opens (Figure 11-9).

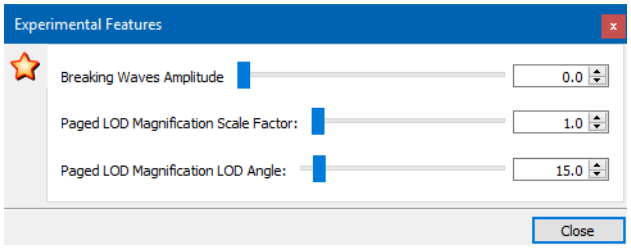


Figure 11-9. Experimental Features dialog box

2. Adjust the Breaking Waves Amplitude slider or type a value in the box. The maximum value is 10.0.

11.6. Displaying Water Droplets and Splash Effects

When it is raining, VR-Vantage can simulate raindrops hitting the observer's camera lens (Figure 11-10). If the observer is close to the surface of the ocean (as if a periscope were being raised or lowered), VR-Vantage can simulate the effect of waves and water droplets splashing on the observer's camera.



Figure 11-10. Rain effect

When dynamic ocean is enabled, VR-Vantage can simulate sea spray (most visible in high seas) (Figure 11-11). It can also display rotor wash for hovering helicopters. You can display rotor wash at two levels of quality.

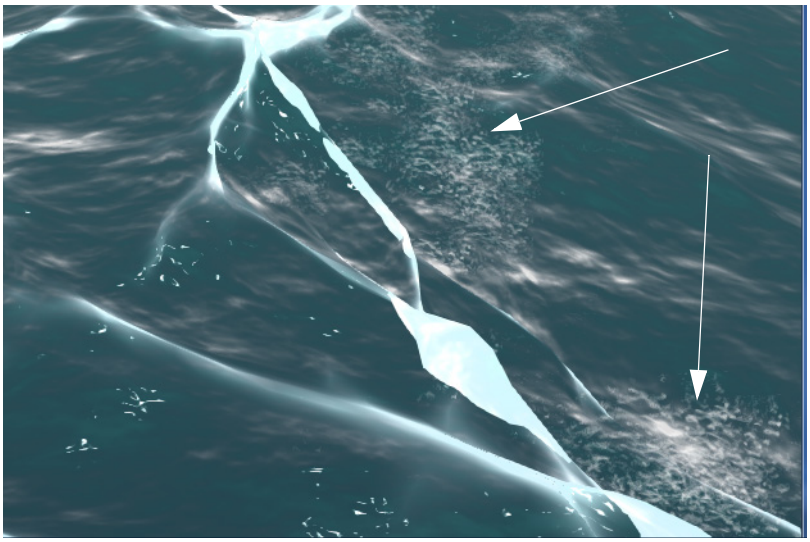


Figure 11-11. Sea spray effects

You can change the size of the water droplets for splash effects and sea spray. For details, please see "Setting the Size of Water Droplets," on page 11-21.

To enable or disable splash effects:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 11-12).

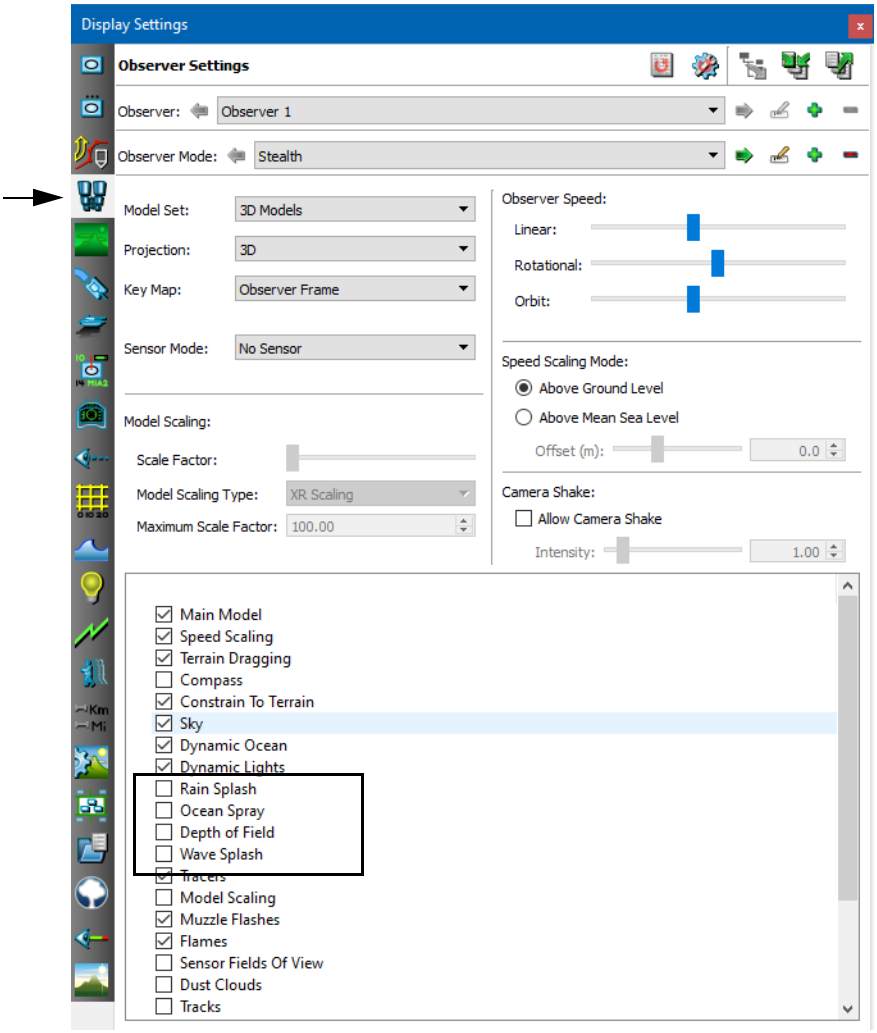


Figure 11-12. Observer Settings page

3. Select or clear the Rain Splash, Wave Splash, or Ocean Spray effect check boxes.

> You can also enable or disable these effects by clicking the appropriate button on the Observer Settings toolbar (rain splash ( ), wave splash ( ), ocean spray ( ).

11.6.1. Enabling Sea Spray

To enable sea spray effects:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Ocean Settings page (Figure 11-13).

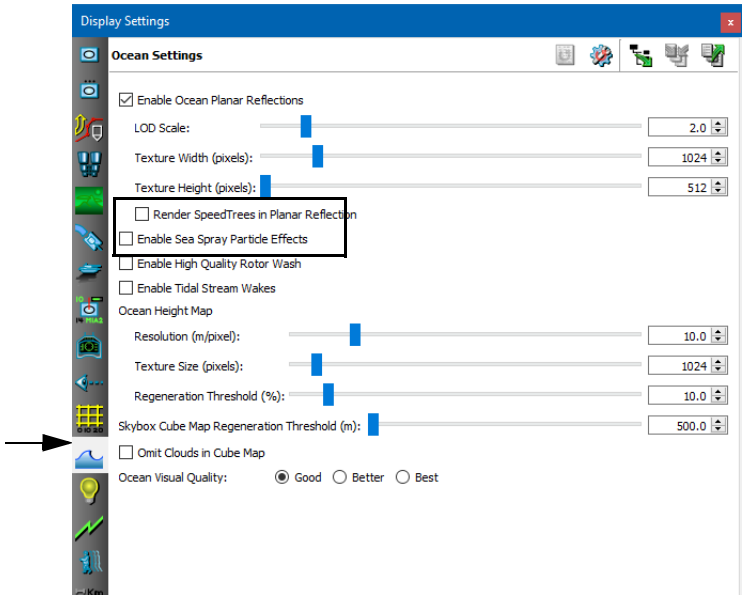


Figure 11-13. Ocean Settings page

3. Select Enable Sea Spray Particle Effects.

11.6.2. Setting the Size of Water Droplets

You can change the size of water droplets for splash effects and sea spray.

To change the size of water droplets:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 11-14).

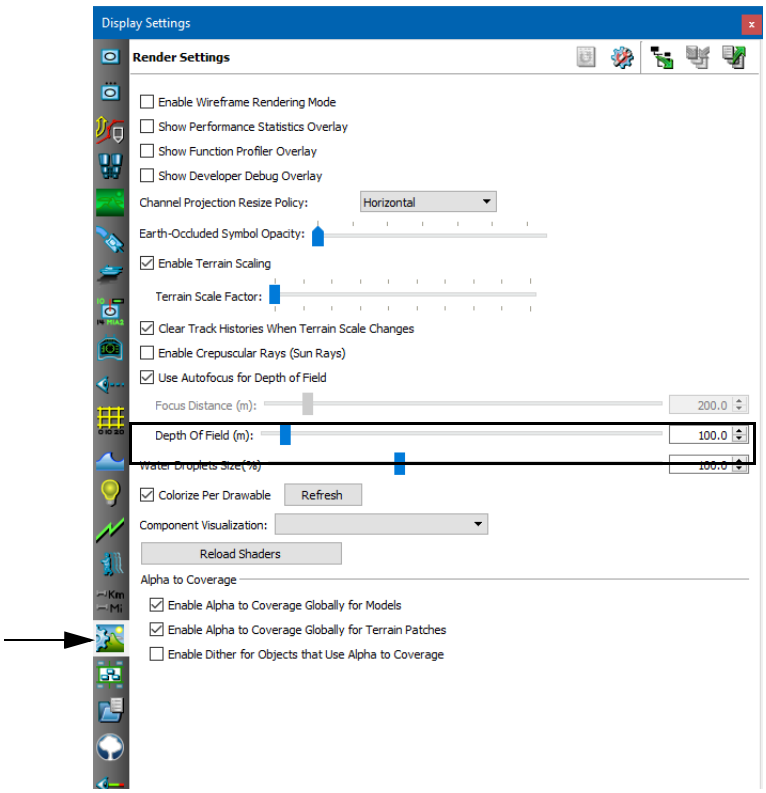


Figure 11-14. Render Settings page

3. Adjust the Water Droplets Size slider or enter a value in the box.

11.6.3. Enabling High Quality Rotor Wash

To enable high quality rotor wash:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Ocean Settings page (Figure 11-13).
3. Select Enable High Quality Rotor Wash.

11.7. Displaying Remote Environment Settings

The weather effects that you configure in VR-Vantage are just visual effects. They do not correspond to any weather effects in the simulation you are viewing, nor do simulation objects in a simulation interact with them in any way. However, some simulation applications can simulate weather effects and these effects do affect simulation object behaviors, such as calculating line-of-sight. VR-Vantage can display weather effects generated by VR-Forces.

When this feature is enabled, the weather messages from VR-Forces override the settings in VR-Vantage for environment conditions and you cannot make any changes to them in VR-Vantage. If VR-Vantage does not receive any weather messages, it continues to use the environment conditions that were in effect when you enabled listening for remote environment settings.



Listening to remote environment conditions also synchronizes date and time of day with VR-Forces.

Changes to fog height or fog color in VR-Forces do not get sent to VR-Vantage.

To enable remote weather effects:

1. Choose **Settings** → **Scene**. The Scene Settings dialog box opens.
2. Select the Environment Condition Settings page (Figure 11-4).
3. Select the Listen to Remote Environment Settings check box.
4. Click the Refresh button ().

11.8. Disabling Lighting and Clouds

When you are using XR observer mode and you move the observer out from the terrain, the lighting and cloud effects can interfere with your ability to view simulation objects. Therefore, VR-Vantage lets you quickly disable lighting effects and clouds. By default, they are enabled in Stealth observer mode and disabled in XR observer mode.

➤ To enable or disable lighting and clouds, choose **Observer** → **Sky**, or click the Show/Hide Sky button () on the Observer Settings Toolbar.



12. Using Sensors

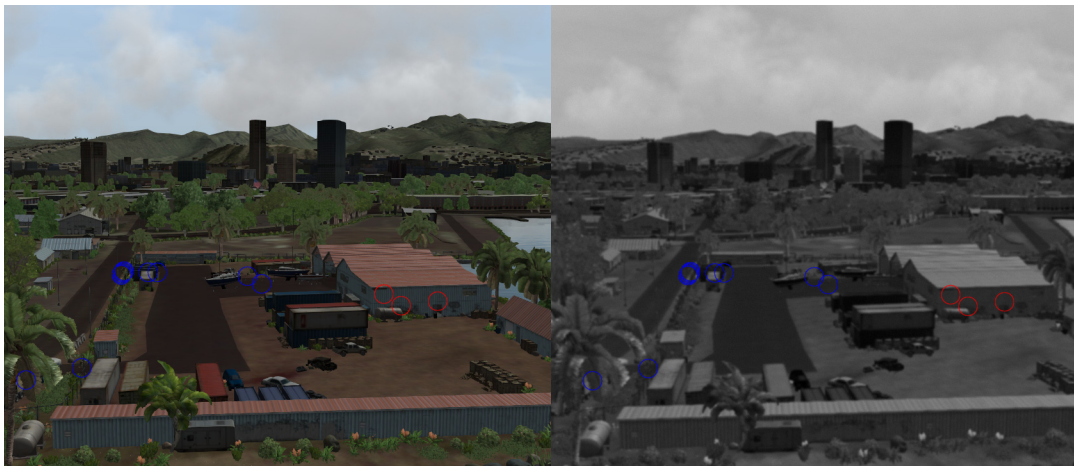
This chapter explains how to use the sensor feature to simulate a night-vision goggle (NVG) or infrared view of the scene.

Using Sensors 12-2
Enabling Sensors 12-3
Configuring Sensors 12-4
Adding New Sensors 12-6

12.1. Using Sensors

VR-Vantage Stealth and VR-Vantage IG can simulate the way that the 3D scene would look if an observer were using visual sensor devices such as night vision goggles or viewing the infrared spectrum (Figure 12-1). This is an observer-specific setting. You can adjust the contrast, blur, and noise of the view.

VR-Vantage includes one sensor module – CameraFX, which uses the SenSim libraries from JRM Technologies. The sensors do not take into account the materials of the objects that you are viewing. They simply filter the view to produce the desired effect. If you want physics-based sensor effects, you can purchase the SensorFX Plug-in to VR-Vantage, developed by JRM. For details, please contact your MAK salesperson.



No sensor (daylight) IR sensor

Figure 12-1. CameraFX

12.2. Enabling Sensors

Sensors are observer-specific (Figure 12-2). VR-Vantage includes observer modes that have sensors enabled. The names of the observer modes match the sensor modes that they enable.

- NVG (CameraFX) (night vision goggles).
- IR (CameraFX) (infrared).
- IR BlackHot (CameraFX)
- Camera (CameraFX). This mode uses the Visual Camera sensor mode.

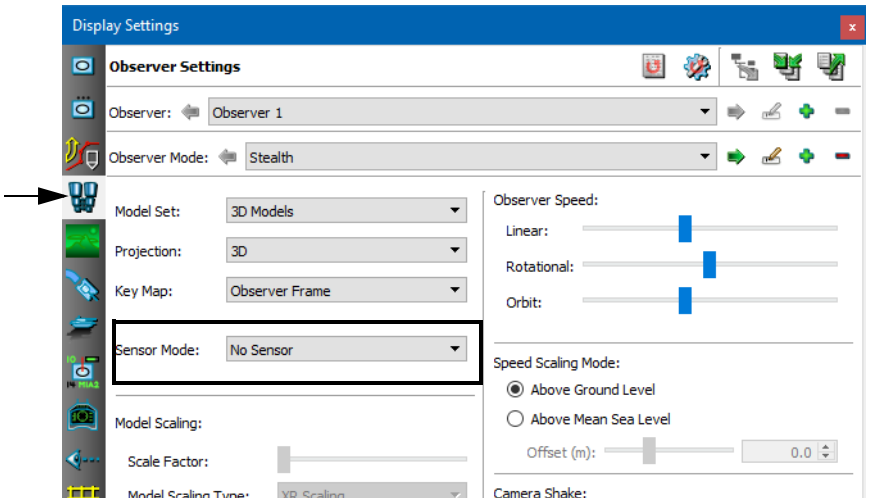


Figure 12-2. Sensor setting for observer mode

The default settings for the Camera observer mode make it look very similar to Stealth observer mode. The difference is that when you use the Camera observer mode, you can change the view by changing the sensor settings, such as blur, noise, and gain.



If you have SensorFX installed, when you run it, the observer modes include the text SensorFX instead of CameraFX. The presence of CameraFX or SensorFX in the observer mode name makes it clear which type of sensor you are using.

You can also specify use of a sensor configuration as a channel attribute.

To enable sensors, do one of the following:

- Select an observer mode that has a sensor configured.
- Edit an observer mode to use a sensor.
- Create a new observer mode and configure it with a sensor.
- Change the Sensor attribute for a channel.

For details about creating and editing observer modes, please see "Creating and Editing Observer Modes," on page 14-4. For details about editing channel attributes, please see "Changing a Channel's Attributes," on page 6-12.

12.3. Configuring Sensors

You can configure the following aspects of sensors:

- Blur. The sharpness of the image.
- Noise. The graininess of the image.
- Gain. The contrast and brightness of the image. (IR only)

You can configure a separate set of settings for each sensor type. VR-Vantage remembers the settings. When you change the settings for a sensor type, they affect the display only if that sensor type is in use. For example, if you select NVG in the sensor list on the Observer Settings page, and select IR in the Sensors list on the Sensor Settings page, the changes you make to the IR configuration do not affect the NVG display you are looking at. If you then switch to IR in the VR-Vantage window, the new IR settings take effect. (This probably sounds obvious, but the Sensor Settings page does not automatically display the configuration for the sensor that is being used. If you want to edit the sensor that you are currently using, you have to be sure that you have selected it to edit.)

To configure sensors:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Sensor Settings page (Figure 12-3).

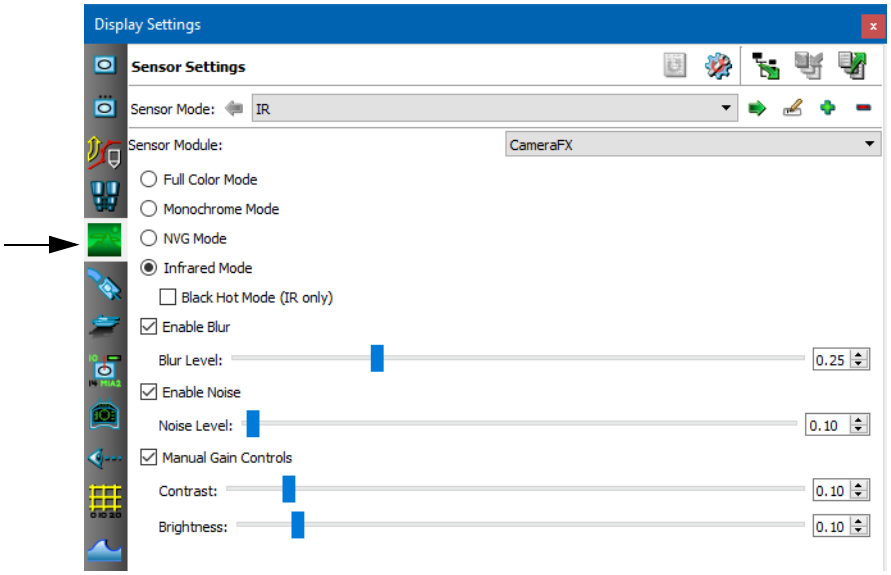


Figure 12-3. Sensor Settings page

3. In the Sensor Mode list, select the sensor that you want to configure.
4. Select the Sensor module from the list.
5. Select the sensor mode (Monochrome, NVG, and so on). If you select Infrared Mode, select or clear the Black Hot Mode check box.

- It may seem obvious that if, for example, you select NVG from the Sensor Mode list, you would select it for the sensor type. However, VR-Vantage lets you add additional sensor configurations to the list and in that case, you would need to specify the correct sensor type for the new sensor. For example, you might create a sensor called NVG-High Contrast. Then you would select NVG as the sensor type and adjust the contrast accordingly.
- Full Color Mode provides natural coloring, as you would see with no sensor modes applied. The benefit of creating a sensor mode using this option is that you can apply noise or blur to the scene.

6. Select the check boxes for the effects that you want to be enabled for this sensor configuration.
7. Adjust the values for the enabled effects.

12.4. Adding New Sensors

You can add, rename, and delete sensors.

To add a sensor:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Sensor Settings page (Figure 12-3).
3. On the Sensor Mode line, click the Add button (). A new sensor called sensor_*number* is added. There are no configuration options on the page. (To rename the sensor, click the Rename button () and enter a new name.)
4. In the Sensor Module list, select CameraFX. The available configuration options are added to the page.
5. Select the type of sensor that you want this to be (Full Color, Monochrome, Infrared, or NVG).
6. Set the Blur, Noise, and Gain levels.



13. Lighting Effects

This chapter describes the lighting effects supported by VR-Vantage.

Displaying Lighting Effects 13-2
Enabling and Disabling Lighting 13-2
Configuring Ambient and Diffuse Lighting 13-3
Specifying the Emissive Lighting Scale 13-4
Displaying Dynamic Lighting 13-5
Daylight Control of Lights 13-6
Displaying Ocean Planar Reflections 13-7
Configuring Ocean Planar Reflections 13-8
Lens Flare 13-9
Displaying Crepuscular (Sun) Rays 13-11
Configuring the Skybox Cube Map 13-13
Flashlight Mode 13-13
Using High Dynamic Range Lighting 13-14
Displaying Shadows (3D Only) 13-16
Configuring Shadows 13-18

13.1. Displaying Lighting Effects

VR-Vantage can simulate a variety of lighting effects, including:

- Dynamic lighting
- Ocean planar reflection
- Lens flare
- Crepuscular (sun) rays.
- Shadows.

Use of lighting effects affects performance. VR-Vantage lets you enable and disable some effects so you can adjust the mixture of performance and effects that best meets your needs.

13.2. Enabling and Disabling Lighting

When lighting is enabled, VR-Vantage takes into account direct and ambient lighting when it renders a scene. During the daytime, objects are brightly lit. At night, you cannot see them unless an object is casting light on them or you use a sensor mode that provides night vision.

If lighting is disabled you can see objects regardless of the time of day. Lighting may not be realistic.

To enable or disable lighting:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Lighting Render Settings page (Figure 13-1).

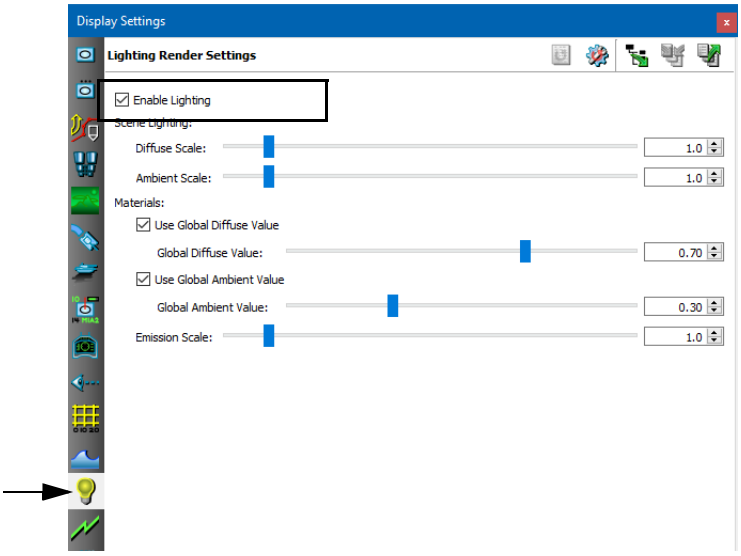


Figure 13-1. Lighting Render Settings

3. Select or clear the Enable Lighting check box.

13.2.1. Configuring Ambient and Diffuse Lighting

Ambient lighting is the lighting available to objects that are not in a direct form of light, such as sunlight. Diffuse lighting is light that is reflected off an object. You can adjust the overall amount of ambient and diffuse lighting in the scene and you can configure whether objects use the material ambient and diffuse values specified in their models or use values provided by VR-Vantage.

Configuring the Ambient and Diffuse Scale Values for Scene Lighting

You can set a scale factor for ambient and diffuse light that is present in the scene. Lower values make objects darker. Higher values make them lighter. The ambient or diffuse scale factor is independent of how material ambient and diffuse values are set.

To specify the ambient and diffuse lighting scale values:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Lighting Render Settings page (Figure 13-1).
3. Adjust the Diffuse Scale slider or change the value in the box.
4. Adjust the Ambient Scale slider or change the value in the box.

Configuring the Material Ambient and Diffuse Values for Materials

Polygons in a 3D scene have a material and a texture. Polygon materials may specify specular, diffuse, and ambient colors that define the way they respond to incoming light.

The ambient color specifies the color used when it is in indirect light. Sometimes the ambient values for different objects do not work well together. You can specify that objects use the ambient material value, or let VR-Vantage globally specify ambient material values.

In addition to specifying which material ambient and diffuse values to use, you can set a scale factor. Lower values make objects darker. Higher values make them lighter. The material ambient or diffuse scale factor is independent of how ambient and diffuse values are set for scene lighting.

To configure the material ambient value:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Lighting Render Settings page (Figure 13-1).
3. To use the ambient values specified by objects, clear the Use Global Ambient Value check box. To have VR-Vantage use a global value, select the Use Global Ambient Value check box.
4. If you selected the Use Global Ambient Value check box, optionally, adjust the Global Ambient Value slider or change the value in the box.

To configure the material diffuse value:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Lighting Render Settings page (Figure 13-1).
3. To use the diffuse values specified by objects, clear the Use Global Diffuse Value check box. To have VR-Vantage use a global value, select the Use Global Diffuse Value check box.
4. If you selected the Use Global Diffuse Value check box, optionally, adjust the Global Diffuse Value slider or change the value in the box.

13.2.2. Specifying the Emissive Lighting Scale

You can change a scale value that controls the brightness of emissive textures. This allows you to make windows, street lights, and other emissive light sources darker or brighter. Increasing the scale value increases the brightness.

To specify the emissive lighting scale value:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Lighting Render Settings page (Figure 13-1).
3. Adjust the Emission Scale slider or change the value in the box.

13.3. Displaying Dynamic Lighting

VR-Vantage can display light cast by light sources such as headlights, street lamps, and spotlights. Display of dynamic lighting is an observer-specific feature. It is disabled by default.

Dynamic lights are either part of the terrain or part of an entity model. If they are part of an entity model, they can be set permanently on or they can be turned on and off by the appearance bit.

Figure 13-2 shows a night-time scene with dynamic lighting enabled. The HMMWV's taillights cast light to the rear. The headlights cast light forward. Street lamps and windows are also casting light.



Figure 13-2. Dynamic Lighting enabled

- Light points can be configured to automatically turn on at dusk and turn off at dawn. For details, please see "Daylight Control of Lights," on page 13-6.
- Emissive lights, such as automobile headlights, are always on.

You cannot change these lighting settings.

To enable dynamic lighting,

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page.
3. In the list of settings, select Dynamic Lights.



You can also enable or disable dynamic lighting by choosing **Observer** → **Dynamic Lights** or clicking the Dynamic Lights button () on the Observer Settings Toolbar.

13.3.1. Daylight Control of Lights

Lights that have a light point type specified can be placed under daylight control.

- Light points that do not have a light point type defined are not automatically placed under daylight control.
- Light points in an OpenFlight file that are loaded as an articulated model use the daylightControlledLights model definition parameter.
- Light points with a type defined use the Daylight Controlled parameter for their type in *./appData/importConfig/Lights_DefaultValues_3_2.csv*.

For more information about configuring light points, please see "Configuring Light Points," on page 35-16.

Light sources (lights that cast light onto other objects) are discovered on the cull pass of the scene graph. Sources found under the terrain root or props root of the scene graph are automatically put under daylight control.

13.4. Displaying Ocean Planar Reflections

VR-Vantage can show the reflection of terrain and objects on water (Figure 13-3). To see planar reflections, dynamic ocean must be enabled. (For details about enabling dynamic ocean, please see "Enabling Marine Effects," on page 11-13.)

Displaying ocean planar reflections degrades performance.



Figure 13-3. Ocean planar reflection

To enable or disable planar reflections:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Ocean Settings page (Figure 13-4).

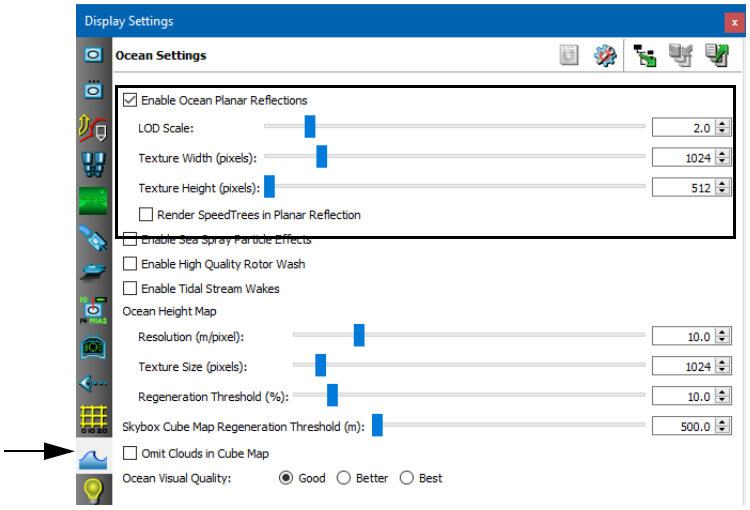


Figure 13-4. Ocean Settings

3. Select or clear the Enable Ocean Planar Reflections check box.

13.4.1. Configuring Ocean Planar Reflections

To reduce the performance impact of ocean planar reflections, you can configure the LOD scale. The LOD scale affects the distance from the observer at which VR-Vantage switches LODs. The higher the value, the more quickly LODs are switched.

The Texture Height and Texture Width parameters affect the resolution of the ocean planar reflection texture, in pixels. Higher resolution decreases performance.

You can also enable and disable display of SpeedTrees as planar reflections.

To configure ocean planar reflections:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Ocean Settings page (Figure 13-4).
3. To change the ocean planar reflection LOD scale, adjust the LOD Scale slider under Enable Ocean Planar Reflections, or change the value in the box.
4. To change the width or height of the ocean planar reflection texture, adjust the Texture Width or Texture Height slider under Enable Ocean Planar Reflections, or change the values in the boxes.
5. To include SpeedTrees in ocean planar reflections, select the Render SpeedTrees in Planar Reflections check box.

13.5. Lens Flare

Lens flare is a visual artifact often seen when looking at light sources such as the sun through a camera lens (Figure 13-5). VR-Vantage can simulate lens flare when the observer looks towards the sun.



Figure 13-5. Lens flare

Lens flare is supported only when high dynamic range lighting is enabled.

To enable or disable lens flare:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the High Dynamic Range Settings page (Figure 13-6).

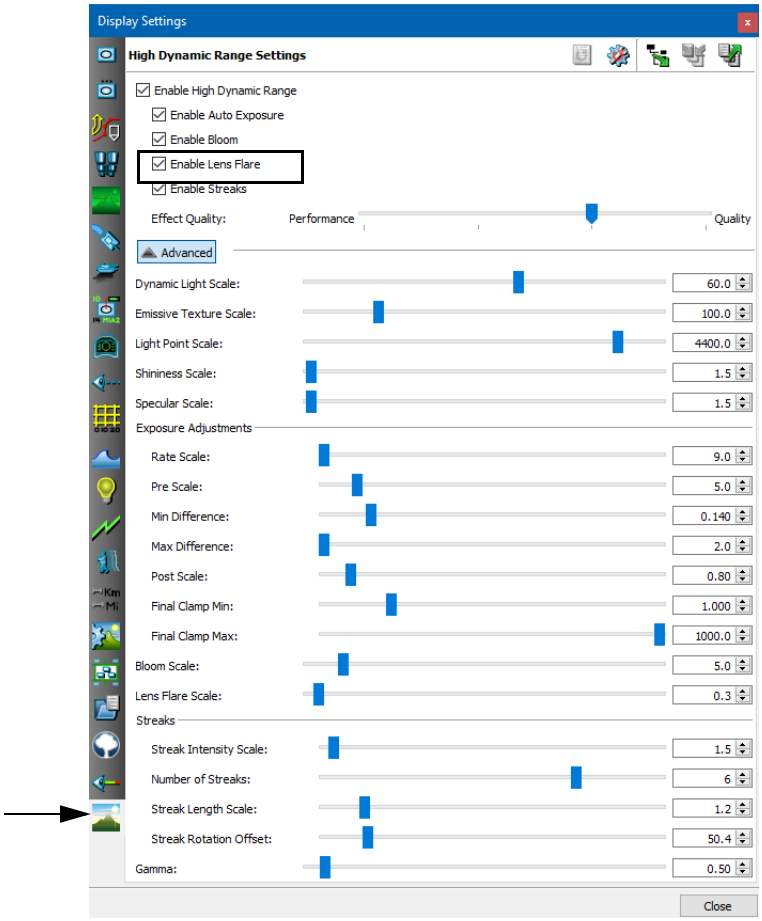


Figure 13-6. High Dynamic Range Settings page

3. Select or clear the Enable Lens Flare check box.

13.6. Displaying Crepuscular (Sun) Rays

VR-Vantage can simulate the effect of crepuscular rays (sunbeams) beaming through clouds (Figure 13-7).

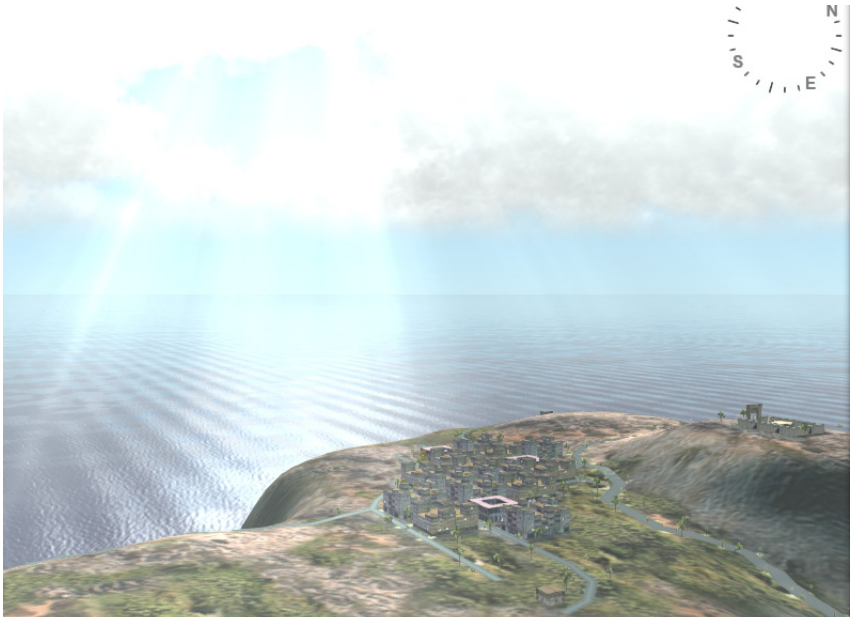


Figure 13-7. Crepuscular rays

To enable or disable crepuscular (sun) rays:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 13-8).

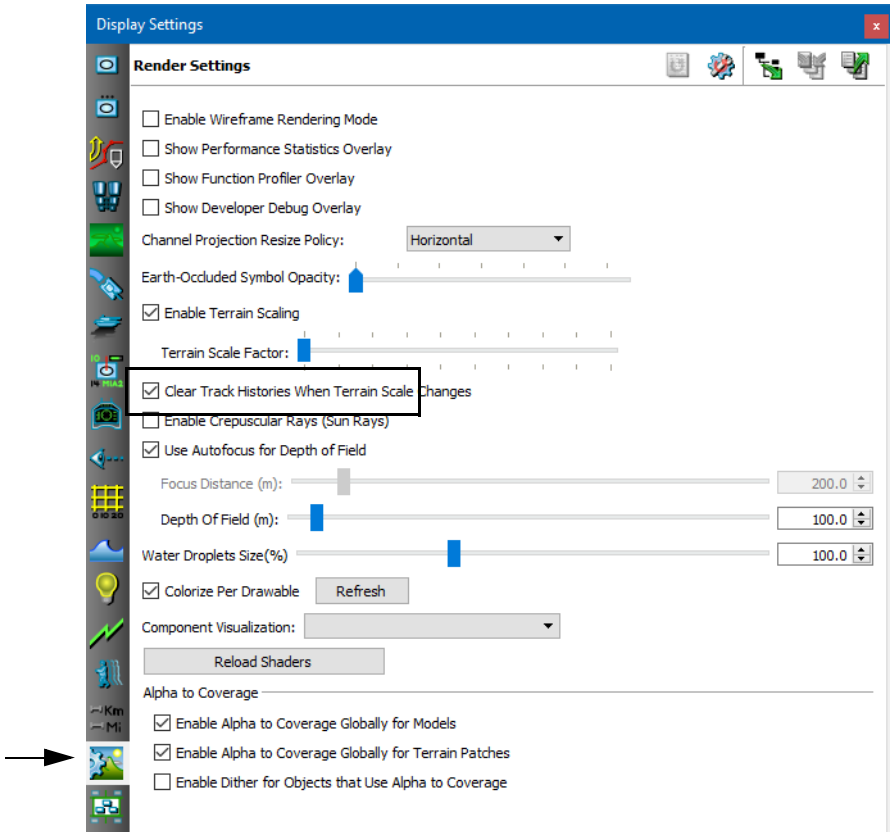


Figure 13-8. Render Settings page

3. Select or clear the Enable Crepuscular Rays (Sun Rays) check box.

13.7. Configuring the Skybox Cube Map

A skybox cube map is a graphics technique used to render reflections and sky effects. The skybox cube map is rendered from the location of the observer. As the observer moves, the cube map needs to be recalculated, which is computationally expensive.

VR-Vantage lets you configure the distance that the observer has to move (the skybox cube map regeneration threshold) before it recalculates the cube map. Increasing the threshold can improve performance, at the cost of visual quality.

You can also choose to omit clouds in the skybox cube map. Omitting clouds can increase performance.

To configure the skybox cube map:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Ocean Settings page (Figure 13-4).
3. Adjust the Skybox Cube Map Regeneration Threshold slider or enter a value in the box.
4. Optionally, select the Omit Clouds in Cube Map check box.

13.8. Flashlight Mode

VR-Vantage can light a circular area as if the observer is using a flashlight. You can move the flashlight beam and increase or decrease the diameter of the flashlight beam.



Figure 13-9. Flashlight mode

- To turn flashlight mode on or off, press **`f`**.
- To move the flashlight beam, hold down **`Alt`** and the right mouse button and move the mouse.
- To change the size of the flashlight beam, hold down **`Alt`** and move the mouse scroll wheel forward or back.

13.9. Using High Dynamic Range Lighting

High dynamic range (HDR) lighting makes scenes look more realistic. They look particularly good on HDR monitors. HDR is enabled by default.

To configure HDR lighting:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the High Dynamic Range Settings page (Figure 13-6).
3. To enable or disable HDR lighting, select or clear the Enable HDR check box.
4. Enable other settings as desired.
5. For those settings that are enabled, adjust their parameters. Table 13-1 describes the HDR settings.

Table 13-1: HDR settings

Setting	Description
Enable HDR	Turn High Dynamic Range (HDR) on/off. When HDR is turned on the scene is rendered in full 16-bits per color floating point and then tone mapped to 8-bits per color before being sent to the display. Additionally, there are post process effects applied to objects that are greater than the display range.
Enable Auto Exposure	Turn on/off auto exposure, which is a post process that scales the intensity of the rendered image based on the intensity of the screen. When auto exposure is turned off the global lighting values are used to scale the intensity to a reasonable value.
Enable Bloom	Turn on/off the bloom post process effect. This effect is a blurring of high intensity objects in the scene which gives a glow around bright objects like light points.
Enable Streaks	Turn on/off the streak post process effect. This effect creates a star effect by streaking the high intensity objects in all directions.
Enable Lens Flare	Turn on/off the lens flare post process effect. This effect creates a lens flare effect for high intensity objects.
Light Point Scale	When HDR is turned on, light point intensity is scaled by this setting.
Dynamic Light Scale	When HDR is turned on, the intensity of dynamic lights are scaled by this setting.
Emissive Texture Scale	When HDR is turned on, the intensity of the emissive textures are scaled by this setting.
Specular Scale	When HDR is turned on the lighting specular component is scaled by this setting.

Table 13-1: HDR settings

Setting	Description
Shininess Scale	When HDR is turned on the lighting shininess component is scaled by this setting. Exposure adjustments are all related to how the intensity of the scene is scaled before being tone mapped. These values are in order based on when they are applied to the scalar.
Rate Scale	When Auto Exposure is enabled, this scales the time it takes for the auto exposure to adjust to changes in the scene. Increase this value to reduce the time it takes for the auto exposure to adjust.
Pre Scale	When Auto Exposure is enabled, this scales the calculated intensity for the scene. Increase this value to reduce the overall intensity of the final image.
Min Difference	When Auto Exposure is enabled, this clamps the minimum value of the calculated intensity to a portion of the global lighting value. This is used to prevent the auto exposure from going too low compared to the global lighting value.
Max Difference	When Auto Exposure is enabled, this clamps the maximum value of the calculated intensity to a portion of the global lighting value. This is used to prevent the auto exposure from going too high compared to the global lighting value.
Post Scale	Scales the scalar after all Auto Exposure calculations have been done. This also affects the global lighting value when Auto Exposure is turned off. Increase this value to reduce the overall intensity of the final image.
Final Clamp Min	Minimum value for the final value that will be used to scale the intensity prior to tone mapping.
Final Clamp Min	Maximum value the final value that will be used to scale the intensity prior to tone mapping.
Bloom Scale	Scales the bloom before it is applied to the scene. Increase this value to increase the intensity of the bloom effect on high intensity objects.
Streak Intensity Scale	Scales the streaks before they are applied to the scene. Increase this value to increase the intensity of the streak effect on high intensity objects.
Number of Streaks	Number of streaks that the streak effect will create for each high intensity object. Maximum of 8 streaks.
Streak Length Scale	Scales the length of the streak effect. Increase this value to increase the length of the streak effect.
Streak Rotation Offset	Rotates the streak effect so that it changes the direction that the streaks point.

Table 13-1: HDR settings

Setting	Description
Lens Flare Scale	Scales the lens flare effect before it is applied to the scene. Increase this value to increase the intensity of the lens flare effect.
Gamma	Final color correction used to adjust the image based on the display. Change this value based on the characteristics of the display being used.

13.10. Displaying Shadows (3D Only)

VR-Vantage can display shadows for entities, props, clouds, and terrain features (Figure 13-10).



Figure 13-10. Shadows

You can enable and disable shadows and you can configure shadow quality.

Enabling shadows can affect performance.

To enable and disable shadows globally:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Shadow Settings page (Figure 13-11).

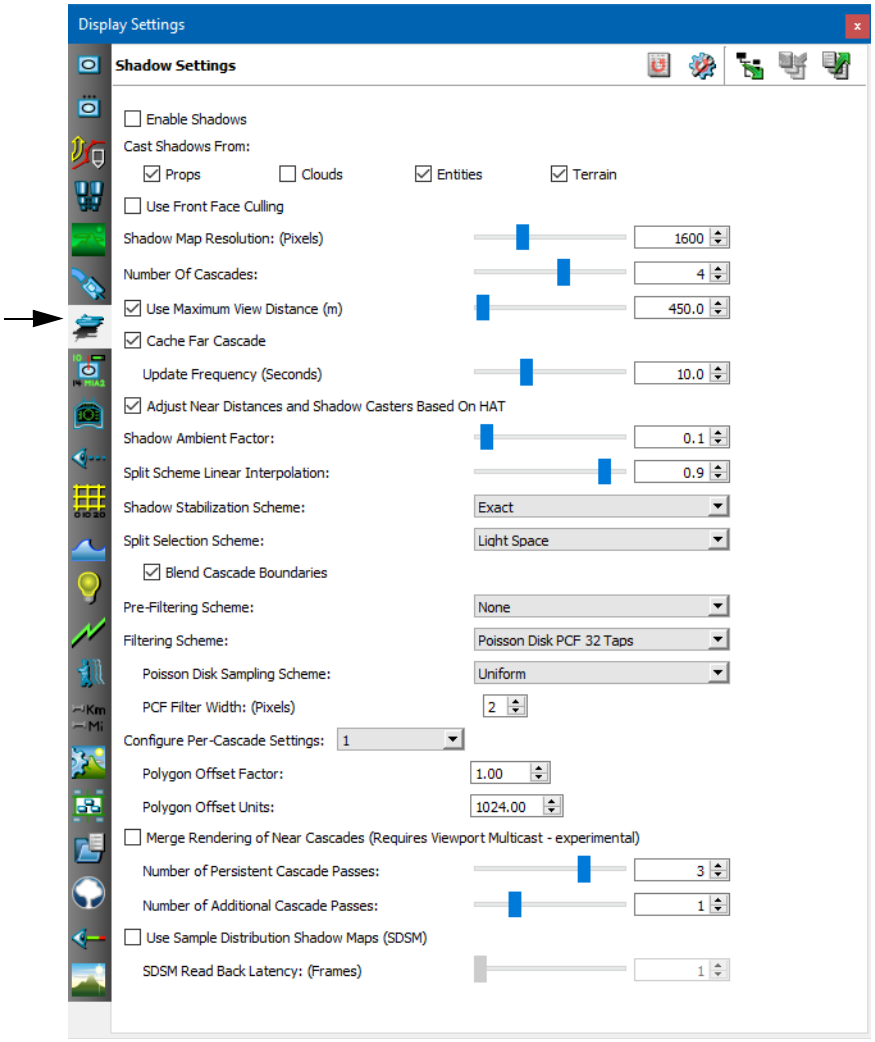


Figure 13-11. Display Settings, Shadow Settings page

3. Select or clear the Enable Shadows check box.
4. To enable or disable shadows for specific object types, select or clear the check boxes under Cast Shadows From.

You can also enable and disable shadows by choosing **Settings** → **Shadows** or clicking the Shadows button ( ) on the Display Settings Toolbar.

13.10.1. Configuring Shadows

You can configure which objects cast shadows and parameters that affect performance and the quality of shadows.

Shadows are drawn based on a shadow map. A shadow map determines what parts of a scene are in shadow relative to a light source. Then the scene is drawn showing the shadows that would be visible from the viewpoint of the observer.

Scenes are split into sections, called cascades. The cascades that are closest to the camera show the highest resolution shadows. Many of the settings control the size of cascades and how they merge together.

For an in-depth understanding of how shadows are rendered in 3D scenes, please see the many articles available on the world wide web.

To configure how shadows are rendered:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Shadow Settings page (Figure 13-11).
3. Change other parameters as desired. Table 13-2 briefly describes them.

Table 13-2: Shadow settings

Setting	Description
Use Front Face Culling	Use of face front culling is a choice between more shadow acne (an effect similar to z-fighting) and more light bleeding artifacts.
Shadow Map Resolution	Increasing shadow map resolution provides more detail at the cost of performance and increased memory use.
Number of Cascades	The more cascades there are, the more detail is rendered, at the expense of memory usage and performance.
Use Maximum View Distance	If enabled, specifies how far away shadows are displayed. The smaller the area in which shadows are displayed, the more quickly they get displayed and the higher the quality. If disabled, uses the scene values.
Cache Far Cascade	Caches the farthest cascade and updates it less frequently than the near cascade, which increases performance.
Adjust Near Distances and Shadow Casters Based On HAT	As the observer moves higher above the terrain, the near cascade shrinks and only entities are rendered to them. The far cascade becomes responsible for the terrain shadows.
Shadow Ambient Factor	Fades out shadows. It determines how much light gets through.
Split Scheme Linear Interpolation	Adjusts how cascades are distributed between a logarithmic approach and a linear approach.

Table 13-2: Shadow settings

Setting	Description
Shadow Stabilization Scheme	Different ways to reduce the shimmer that is common in shadow maps.
Split Selection Scheme	Different ways to choose which cascade a piece of geometry is in.
Pre-Filtering Scheme	Offers different shadow casing techniques to try to smooth shadows.
Filtering Scheme	Affects how blurred the shadow map is.
Configure Per-Cascade Settings	Lets you configure the polygon offset for cascades.
Merge Rendering of Near Cascades	Lets you draw cascades all at once instead of individually. This improves performance. However, it is only available on newer video cards (NVidia 980 or newer). If disabled, you can configure the number of cascades passes to draw shadows.
Use Sample Distribution Shadow Map (SDSM)	Adjusts quality based on the depth complexity of an area of the scene.



IV. The Observer

The observer, or eyepoint, is the location in the 3D or 2D environment from which you observe the scene. You can move the observer (navigate) through the scene. You can attach the observer to a simulation object or prop. When the observer is attached to a simulation object, it moves automatically with that object. You can also control the observer from a remote application using view control messages.

You can have multiple observers in a VR-Vantage session. They might be associated with different windows. For example you could have an observer for the main window and one for an inset window.



14. The Observer and Observer Modes

This chapter explains how to create, edit, and change the observer mode. For details about moving the observer, please see Chapter 15, *Moving the Observer*. For details about attaching the observer to entities and props, please see Chapter 16, *Attaching the Observer to Objects*.

Switching Between the 2D and 3D Projections 14-2
Observer Modes 14-2
Changing the Observer Mode 14-3
Creating and Editing Observer Modes 14-4
- Creating an Observer Mode 14-4
- Deleting an Observer Mode 14-5
- Editing an Observer Mode 14-5
- Renaming an Observer Mode 14-5

Adding an Observer 14-6
- Adding an Observer on the Observer Settings Page 14-6
- Adding an Observer in the Display Engine Configuration Editor . 14-7

Removing an Observer 14-9
Communicating Information About the Observer 14-9
- Making the Observer Visible to Other Applications 14-10
- Viewing and Attaching to Other Observers 14-11

14.1. Switching Between the 2D and 3D Projections

VR-Vantage Stealth supports both 2D and 3D projections. You can switch between them simply by changing the observer mode from Stealth or Out-the-Window, to Plan View. (Changing observer modes is described in "Changing the Observer Mode," on page 14-3.) When you go from a 3D observer mode to 2D, VR-Vantage calculates a zoom level so that the observer should appear as if it is roughly the same height above the terrain. When you go from a 2D observer mode to 3D, VR-Vantage calculates an altitude to do the same thing, and keeps the 3D view pointing down (so the change in overall view stays fairly small).

14.2. Observer Modes

Observer modes let you configure VR-Vantage settings that affect how the observer moves and what it sees. You can then quickly change from one group of settings to another by switching between observer modes. An observer mode specifies:

- A model set. A collection of models of a particular type, such as 3D models.
- A key map. A coordinated set of key mappings to movement functions.
- A projection (2D or 3D).
- Settings for simulation object graphics and observer movement.

VR-Vantage includes the following observer modes:

- Stealth. Stealth observer mode is a 3D projection configured for situational awareness.
- Out-the-Window. Out-the-Window observer mode is configured for when you use VR-Vantage IG or if you want to use VR-Vantage Stealth as a desktop image generator. It hides many of the explanatory visual effects present in the Stealth observer mode.
- Plan View. The Plan View observer mode provides a 2D "big picture" view of the terrain.
- XR. XR, or exaggerated reality, observer mode is a 3D mode that helps you understand the relationships between entities and the terrain when the observer is too far away from entities to view them at realistic sizes. It allows you to exaggerate entity size and terrain.
- IR BlackHot (CameraFX). A view using the CameraFX IR (infrared) BlackHot sensor.
- IR (CameraFX). A view using the CameraFX IR sensor.
- NVG (CameraFX). A view using the CameraFX NVG (night vision goggles) sensor.
- Camera (CameraFX). A view using the Visual Camera CameraFX sensor.

You can edit observer modes and add new ones. For details, please see "Creating and Editing Observer Modes," on page 14-4.

14.3. Changing the Observer Mode

When you change the observer mode, terrain may be displayed differently and the model sets used to display simulation objects may change. For descriptions of the observer modes, please see "Observer Modes," on page 14-2.

To enable marine effects (dynamic ocean):

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 11-12).

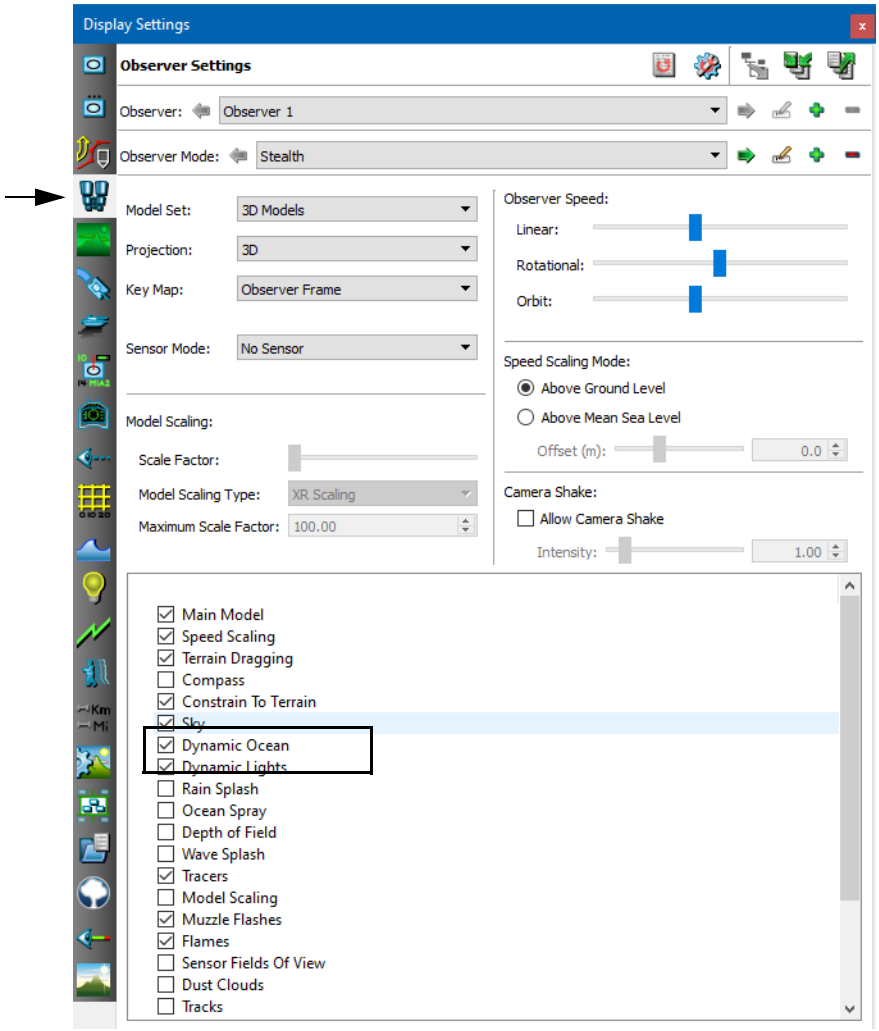


Figure 14-1. Observer Settings page

3. Select an option in the Observer Mode list.



You can also change the observer mode by choosing **Observer** → **Set Observer Mode** → *mode*, or on the Observer Settings Toolbar, choose an option from the Observer Mode list.

14.4. Creating and Editing Observer Modes

VR-Vantage comes with 3D and 2D observer modes. (Table 15-1 lists the observer modes available for each VR-Vantage application.) They are configured with the display settings, key mappings, and model sets that users most commonly want to use. However, if they do not meet your needs, you can easily edit them or create new observer modes.

14.4.1. Creating an Observer Mode

To create an observer mode:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 14-2).

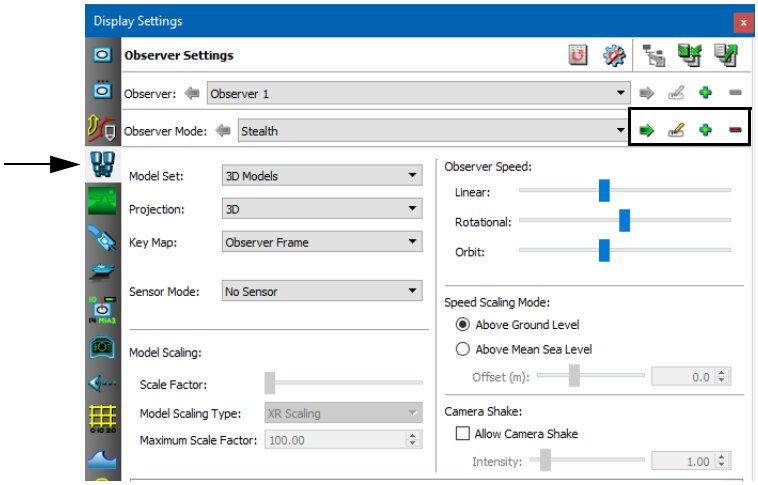


Figure 14-2. Observer Settings page

3. Click the Add button (+) to the right of the Observer Mode list. The Create New Observer Mode dialog box opens (Figure 14-3).

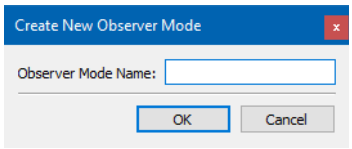


Figure 14-3. Create New Observer Mode dialog box

4. Type a name for the new observer mode.

5. Click OK.
6. In the Model Set list, select a model set. (VR-Vantage Stealth only.)
7. Select a projection (2D or 3D). (VR-Vantage Stealth only.)
8. Select the key map you want to use.
9. Optionally, specify a sensor mode.
10. Optionally, change the observer speed scaling settings and speed scaling mode.
11. Optionally, for 3D model sets, change the Model Scaling settings.
12. Enable or disable the mode-specific settings. The list of settings varies depending on the projection, model set, and application.

14.4.2. Deleting an Observer Mode

To delete an observer mode:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 14-2).
3. In the Observer Mode list, select the observer mode that you want to delete.
4. Click the Delete button ( ).

14.4.3. Editing an Observer Mode

To edit an observer mode:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 14-2).
3. In the Observer Mode list, select the observer mode that you want to edit.
4. Edit the attributes of the observer mode as desired.

14.4.4. Renaming an Observer Mode

To rename an observer mode:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 14-2).
3. In the Observer Mode list, select the observer mode that you want to rename.
4. Click the Rename button ( ). The Rename Observer Mode dialog box opens.
5. Type the new name.
6. Click OK.

14.5. Adding an Observer

You can add and remove observers. Having multiple observers lets you assign different observers to different windows or channels. Changing the position and orientation of the observer in one channel does not affect the view of a different observer in a different channel. (For information about windows and channels, please see Chapter 6, *Windows, Channels, and Display Engines*.)



If you have multiple observers, the observer listed in the Observer Settings Toolbar is the observer assigned to the window that has focus. If you open the Display Engine Configuration Editor Panel and select a different observer in the Channel Attributes list, the data in relevant panels will change to reflect the values for the selected observer. However, as soon as you return focus to a window, the observer selection will change to match that of the window.

You can add observers in the Display Engine Configuration Editor Panel or on the Observer Settings page. (For information about the Display Configuration Editor Panel, please see Chapter 6, *Windows, Channels, and Display Engines*.)

If you want to save your observers, save the display engine configuration in which you created them. (Saving an observer just saves its name. It does not save any information about a particular observer mode or location that you may have used for that observer. To save location information, save an observer view.)

14.5.1. Adding an Observer on the Observer Settings Page

To add an observer on the Observer Settings page:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 14-2).
3. Click the Add button () to the right of the Observer list. The Create New Observer dialog box opens.
4. Type a name for the new observer.
5. Click OK. The new observer is added to the list.
6. To save the observer, save the display configuration. (Saving the observer settings does not save observers.) (For information saving a display configuration, please see "Saving a Display Engine Configuration," on page 6-8.)

14.5.2. Adding an Observer in the Display Engine Configuration Editor

To add an observer on the Display Engine Configuration Editor Panel:

1. Choose **View** → **Display Engine Configuration Editor Panel**. The Display Engine Configuration Editor Panel opens (Figure 14-4).

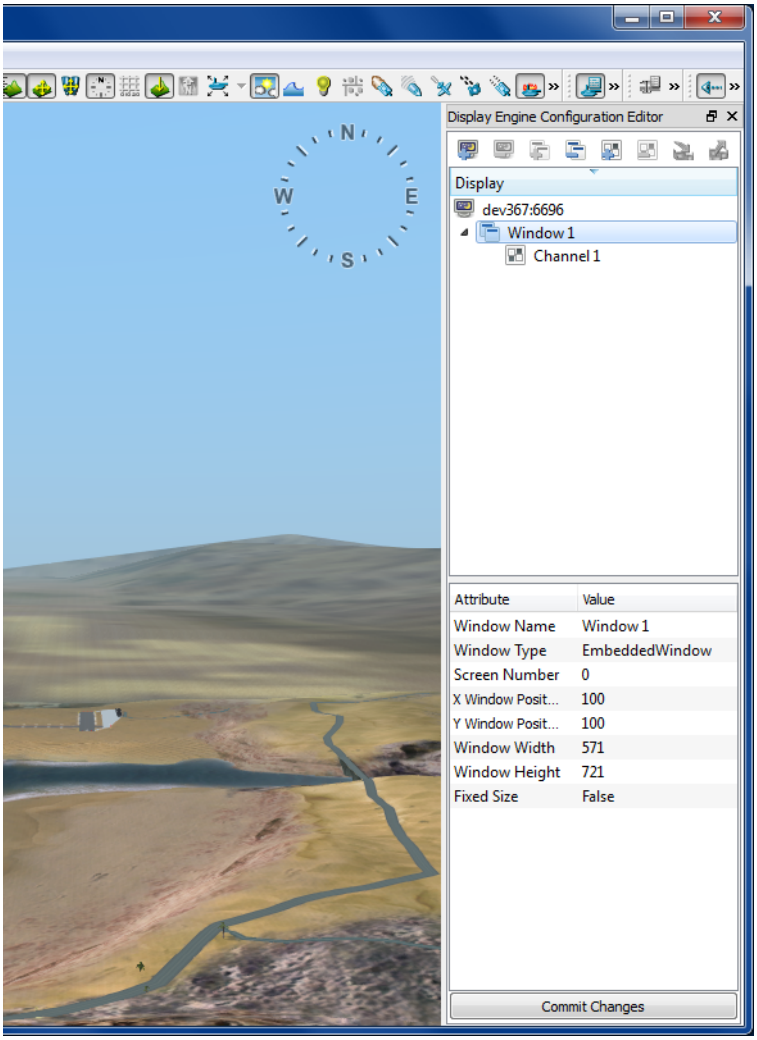


Figure 14-4. Display Engine Configuration Editor Panel

2. Select the channel on which you want to add an observer.
3. Click the Value field for the Observer Name attribute to make it editable.
4. In the list, select New (Figure 14-5).

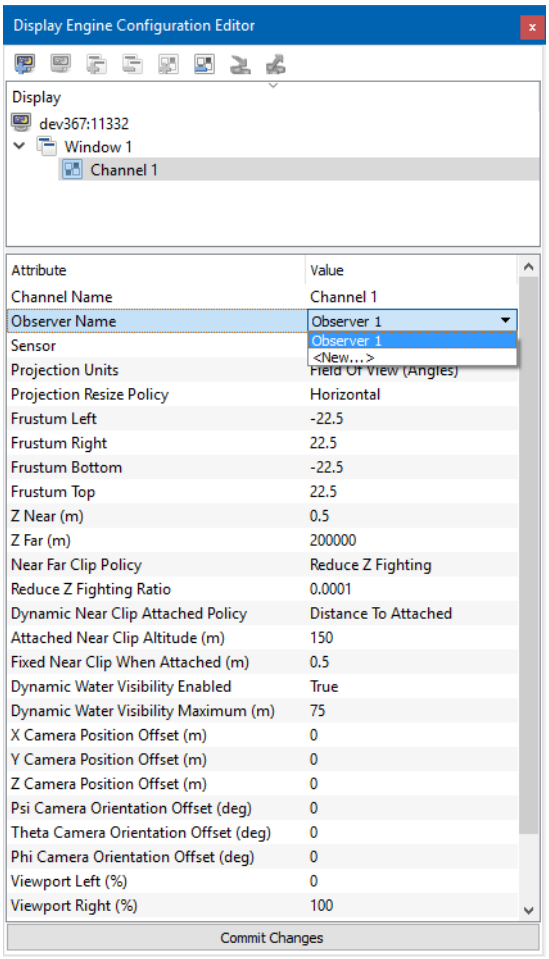


Figure 14-5. Observer Name attribute

5. Click Commit Changes. The Add Observer dialog box opens.
6. Type a name for the new observer.
7. Click OK. The new observer becomes the selected observer on the Observer Settings Toolbar, and is listed in the Observer Name attribute. (You may need to refresh the channel attributes before the observer name is listed.)
8. To save the observer, save the display configuration. (For information about saving a display configuration, please see "Saving a Display Engine Configuration," on page 6-8.)

14.6. Removing an Observer

When you close VR-Vantage, all observers get removed. They are not saved unless you save a display configuration. You can also remove observers during a session.

To remove an observer:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 14-2).
3. In the Observer list, select the observer that you want to remove.
4. Click the Delete button (–). The observer is removed.
5. If the observer was saved as part of a display configuration, save the updated display configuration.

14.7. Communicating Information About the Observer

VR-Vantage can transmit state updates describing the observer's position and orientation, and can respond to view control messages sent by other applications.

14.7.1. Making the Observer Visible to Other Applications

By default, VR-Vantage sends state update messages to make its state information known to other applications. This makes it possible for a plan view display to represent VR-Vantage as an icon, or for one VR-Vantage application to synchronize its observer movement with that of another VR-Vantage application.

To enable or disable broadcasting of observer information:

1. Choose **Settings** → **Connection**. The Simulation Connections dialog box opens (Figure 14-6).

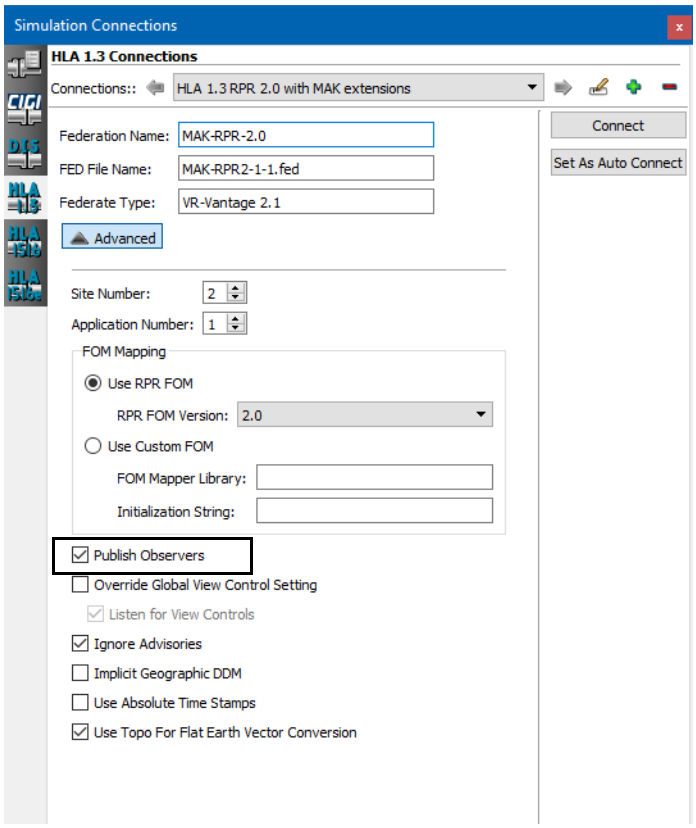


Figure 14-6. HLA 1.3 Connections page

2. Select the page for the connection that you want to configure.
3. Select or clear the Publish Observers check box.
4. Click Close.

Making the Observer Visible to Just One Application (DIS Only)

In DIS you can send observer information to just one other application, rather than to everyone in the exercise. To do this, on the DIS Connections page, instead of using a broadcast address, set the destination address to the address of a specific computer.

14.7.2. Viewing and Attaching to Other Observers

If an exercise includes other stealth applications, and they are sending state updates, VR-Vantage can detect them. A remote observer entity is listed in the object hierarchy, so you can attach the local observer to it just as you would attach to any other simulation object.

Observer icons are not normally visible in the VR-Vantage window, even when they appear in the entity list. If you want to display visual representations of remote observers, map a visual model to the observer's entity enumeration.

For related information, please see "Displaying Models for Remote Observers," on page 19-16.



15. Moving the Observer

This chapter explains how to move the observer (eyepoint) in the VR-Vantage window.

Moving the Observer 15-3
Using the Compass to View the Observer's Heading 15-4
Editing the Compass's Attributes 15-5
Observer Movement Functions and Keyboard Mappings 15-6
3D Observer Movement Coordinate Systems 15-7
2D Observer Coordinate System 15-7
The Attached Context and Observer Movement (3D Only) 15-8
Moving the Observer from the Keyboard in the 3D Projection 15-10
Navigation Options 15-12
Orbiting 15-13
Moving the Camera 15-13
Moving the Observer Using the Mouse (3D Projection) 15-14
Dragging the Terrain 15-14
Changing the Observer's Orientation with the Mouse 15-16
Dragging the View 15-16
Orbiting the Terrain with the Mouse 15-16
Teleporting the Observer 15-17
Moving the Observer in the 2D Projection 15-17
Moving the Observer from the Keyboard in the 2D Projection 15-18
Moving the Observer Using the Mouse in the 2D Projection 15-19
Zooming the Observer 15-19
Zooming the Observer in the 2D Projection 15-19
Zooming the Observer in the 3D Projection 15-20
Setting the Scale Factor for Paged LODs 15-21
Setting the Magnification LOD Angle for Paged LODs 15-22
Simulating Camera Jitter 15-23
Forcing Orientation North (Plan View Mode Only) 15-24

Enabling and Disabling View Constraints (3D Only) 15-25
Controlling Navigation Speed 15-25
Enabling or Disabling Speed Scaling 15-26
Changing the Observer's Movement Speed 15-27
Saving and Recalling Views 15-28
Recalling a View 15-29
Deleting Views 15-30
Loading Views Automatically When You Load a Terrain 15-30
Saving Views to a File 15-31
Loading Views 15-32
Controlling the Observer from Other Applications 15-32
Enabling the Processing of View Control Messages 15-32
Overriding the Global View Control Settings 15-33

15.1. Moving the Observer

When you view a simulation, you usually want to move the observer (eyepoint) to different locations to view the terrain and scene elements being simulated. In VR-Vantage, you can move the observer using the keyboard and the mouse. There are two basic contexts in which you can move the observer:

- You can move it independently of any scene element. This is called absolute, or free-fly mode.
- You can attach the observer to a simulation object, in which case it moves with the object. (You can also attach the observer to a prop. For details about attaching the observer to a simulation object or prop, please see Chapter 16, *Attaching the Observer to Objects*.)

Absolute mode and the various attachment types are collectively called attach modes.

Regardless of which attach mode you are using, you can change the position and orientation of the observer. In the 3D projection, position is defined by X, Y, and Z coordinates that identify its location in the 3D world. Orientation describes the observer's yaw (heading), pitch, and roll (Figure 15-1).

In the 2D projection, the Z coordinate is fixed. You can change the X and Y coordinates. You can change the observer's yaw, but pitch and roll are fixed.

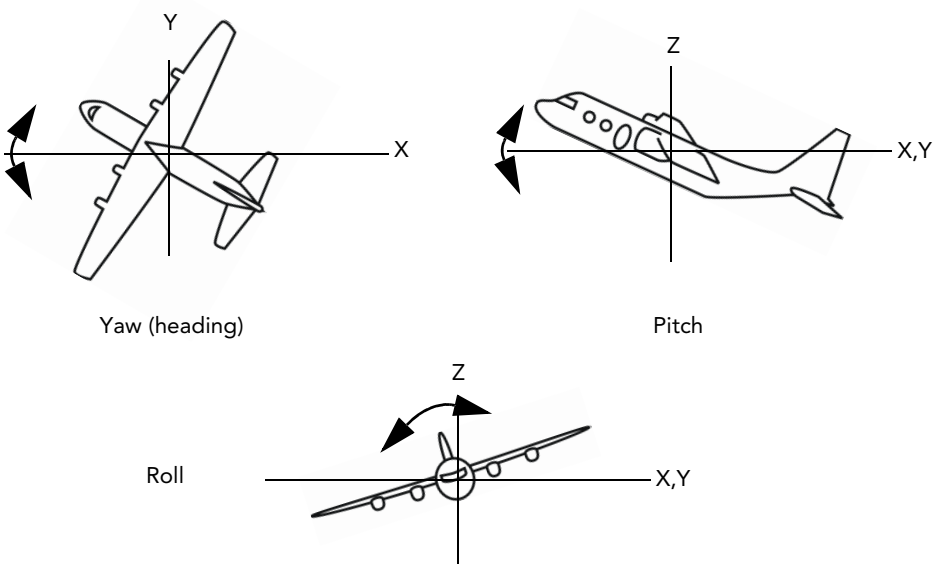


Figure 15-1. Orientation

The Observer Information Panel (Figure 15-4) displays the location and orientation of the observer in whichever coordinate system you select.

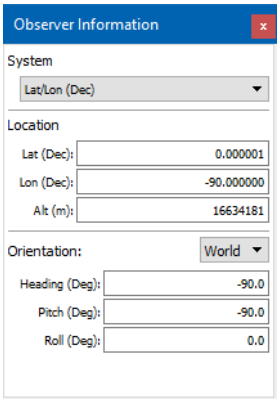


Figure 15-2. Observer Information Panel

15.1.1. Using the Compass to View the Observer's Heading

VR-Vantage can display a compass that always shows the heading of the observer. The compass is an observer mode setting. In Figure 15-3, the observer's heading is approximately Northwest.

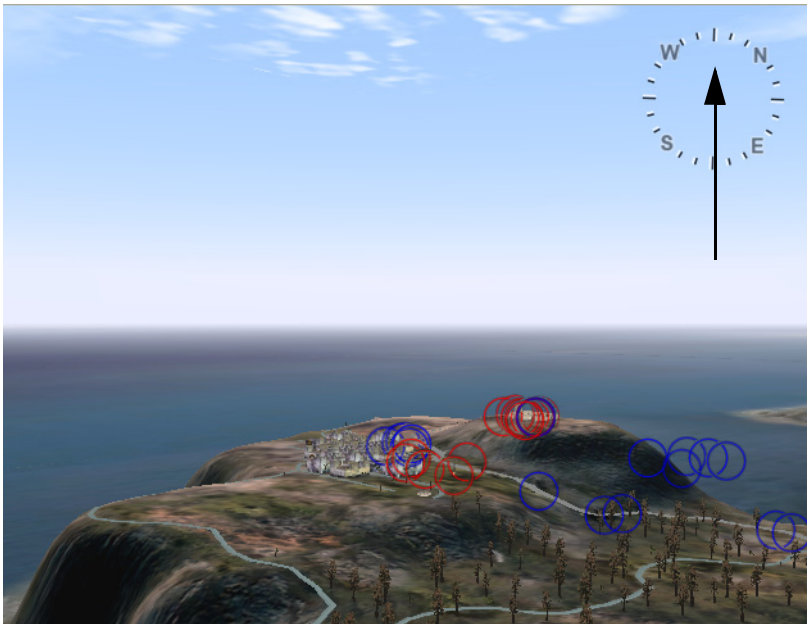


Figure 15-3. Compass

To enable or disable the compass:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 14-2).
3. Select the observer mode for which you want to change the compass setting.
4. In the list of observer settings, select or clear Compass.



You can also enable or disable the compass by choosing **Observer** → **Compass** or clicking the Compass button () on the Observer Settings Toolbar.

15.1.2. Editing the Compass's Attributes

You can configure the compass by editing its schema (ChannelCompass) and model definition (smallCompass and largeCompass). (For details about editing schemas, please see "Creating and Editing Schemas," on page 31-2.)

By default, all channels have the largeCompass keyword. To change the style of compass used for a channel, change the keyword in the Keywords attribute for the channel. To completely remove the compass, delete the keyword. For details about setting channel attributes, please see "Changing a Channel's Attributes," on page 6-12.

You can change where the compass is displayed in the window by editing the location attribute. The permissible values are top, topRight, right, bottomRight, bottom, bottomLeft, left, and topLeft.

You can change the color of the compass by editing the darkTickColor and lightTickColor attributes. The color is defined using the OpenGL scheme for specifying RGBA. The first three values are the RGB color values. The fourth value specifies opacity. Each value can be from 0.0 through 1.0.

15.2. Observer Movement Functions and Keyboard Mappings

When you move the observer, you are executing observer movement functions. An observer movement function is a specific way that the observer can move, such as Move Observer Forward or Move Level Observer Forward. (These functions both move the observer forward, but they use different observer coordinate systems. Observer movement coordinate systems are described in "3D Observer Movement Coordinate Systems," on page 15-7.)

The various movement functions are mapped to the keys on the keyboard and to mouse buttons and the mouse wheel. If you use the default key mappings for the default observer modes, then most of the time, the observer will behave exactly as you would expect it to. However for some types of movement, such as forward and back, the observer can move differently depending on the projection, the observer coordinate system, or the attach mode. Therefore, VR-Vantage provides several key mappings that let you choose which type of observer movement to use. You can change these key mappings and create your own key mappings. (For details about editing key maps, please see Chapter 46, *Creating and Editing Key Mappings*.) This section describes how observer movement can vary based on the observer coordinate system and the attach mode.

Table 15-1 lists the observer modes and key map settings provided with VR-Vantage applications.

Table 15-1: Observer modes

Application	Observer Mode	Projection	Key Map
VR-Vantage IG	Out-the-Window CameraFX modes	3D	Observer Frame
VR-Vantage Stealth	Stealth CameraFX modes	3D	Observer Frame
	XR	3D	Observer Frame
	Plan View	2D	2D Level Observer Frame
VR-Vantage PVD	Plan View	2D	2D Level Observer Frame

15.2.1. 3D Observer Movement Coordinate Systems

In the 3D projection, the observer moves in the context of one of the following coordinate systems:

- Level Observer Coordinates: Movement is with respect to the terrain. The level observer coordinate system is defined by the current up vector (Z), and a plane normal to that vector into which the observer's X and Y axes are projected. (Essentially, this means that the observer's heading is used while its pitch and roll are considered to be 0.0.)



Movement is always with respect to a plane tangent to the earth's surface. However, in a projected terrain database, such as UTM, Lambert Conformal Conic, or flat earth, the plane is fixed, whereas in a geocentric terrain database the up vector (and hence the tangent plan) changes as the observer's position changes, so movement follows the curvature of the earth.

- Observer Coordinates: Movement is with respect to the observer's body coordinate system.
- Entity Level Body Frame Coordinates: Movement is with respect to an entity's body frame, but ignores pitch and roll.
- Entity Body Frame Coordinates: Movement is with respect to an entity's body frame, including pitch and roll.

The practical difference between observer coordinates and level observer coordinates is that in level observer coordinates, if you are looking down at the terrain, when you move forward and backward, you fly over the terrain, looking down. In observer coordinates, when you move forward and backward, you move down towards the terrain or away from it at whatever angle you are oriented towards it.

15.2.2. 2D Observer Coordinate System

When you use the 2D projection (VR-Vantage Stealth and PVD), the observer uses a variation of level observer coordinates. Movement is restricted to two dimensions, so there are no movement functions for changing the altitude, pitch, or roll. You can zoom in and out in the 2D projection, but this does not change the observer's altitude, it changes the field of view and resolution of the terrain image.

15.2.3. The Attached Context and Observer Movement (3D Only)

The key maps control navigation using a particular coordinate system (observer frame or level observer frame). They also support navigation in alternate coordinate systems. The alternate system varies depending on the attached context of the observer. It represents MAK's opinion of which alternate coordinate system would be most useful given the attach mode that the observer is using.

The Move Attached Context *direction* (forward, back, up, down, left, right) functions move the observer in the context of its attach mode. Movement is modified by attach modes in the following ways:

- Follow mode: movement is in the entity's level body frame.
- Track mode: movement is in the entity's level body frame.
- Mimic mode: movement is in the entity's body frame.
- Tether mode: movement is in level observer frame.
- Unattached: movement is in level observer coordinates.

Figure 15-4 and Figure 15-5 illustrate the differences between moving the observer in an attached context and without an attached context.

Figure 15-4 illustrates movement in entity level body coordinates (Follow and Track modes). The Move Observer Left function (**A** in Observer Frame key map) moves the observer to the left in observer coordinates, that is left relative to the observer's heading. The Move Attached Context Left function (**Shift** + **A**) moves the observer left in the attached context. In entity level body frame, this means the observer moves left relative to the entity's heading.

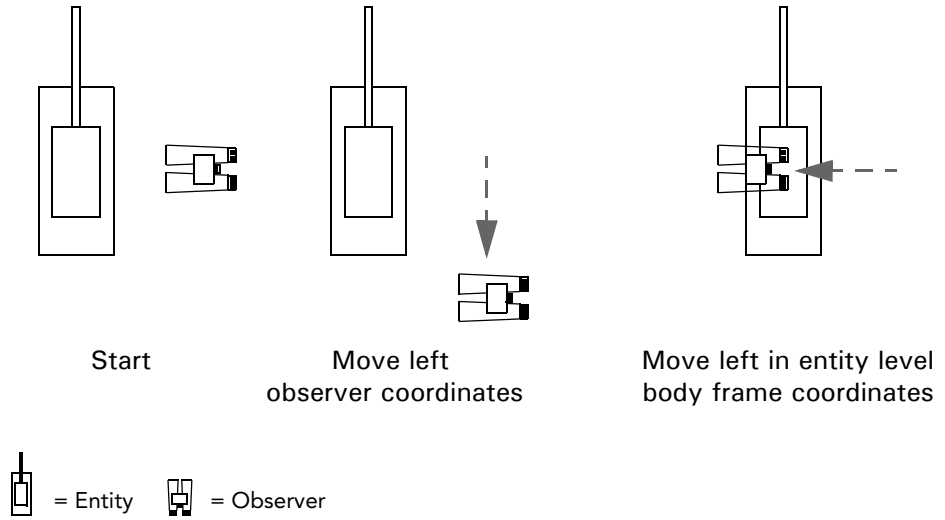


Figure 15-4. Observer movement in entity level body frame coordinates

Figure 15-5 illustrates movement in entity body frame (Mimic mode). The Move Observer Left function moves the observer to the left in observer coordinates, that is left relative to the observer's heading, but ignoring pitch and roll. The Move Attached Context Left function moves the observer left in the attached context. In entity body frame, this means the observer moves left relative to the entity's heading and its pitch and roll.

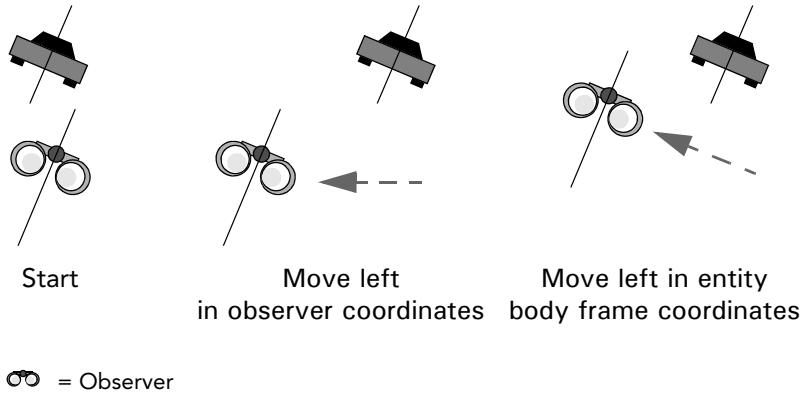


Figure 15-5. Movement in entity body frame coordinates

15.3. Moving the Observer from the Keyboard in the 3D Projection

VR-Vantage maps the observer movement functions to the keyboard using key maps that support the most typical ways that users want to move through the 3D world, either in absolute mode or attached to a simulation object.

! References to movement key combinations use the key mappings provided with VR-Vantage. If you change these key mappings, the instructions may no longer be valid.

As described in "Moving the Observer," on page 15-3, the observer typically moves with respect to the terrain in level observer coordinates or observer coordinates. The corresponding key maps are Level Observer Frame and Observer Frame.

Figure 15-6 illustrates the positional movement keys. Figure 15-7 illustrates the keypad layout for changing orientation and movement speed. For a complete list of movement functions and their keyboard and mouse mappings, please see *VR-Vantage Quick Reference Card* or the Key Mapping Editor.

! To use the keypad for navigation, NumLock must be on.



Figure 15-6. Primary movement keys, observer coordinates (3D)

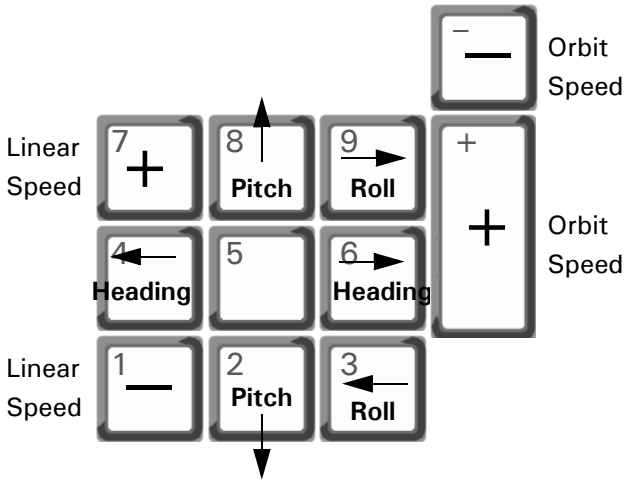


Figure 15-7. Keypad layout, heading, pitch, and roll (3D)

If your keyboard does not have a keypad, as on most laptops, the Laptop key map lets you use the arrow keys to control the left, right, up, and down movement mapped to 4, 6, 8, and 2 on keypads.

➤ To move the observer, press a key combination and hold it for as long as you want the observer to keep moving.

You can press more than one key at a time to combine movement functions. For example, pressing the move forward and move left keys at the same time moves the observer diagonally left and forward.

15.3.1. Navigation Options

This section lists the different ways that you can move the observer using the keyboard. For a complete list of movement functions and their keyboard and mouse mappings, please see *VR-Vantage Quick Reference Card*. You can:

- Move forward and backward (observer, level observer, entity, level entity, and attached context coordinates).
- Move up and down (observer, level observer, entity, level entity, and attached context coordinates).
- Move left and right (observer, level observer, entity, level entity, and attached context coordinates).
- Rotate (heading) left and right (observer and level observer coordinates).
- Rotate (pitch) up and down (observer and level observer coordinates).
- Rotate (roll) left and right (observer and level observer coordinates).
- Orbit left and right.
- Orbit up and down.
- Look (move camera) left and right.
- Look (move camera) up and down.
- Tilt (move camera) left and right.
- Reset the observer to the default position and orientation for the attached mode.
- Reset the camera to the default position and orientation.
- Zoom the view.

For details about orbiting, please see "Orbiting," on page 15-13. For details about moving the camera, please see "Moving the Camera," on page 15-13.

When you view a geocentric terrain and move the observer more than 6 million meters away from the earth's surface, the view is forced to focus toward the center of the earth. The practical effect of this mode is that the left and right movement keys orbit the eyepoint around the earth, rather than moving the entire earth to the left or right.

15.3.2. Orbiting

You can orbit the observer (revolve left, right, up, and down) around an object or the point on the terrain that is at the center of the window. (If the center of the screen is in the sky, orbit does not work.) When you orbit, the observer's heading continuously changes so that it always faces the point or object around which it orbits. You can change the camera to look at something other than the orbit point while you are orbiting.

To orbit around an object:

1. Attach the observer to the object.
2. Press the right or left arrow keys.

To orbit around a point:

1. If you are attached to a simulation object, detach (press **`z`**).
2. Press the right or left arrow keys.

15.3.3. Moving the Camera

By default, the observer always looks in the same direction as its heading. However, just as the turret on a tank can swivel and fire in directions other than its body's heading, the observer can look in a direction other than its heading. The ability to look around is enabled in the camera. You can move the observer's camera up, down, left, right, tilt left, and tilt right. (The default key mappings are **`Ctrl`** + *Keypad number*, where *number* is 2, 3, 4, 6, 8, or 9.)

! To use the keypad for navigation, **`NumLock`** must be on.

---

i You can tell that the camera is moving, but the heading is not, by observing the compass. If you change the heading, the view of the terrain changes and the compass moves. If you move the camera, but keep the heading static, the view of the terrain changes, but the compass does not move.

➤ To move the camera, press the keys mapped to the camera movement functions.

15.4. Moving the Observer Using the Mouse (3D Projection)

You can change the observer's position and orientation with the mouse. (To see the full list of mouse mappings, view the Mouse Mappings page on the Application Settings dialog box.)

Using the mouse, you can:

- Drag the terrain.
- Drag the view.
- Orbit the terrain.
- Change orientation.
- Teleport (jump).

15.4.1. Dragging the Terrain

Dragging the terrain combines the move forward, backward, left, and right functions.

To drag the terrain:

1. Click a point on the terrain and hold down the left mouse button.
2. Drag the mouse left, right, forward, and backward.



Dragging the terrain is most effective when you are moving large distances, such as several kilometers. To move small distances, use the keyboard.

Disabling Terrain Dragging

There may be times when you want to disable terrain dragging.

To enable or disable terrain dragging:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 15-8).

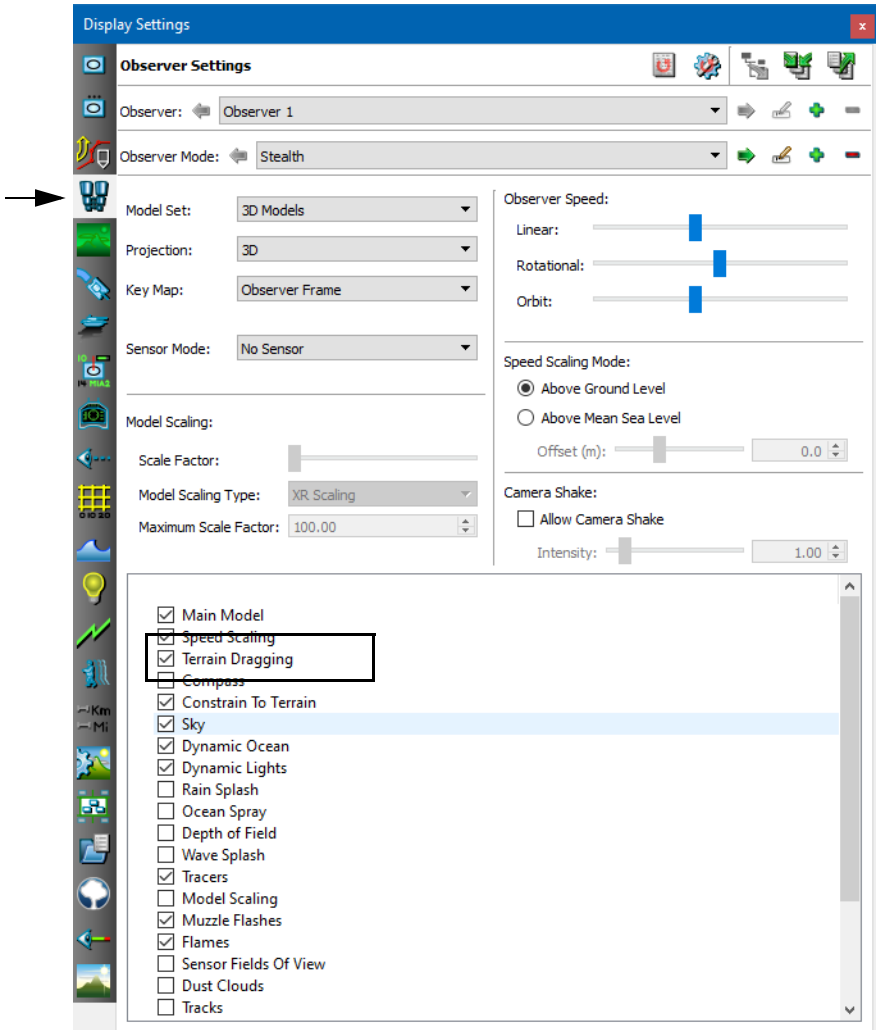


Figure 15-8. Observer Settings page

3. Select or clear the Terrain Dragging check box.

> You can also enable or disable terrain dragging by clearing the Allow Terrain Dragging check boxy on the Observer Control Panel (Figure 15-9), or clicking the Terrain Dragging button ( ) on the Observer Settings toolbar.

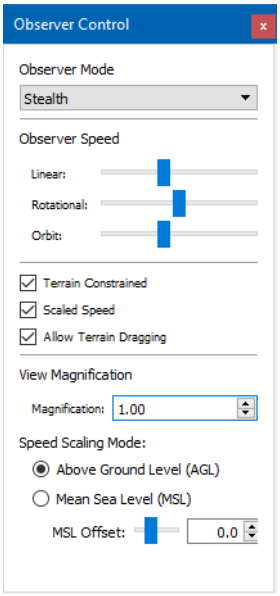


Figure 15-9. Observer Control Panel

15.4.2. Changing the Observer's Orientation with the Mouse

➤ To change the observer's orientation, hold down the right mouse button and drag the mouse. The observer rotates in the direction you drag the mouse.

15.4.3. Dragging the View

Dragging the view combines the observer up, down, left, and right movement functions. Unlike dragging the terrain, which requires you to click a point on the terrain, you can drag the view from any observer location.

➤ To drag the view, hold down the middle mouse button (or wheel) and drag the mouse.

15.4.4. Orbiting the Terrain with the Mouse

➤ To orbit the terrain with the mouse, press **`Ctrl`** and drag the mouse.

15.4.5. Teleporting the Observer

You can teleport the observer to a simulation object, a prop, or a place on the terrain. Teleporting moves the observer 90% of the way from its current position to the location or object clicked on.

To teleport the observer:

- **Shift**+click on the terrain or prop that you want to jump to.



If you **Shift**+click on a filled area, the observer attaches to the area; it does not teleport. If you want to teleport to the terrain under an area, hide the area first. You can hide an area by hiding all tactical graphics or hiding the overlay on which the area is located.

- **Ctrl**+**Shift**+click on the simulation object that you want to jump to.

15.5. Moving the Observer in the 2D Projection

In the 2D projection, you move the observer using the mouse and keyboard just as you do in the 3D projection. Many of the key and button mappings for the 2D projection are the same as those in the 3D projection. However, the 2D mappings do not support movement that is not meaningful in two dimensions, such as up and down (altitude), pitch, and roll.



When Orient North is enabled, you cannot change the heading of the observer. For information about Orient North, please see "Forcing Orientation North (Plan View Mode Only)," on page 15-24.

15.5.1. Moving the Observer from the Keyboard in the 2D Projection

Figure 15-10 illustrates the positional movement keys for the 2D projection. Figure 15-11 illustrates the keypad layout for changing heading and movement speed. For a complete list of movement functions and their keyboard and mouse mappings, please see *VR-Vantage Quick Reference Card*.

! To use the keypad for navigation, NumLock must be on.

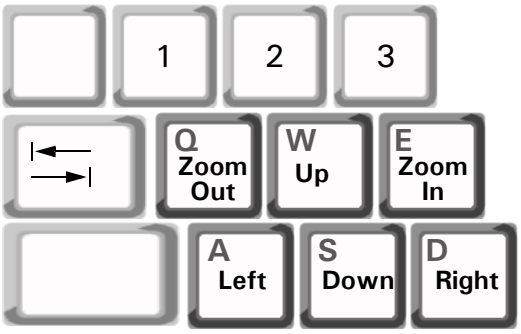


Figure 15-10. Primary movement keys, observer coordinates (2D)

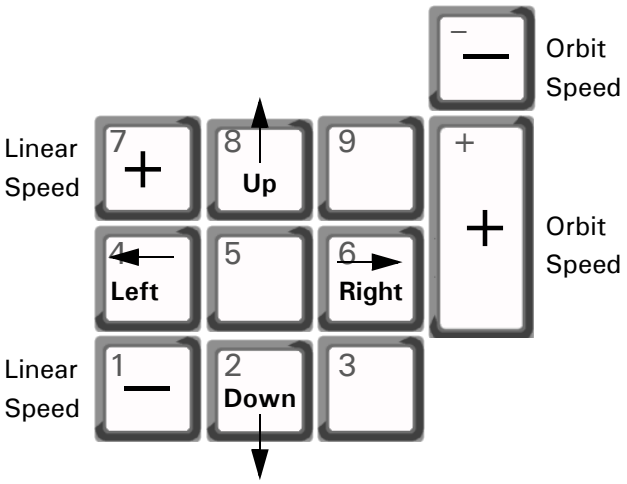


Figure 15-11. Keypad layout, heading, pitch, and roll (2D)

15.5.2. Moving the Observer Using the Mouse in the 2D Projection

You can change the observer's position and heading with the mouse. You can change the observer's position by dragging the terrain, as described in "Dragging the Terrain," on page 15-14. You can change heading by orbiting, as described in the next section. (To see the full list of mouse mappings, view the Mouse Mappings page on the Application Settings dialog box.)

Orbiting the Terrain with the Mouse

When Orient North is enabled, you cannot orbit. For information about Orient North, please see "Forcing Orientation North (Plan View Mode Only)," on page 15-24.

➤ To orbit the terrain with the mouse, press the right mouse button and drag the mouse.

15.6. Zooming the Observer

You can zoom the observer in the 3D and 2D projections. Although the apparent effect is similar for top-down views, they are conceptually and behaviorally different.

15.6.1. Zooming the Observer in the 2D Projection

In the 2D projection you can zoom the observer view closer to and away from the terrain. However, unlike in 3D, in which you cannot zoom out past the location of the observer, there is essentially no limit on zooming in and out because in 2D, the observer does not actually have altitude.

➤ To zoom in, press **e**, scroll the mouse wheel forward, or to step the zoom, press the **=** key.

➤ To zoom out, press **q**, scroll the mouse wheel backward, or to step the zoom, press the **-** (dash) key.

➤ To reset the observer to the default zoom level, press the **Space bar**.

15.6.2. Zooming the Observer in the 3D Projection

If you want to see something at a higher level of detail without moving the observer, you can zoom the observer view. Zooming the observer narrows the field of view and magnifies objects in the view. The magnification level is saved in saved views. 3D zoom has the following characteristics:

- When you zoom the observer view, its movement and mouse look speeds are decreased as the magnification increases, to allow for increased control.
- When you switch between 3D and 2D (Stealth only), VR-Vantage calculates a 2D zoom level to match the 3D view. If you then switch back to 3D, it remembers the observer location and zoom level.
- If you zoom in from a very high altitude there may be scene corruption due to insufficient precision in the z-buffer. This can be limited by moving the observer closer to the subject before zooming in. This issue is typically only encountered above 10,000 meters.
- When you zoom, VR-Vantage respects the environmental conditions at the observer's location. For example, if the observer is above the cloud layer, zooming does not let you see below the clouds. Or, if it is raining at the observer's location, the visibility will not improve as you zoom in on a distant object.
- After zooming in, when you zoom out, you can only zoom out to the 0 magnification level. That is, you can only zoom out back to the observer's location. If you want to see more of the terrain, you must move the observer to a greater distance from the terrain.
- Some terrains are not designed to be viewed at higher magnification. Performance may decrease if magnification increases the level of detail of large areas of the terrain. For example, suppose you are looking down a line of trees that is in the center of the view. At the normal magnification level there are a certain number of trees at each quality level, say 10 high, 20 medium, and 100 low (for a total of 130 trees). Suppose also that performance will be acceptable only when visualizing 15 or fewer high quality trees. If you magnify the scene 2x, you may now see 20 high, 40 medium, and 70 low quality trees. You see the same number of trees, but the performance will decrease because you are visualizing more high quality models.

➤ To zoom in, press the **=** key, press **`Ctrl`** and scroll the mouse wheel forward, or on the Observer Control Panel specify a value in the Magnification box.

➤ To zoom out, press the **–** (dash) key, press **`Ctrl`** and scroll the mouse wheel backward, or on the Observer Control Panel specify a value in the Magnification box.

---

i Each press of the **=** or **–** key, or increment of the mouse wheel, increases or decreases magnification by a factor of 1.25.

➤ To reset magnification to 1.0, press zero (**0**) or press the middle mouse button.

15.6.3. Setting the Scale Factor for Paged LODs

When you magnify the view, all distances used for determining LODs (including paged LODs) are scaled by the view magnification. In some cases, this is more aggressive than desired for the paged LODs. To remedy this problem, you can change the amount by which the paged LOD distance is scaled relative to the view magnification. For example, if view magnification is 8.0 and the Paged LOD Magnification Scale Factor is 2.0, then the distance used for paging would be scaled by 4.0 (8.0/2.0) rather than by the full factor of 8.0.

To set the scale factor for paged LODs:

1. Choose **Settings** → **Experimental Features**. The Experimental Features dialog box opens.
2. Adjust the Paged LOD Magnification Scale Factor slider.

15.6.4. Setting the Magnification LOD Angle for Paged LODs

When the observer zooms in, VR-Vantage pages in higher LOD tiles for the terrain. This increases memory usage. You can tune performance by restricting high LOD tiles to the areas you are interested in and using lower LOD tiles for the rest of the terrain. The Paged LOD Magnification LOD Angle setting lets you adjust the angle within which higher LOD tiles are used (Figure 15-12).

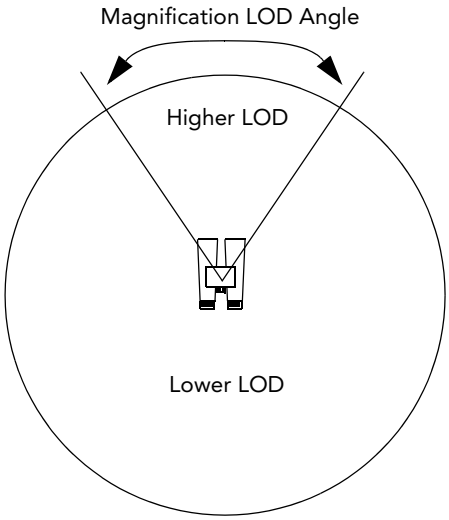


Figure 15-12. Paged LOD magnification LOD angle

If you want the observer to be able to quickly scan a wide angle of terrain, you should increase the magnification LOD angle or you may have to wait for high LOD tiles to page in as you move the observer.

On some terrains, when high LOD tiles are adjacent to low LOD tiles, there can be cracks in the terrain along the boundaries. If you experience this problem, increase the magnification LOD angle.

This setting is saved when you close VR-Vantage.

To set the magnification LOD angle for paged LODs:

1. Choose **Settings** → **Experimental Features**. The Experimental Features dialog box opens.
2. Adjust the Paged LOD Magnification LOD Angle slider.

15.7. Simulating Camera Jitter

You can simulate the jitter that cameras attached to aircraft or unsteady human hands may experience.

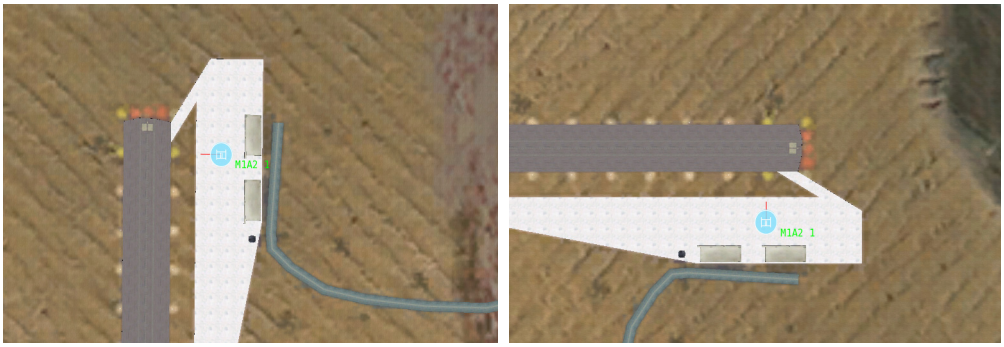
To simulate camera jitter:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 14-2).
3. In the Camera Shake section, select Allow Camera Shake.
4. Optionally, adjust the Intensity slider or change the value in the box.

15.8. Forcing Orientation North (Plan View Mode Only)

When you are in Plan View observer mode (VR-Vantage Stealth and VR-Vantage PVD only), by default the observer's heading is forced to stay North (up), regardless of the heading of the simulation object to which it is attached. You can disable this setting, in which case if you attach to a simulation object, the view of the terrain rotates to match the heading of the attached simulation object.

Figure 15-13 illustrates the view when the observer is attached to an entity with orientation forced to North or matched to that of the entity.



Orient North Orient to entity heading

Figure 15-13. Forcing orientation North

To enable or disable a northward orientation:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 21-5).
3. In the Observer Mode list, select Plan View.
4. In the list of observer settings, select or clear Orient North.

> You can also enable or disable northward orientation by choosing **Observer** → **Orient North** or clicking the Orient North button ( ) on the Observer Settings Toolbar.

15.9. Enabling and Disabling View Constraints (3D Only)

View constraints control how the observer behaves when it approaches solid objects in the view, such as the ground, buildings, and vegetation. When constraints are enabled, the observer has a 1 meter bounding sphere that prevents it from moving through terrain or props.



If view constraints are enabled, you may not be able to move the observer into buildings to view entities. Changing the near clipping plane or transparency will let you see through walls, but in constrained mode, you will still not be able to move the observer through the walls.

To enable or disable view constraints, use one of the following methods:

- Press **/** (forward slash).
- Choose **Observer** → **Constrain to Terrain**.
- On the Observer Control Panel (Figure 15-9), select or clear the Terrain Constrained check box.
- Click the Constrain to Terrain button () on the Observer Settings Toolbar.

15.10. Controlling Navigation Speed

When you navigate through the terrain, there are times you want to move quickly, covering large amounts of terrain, and times you want to move slowly, such as when you are moving down a street or within a building. VR-Vantage has the following options for controlling navigation speed:

- Set the navigation speed manually. The observer always moves at the specified speed.
- Use speed scaling to automatically move at different speeds under different circumstances. You can change the rate at which speed is scaled.

In either case, the controls for changing speed are the same. Their effect varies depending on whether or not speed scaling is enabled.

15.10.1. Enabling or Disabling Speed Scaling

When speed scaling is enabled, VR-Vantage makes the following assumptions:

- If the observer is unattached, the higher above the terrain it is, the faster you want to move.
- The closer you are to an attached object, the more slowly you want to move.

To enable or disable speed scaling, use any of the following methods:

- On the Observer Control Panel (Figure 15-9), select or clear the Scaled Speed check box.
- Choose **Observer** → **Speed Scaling**.
- Press the multiply key (*) (not the asterisk key).
- Click the Speed Scaling button () on the Observer Settings Toolbar.

Setting the Speed Scaling Mode

When speed scaling is enabled, VR-Vantage scales observer speed based on the its height above the terrain. Even if the observer is higher than all terrain points, speed can still be affected if there are changes to the terrain, such as mountains. This can be annoying if it happens unexpectedly. Therefore, you can choose to base speed scaling on height above ground level, or height above mean sea level. In that latter case, changes to the terrain would not affect the speed at which the observer moves. If you set speed scaling to use height above mean sea level, you can specify an offset.

To set the speed scaling mode, use one of the following methods:

- On the Observer Control Panel (Figure 15-9), set the Speed Scaling Mode to Above Ground Level (AGL) or Above Mean Sea Level (MSL)
- On the Display Settings dialog box, Observer Settings page, set the Speed Scaling Mode to Above Ground Level (AGL) or Above Mean Sea Level (MSL)

15.10.2. Changing the Observer's Movement Speed

You can change the observer's linear, rotational, and orbital speed. If speed scaling is enabled, these controls change the speed gain. If speed scaling is disabled, the controls change the absolute speed.

To change the observer's speed, use any of the following methods:

- Adjust one of the Observer Speed sliders on the Observer Control Panel (Figure 15-9) or on the Observer Settings page on the Display Settings dialog box. The default (center of the slider) is 1. The range is 0.1 to 10.
- Press one of the following keys to change linear speed:
  - **`Keypad 7`** (faster) or **`1`** (slower).
  - **`Ctrl`** + mouse wheel forward or back.
  - Movement key + mouse wheel forward or back.
- Press **`Keypad +`** (faster) or **`-`** (slower) to change orbital speed.



When you change the observer's speed using the keyboard or mouse, if the Observer Control Panel or the Observer Settings page is open, you can see the scale slider moving to the new setting.

15.11. Saving and Recalling Views

At any given moment, the state of the observer is called its view. You can capture views so that you can quickly return to them and you can save them for later use. VR-Vantage supports the following use cases for saving observer views:

- Associate views with a terrain. Save one or more absolute observer mode views and associate them with a terrain. The views get loaded automatically when the terrain is loaded. You can specify a default view that the observer jumps to immediately.
- Save a mix of absolute mode and attached mode views to a file. This use case is best when you want to associate a set of views with a particular simulation or simulation recording for analysis and demonstration purposes.

Nothing prevents you from including observer views that are attached to a simulation object in a terrain's saved views file, but they will not work unless the specific simulation object is available in the simulation.

If the observer is not attached to a simulation object, its view is its position and orientation in the simulated environment. If the observer is attached to a simulation object, a view saves the following information:

- Attach mode.
- Positional offset.
- Marking text.

When you save a view in the 2D view, the X and Y values are saved. The zoom level is not saved.

Saved views are managed in the Observer Views Panel (Figure 15-14).

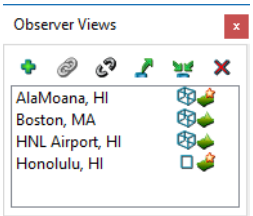


Figure 15-14. Observer Views Panel

Its buttons have the following meanings:

- Save the current view.
- Associate the views with the terrain.
- Dissociate views from the terrain.
- Save the views to a file.
- Load a saved views file and append it to the list.
- Clear the list.

The icons in the observer views list have the following meanings:

- 3D view, such as Stealth, XR, or Out-the-Window.
- 2D view, such as Plan View.
- View is associated with the terrain.
- Default view for the terrain.



Saved views are keyed to the marking text of the object to which the observer is attached.

To save the current view:

1. Choose **View** → **Observer Views Panel**. The Observer Views Panel opens (Figure 15-14).
2. Click the Add button (). The current view is added to the list.



Views are remembered only while VR-Vantage remains running. To save a view for future use, you must save it to a file, associate with a terrian, or both. Please see "Saving Views to a File," on page 15-31.

15.11.1. Recalling a View

➤ To recall a view, select it in the Observer Views Panel. The view changes immediately.

Animating Transitions Between Observer Views

Normally, when you select an observer view, the observer jumps immediately to the new view. However, VR-Vantage can smoothly transition from the current observer view to a saved view.

➤ To transition between observer views, hold down the **`Ctrl`** key and click the view in the Observer Views Panel that you want the observer to move to.

15.11.2. Deleting Views

To delete a view:

1. In the Observer Views Panel, select the view you want to delete.
2. On the right-click menu, select **Delete Observer View**.

Clearing the View List

➤ To remove all views from the list, in the Observer Views Panel, click the Clear List button (✖).

15.11.3. Loading Views Automatically When You Load a Terrain

You can associate a saved views file with a terrain so that it loads automatically when the terrain loads. You can also specify that one of the views be the default view. When you associate views with a terrain, VR-Vantage saves a file to the *./userData/terrains* directory. The file has the same name as the terrain and the extension *.osrx*.



The file used for associating views with a terrain is different than the file that is created when you manually save views to a file, as described in "Saving Views to a File," on page 15-31.

➤ To associate views with a terrain, click the Associate Views button (). An icon () is added next to each view to show that it is associated.

Specifying a Default Observer View

If you associate views with a terrain, you can specify that one of the views be the default view to use. A star is added to the association icon for default views ().

You can specify a 3D default view and a 2D default view. They work as follows:

- If there are two default views in VR-Vantage Stealth and the current observer mode is a 3D mode, it uses the 3D default view. If the current observer mode is a 2D mode, it uses the 2D default view.
- VR-Vantage PVD ignores 3D views.
- VR-Vantage IG ignores 2D views and 3D views for observer modes that it does not support, such as Stealth observer mode.

To specify a default observer view:

1. Right-click the observer view that you want to be the default.
2. On the right-click menu, choose **Set As Default View**.

Unsetting a Default View

To unset a default observer view:

1. Right-click the default observer view.
2. On the right-click menu, choose **Unset Default View**.

15.11.4. Saving Views to a File

You can save views to a file and load them at a later time. The default location for saved views is *./appData/settings/application*, where *application* is the name of the VR-Vantage application.

Saving views to a file is not the same thing as associating views with a terrain. When you associate views with a terrain, they get loaded automatically and you can specify a default view to use immediately. When you save views to a file, you need to load them manually, or from the command line. You cannot specify a default view.

Associating views with a terrain is most beneficial for saving a set of Absolute mode observer views that you always want to be available to the terrain, regardless of the simulation that you are connected to. Saving views to a file manually is most beneficial for saving views when the observer is attached to simulation objects in a simulation or for simulation-specific Absolute mode views.

To save all views to a file:

1. In the Observer Views Panel, click the Export Views button (). The Save Observer Views dialog box opens.
2. Select the file to which you want to save the views, or type the name of a new file.
3. Click OK.

Saving an Individual View to a File

You can save a single view to a file.

To save an individual view to a file:

1. Right-click the view that you want to save.
2. On the menu, choose **Save Observer View**. The Save Observer View File dialog box opens.
3. Type a name for the file.
4. Click Save.

15.11.5. Loading Views

You can load saved views and append them to the list in the Observer Views Panel.

To load views and append them to the list of views:

1. In the Observer Views Panel (Figure 15-14), click the Load Views and Append button (). The Open Observer Views dialog box opens.
2. Select the saved views file that you want to load.
3. Click OK.

15.12. Controlling the Observer from Other Applications

VR-Vantage can respond to view control messages sent by applications such as the MAK Logger and from VR-Forces plans. You can also write VR-Link-based applications to send view control messages to VR-Vantage using the classes in the VR-Vantage Control Toolkit. You can use view control messages to:

- Specify the attach mode and simulation object.
- Set the observer's position and orientation.

Acceptance of view control messages is a global setting. You can set it per connection.

The view control PDUs are experimental, and are not an official part of DIS. They are of PDU kind 202.

Some legacy applications were created with the Stealth Control Toolkit. The Stealth Control Toolkit is no longer distributed with VR-Vantage. It has been replaced with the VR-Vantage Control Toolkit. VR-Vantage responds to move, absolute, tether, mimic, track, tether track, mimic track, follow, orbit, and compass Stealth Control PDUs. It does not support the other PDUs in the Stealth Control Toolkit.

15.12.1. Enabling the Processing of View Control Messages

➤ To enable or disable the processing of view control messages, choose **Settings** → **View Controls**, or click the View Control button () on the Display Settings Toolbar.

You can specify view control settings when you connect to an exercise that override the setting in the GUI.

15.12.2. Overriding the Global View Control Settings

Processing of view control messages is a global setting. It is either on or off for all observers. However, you can override the setting on a per-connection basis (DIS and HLA only). You might want to do this if you are running VR-Vantage with several connections and want to enable or disable view controls coming from some connections, but not others.

If you override the global view control setting, you will not be able to change the view control setting in the GUI.

To override the global view control setting:

1. On the Simulation Connection page for the protocol for which you want to override the global view control setting, click Advanced.
2. Select Override Global View Control Settings.
3. Select or clear the Listen for View Control Settings check box depending on whether you want to enable view controls or disable them.
4. Set any other connection parameters desired.
5. Click Connect.



16. Attaching the Observer to Objects

This chapter describes attachment modes and how to attach the observer to simulation objects and props.

Attaching the Observer to a Simulation Object or Prop 16-2
Attaching to an Articulated Part 16-4
Attaching the Observer in Mimic Track or Tether Track Mode..... 16-4
Attaching the Observer in Mimic Location Track or Tether Location Track Mode..... 16-4
Filtering the Object Attachment List 16-5
Detaching from a Simulation Object..... 16-5
Hiding the Attached Model (3D Projection Only) 16-5
Attaching to Props 16-5
Improving Performance when Attaching to Ground Objects 16-6
Attach Modes in the 3D Projection..... 16-6
Absolute Mode..... 16-7
Follow Mode 16-7
Mimic Mode..... 16-8
Mimic Track Mode 16-9
Mimic Location Track Mode..... 16-9
Space Follow Mode..... 16-10
Tether Mode 16-11
Tether Track Mode..... 16-12
Tether Location Track Mode 16-12
Track Mode..... 16-13
Attach Modes in the 2D Projection..... 16-14
Attaching the Observer at Startup..... 16-15
Observer Attachment Tutorial..... 16-16
Binding Multiple Observers Tutorial 16-17

16.1. Attaching the Observer to a Simulation Object or Prop

You can attach the observer to a simulation object and follow it as it moves through the simulation. You can also attach the observer to props and to other observers.



This procedure applies to Follow, Tether, Track, Mimic, and Space Follow modes. To attach in Tether Track or Mimic Track mode, please see "Attaching the Observer in Mimic Track or Tether Track Mode," on page 16-4. To attach in Tether Location Track or Mimic Location Track mode, please see "Attaching the Observer in Mimic Location Track or Tether Location Track Mode," on page 16-4.

To attach the observer to a simulation object or prop, do one of the following:

- Right-click an object in the Objects List Panel (Figure 16-1) or on the terrain and select **Attach** *mode*, where *mode* is one of the supported attach modes. (For details about attach modes, please see "Attach Modes in the 3D Projection," on page 16-6 and "Attach Modes in the 2D Projection," on page 16-14.)

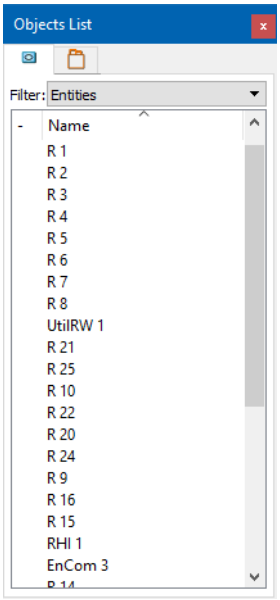


Figure 16-1. Objects List Panel

- Select a simulation object or prop. (For details about selecting simulation objects and props, please see "Selecting Simulation Objects," on page 19-2 and "Selecting Props," on page 25-25.) On the Attachments Panel (Figure 16-2), click the Attach button () to attach with the current attachment mode.

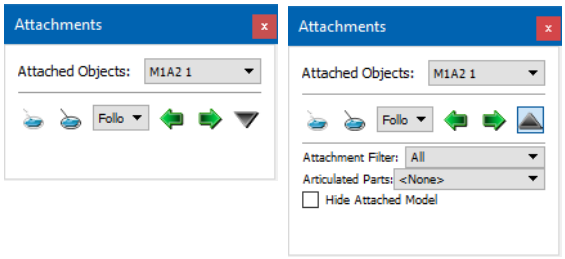


Figure 16-2. Attachments Panel

- On the Attachments Panel, click the next or previous arrows to attach to the next (or previous) object in the object list.
- **Shift**+click the object you want to attach to.
- Press greater than (**>**) to attach to the next simulation object or prop in the object list without regard to the attachment filter. Press less than (**<**) to attach to the previous object. (For details about the attachment filter, please see "Filtering the Object Attachment List," on page 16-5.)
- Press period (**.**) to attach to the next simulation object or prop allowed by the attachment filter. Press comma (**,**) to attach to the previous object.

- When you select the next or previous object, VR-Vantage chooses next and previous based on the order in which the objects were discovered. This may or may not represent the order in which they are listed in the Objects List Panel.
- When you move the attachment from one object to the next, VR-Vantage preserves whatever changes you have made to the observer's view relative to the attached object. So if, for example, you moved the observer so that it is not looking at the attached object, when you attach to the next simulation object, the observer will not be looking at this object. It will be in the same relative position to the object that you previously established. To quickly return to the default offset for the attachment mode, press **Space bar**.

16.1.1. Attaching the Observer to Another Observer

You can attach observers to other observers if you have more than one window open.

To attach an observer to another observer:

1. Filter the Objects List to show All objects or to show observers.
2. Select the window whose observer you want to attach to another observer.
3. In the Objects List, select the observer you want to attach to.
4. Right-click and select **Attach** *mode*, where *mode* is one of the supported attach modes.

16.1.2. Attaching to an Articulated Part

You can attach the observer to an articulated part on an entity, such as a turret. When you attach to an articulated part, the observer moves with the articulated part instead of the main body of the entity.

To attach to an articulated part:

1. Attach the observer to an entity, as described in "Attaching the Observer to a Simulation Object or Prop," on page 16-2.
2. On the Attachments Panel (Figure 16-2), click the down arrow. The Attachments Panel expands.
3. In the Articulated Parts list, select the part to which you want to attach the observer.

16.1.3. Attaching the Observer in Mimic Track or Tether Track Mode

To attach the observer in Tether Track and Mimic Track modes, you must specify a primary simulation object and a secondary simulation object (the tracked simulation object).

To attach the observer in Mimic Track mode or Tether Track mode:

1. Right-click the primary simulation object in the Objects List Panel or in the window. A popup menu is displayed.
2. Choose **Attach Mimic Track** or **Attach Tether Track**. The cursor changes to show an arrow and a Plus sign.
3. Click the secondary simulation object in the Objects List or in the window.

16.1.4. Attaching the Observer in Mimic Location Track or Tether Location Track Mode

To attach the observer in Tether Location Track and Mimic Location Track modes, you must specify a primary simulation object and a location.

To attach the observer in Mimic Location Track mode or Tether Location Track mode:

1. Right-click the primary simulation object in the Objects List Panel or in the window. A popup menu is displayed.
2. Choose **Attach Mimic Location Track** or **Attach Tether Location Track**. The cursor changes to show an arrow and a Plus sign.
3. Click a location in the display window.

16.1.5. Filtering the Object Attachment List

Although you can attach the observer to simulation objects and props, it is likely that at any given time you will only want to attach to one type of object (usually simulation objects). To reduce the number of objects that you have to iterate through when using Attachments Panel arrows to pick the next or previous object, you can filter the attachment list.



The filter for buoys and beacons applies to S-57 feature data, not buoys that are simulation objects, such as sonobuoys.

To filter the attachment list:

1. On the Attachments Panel (Figure 16-2), click the down arrow. The Attachments Panel expands.
2. In the Attachment Filter list, select the type of scene element that you want to attach to.

16.1.6. Detaching from a Simulation Object

➤ To detach the observer from the simulation object to which it is attached, on the Attachments Panel (Figure 16-2), click the Detach button (), or press **z**.

16.1.7. Hiding the Attached Model (3D Projection Only)

You can hide the model for the simulation object to which the observer is attached.

To hide the attached model:

1. On the Attachments Panel (Figure 16-2), click the down arrow. The Attachments Panel expands.
2. Select the Hide Attached Model check box.

16.1.8. Attaching to Props

Since props do not move, attaching the observer to a prop does not result in any observer movement. The principal benefit of attaching to a prop is to let you quickly refocus the observer on a particular location in the scene. If you attach in Track mode, using the keyboard movement keys allows you to orbit around the prop.

16.1.9. Improving Performance when Attaching to Ground Objects

When the observer is attached to air entities, Z-fighting can affect visual quality. To prevent this problem, VR-Vantage needs to calculate the entity's height above the terrain. However, the calculation can affect performance. Unfortunately, when this calculation is enabled (the default), it must be applied to all entity types. If you know that you will want to attach the observer to ground simulation objects, but not air simulation objects (or if you do not care about Z-fighting), you can disable the height above terrain calculation, which may improve performance when the observer is attached to simulation objects. To disable the height above terrain calculation, start VR-Vantage with the `-S | --suppressHatIntersectOnAttach` command line option.

16.2. Attach Modes in the 3D Projection

Attach modes are combinations of position and orientation models and constraints that produce a useful behavior for the observer. Except for Absolute mode, attach modes describe how movement of the simulation object to which the observer is attached affects the position and orientation of the observer.

VR-Vantage supports the following attach modes:

- Absolute: The observer is not attached to a simulation object; it moves in space independently of all simulation objects.
- Follow: The observer position and heading changes with the simulation object's position and heading.
- Mimic: The observer position and orientation changes with the simulation object's position and orientation.
- Mimic Track: The observer's position changes with the simulation object's position while its orientation is focused on a second entity.
- Mimic Location Track: The observer's position changes with the simulation object's position while its orientation is focused on a location.
- Space Follow: The observer is a fixed distance from the simulation object and the simulation object model is scaled to be observable from a great distance.
- Tether: The observer's position changes with the simulation object's position.
- Tether Track: The observer's position changes with the simulation object's position while its heading is focused on a second simulation object.
- Tether Location Track: The observer's position changes with the simulation object's position while its heading is focused on a location.
- Track: The observer's heading changes to track the simulation object's position and heading.

The attach modes are described in detail in the following sections. For details about how to attach the observer to a simulation object, please see "Attaching the Observer to a Simulation Object or Prop," on page 16-2.

16.2.1. Absolute Mode

In Absolute, or free-fly mode, the observer is not attached to a simulation object. Movement of simulation objects does not affect the observer in any way.

16.2.2. Follow Mode

In Follow mode, the observer maintains its relationship to the simulation object as follows:

- When the simulation object's position changes, the observer's position changes to maintain its offset.
- When the simulation object's heading changes, the observer's heading and position change to maintain the offset.
- When the simulation object's pitch or roll change, the observer pitch and roll do not change.

Figure 16-3 illustrates the characteristics of Follow mode.

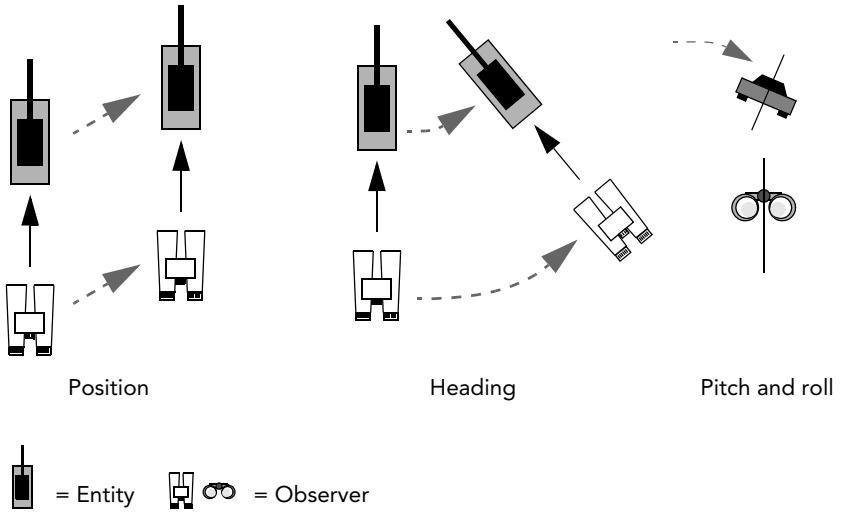


Figure 16-3. Follow mode

16.2.3. Mimic Mode

In Mimic mode, the observer behaves as though it is rigidly connected to the entity. Use Mimic mode to create the effect of riding in a vehicle cockpit, or of following a vehicle from an external location while connected to it by a solid, invisible rod. The observer maintains its relationship to the simulation object is as follows:

- When the simulation object's position changes, the observer's position changes to maintain its offset.
- When the simulation object's heading changes, the observer's heading and position change to maintain the offset.
- When the simulation object's pitch or roll change, the observer's pitch and roll change and the observer's position changes to maintain its offset.



When cockpit displays are enabled, you cannot change the position or orientation of the observer relative to the entity.

Figure 16-4 illustrates the characteristics of Mimic mode.

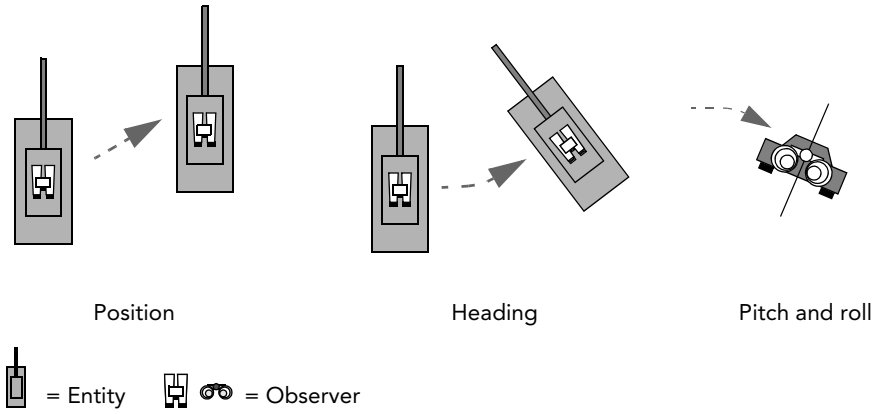


Figure 16-4. Mimic mode

16.2.4. Mimic Track Mode

Mimic Track mode combines Mimic mode and Track mode. The observer's position changes with that of the entity to which it is attached, as in Mimic mode, while its orientation changes to remain focused on a secondary, tracked entity.

Orientation controls are disabled.

Figure 16-5 illustrates the characteristics of Mimic Track mode.

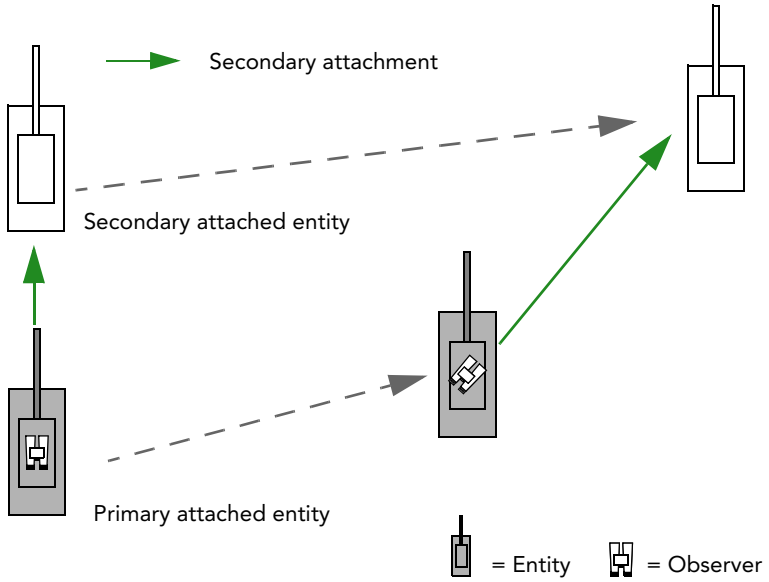


Figure 16-5. Mimic Track mode

16.2.5. Mimic Location Track Mode

Mimic Location Track mode works like Mimic Track mode except it tracks a location on the ground rather than an entity.

16.2.6. Space Follow Mode

Space Follow mode attaches the observer above an entity, looking down at it. Although it can be used with any entity, its primary purpose is for attaching to space objects such as satellites. You can also usefully use it with fixed-wing and rotary-wing entities.

The observer is attached at a fixed distance, determined by the near clipping plane for the object. For a satellite, this distance is approximately 500,000 meters. Since it is not possible to see regular models at that distance, the attached object is scaled to be visible. (You cannot change the scaling of the attached object.) Other objects are not scaled (except to the extent that model scaling might be enabled in VR-Vantage Stealth).

The observer maintains its relationship to the simulation object as follows:

- When the simulation object's position changes, the observer's position changes to maintain its offset.
- When the simulation object's heading changes, the observer's heading changes.
- When the simulation object's pitch or roll changes, the observer's pitch or roll does not change.

You cannot move the observer towards or away from the simulation object. You can rotate the observer around the entity, but you cannot go below the horizontal plane of the entity.

Figure 16-6 illustrates the characteristics of Space Follow mode.

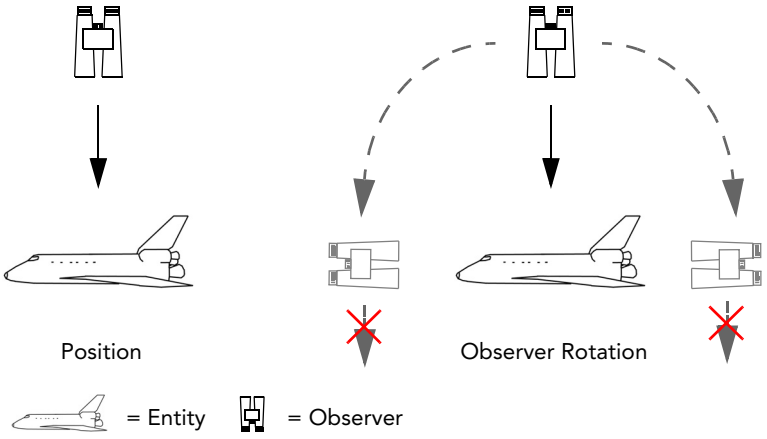


Figure 16-6. Space Follow mode

16.2.7. Tether Mode

In Tether mode, the observer's maintains a constant offset from the simulation object, but it is not affected by the simulation object's heading, pitch, or roll, as follows:

- When the simulation object's position changes, the observer's position changes to maintain its offset.
- When the simulation object's heading changes, the observer's heading does not change.
- When the simulation object's pitch or roll change, the observer's pitch and roll do not change.

Figure 16-7 illustrates the characteristics of Tether mode.

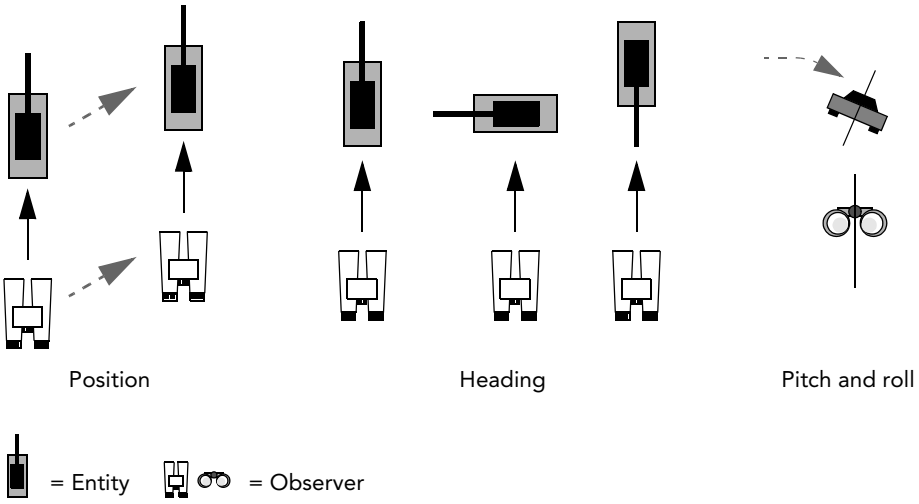


Figure 16-7. Tether mode

16.2.8. Tether Track Mode

Tether Track mode combines Tether mode and Track mode. The observer's position changes with that of the entity to which it is attached, as in Tether mode, while its orientation changes to remain focused on a secondary, tracked entity.

Orientation controls are disabled.

Figure 16-8 illustrates the characteristics of Tether Track mode.

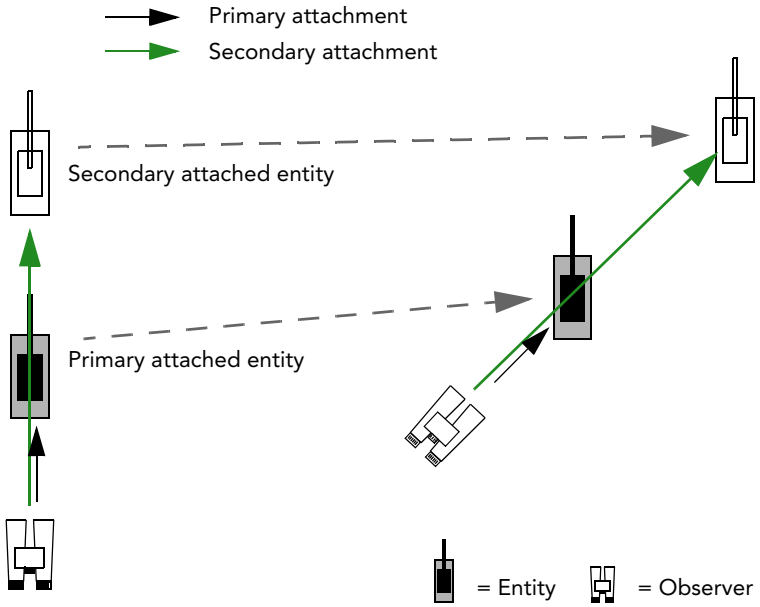


Figure 16-8. Tether Track mode

16.2.9. Tether Location Track Mode

Tether Location Track mode works like Tether Track mode, except that it tracks a location on the ground instead of an entity.

16.2.10. Track Mode

In Track mode, the observer's heading changes to track the simulation object's movement, keeping it centered in the field of view. The observer's position does not change. The observer maintains its relationship to the simulation object as follows:

- When the simulation object's position changes, the observer's heading changes to keep the entity centered in the observer's view; the observer's position does not change.
- When the simulation object's heading changes, the observer's heading does not change.
- When the simulation object's pitch or roll change, the observer's pitch and roll do not change.

Figure 16-9 illustrates the characteristics of Track mode.

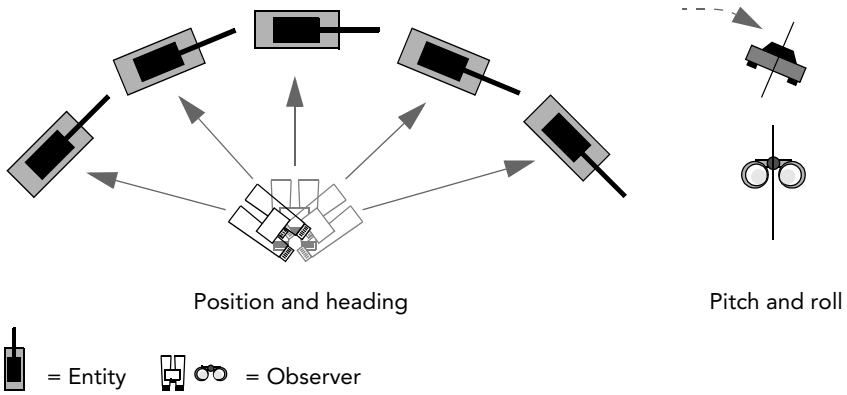


Figure 16-9. Track mode

16.3. Attach Modes in the 2D Projection

The 2D projection supports Follow mode and Tether mode. In Follow mode, the heading of the observer changes with the entity (if Orient North is disabled); in Tether mode, it does not. You can use the standard movement keys (**w**, **a**, **s**, **d**) to offset the observer with respect to the simulation object's position. You can use the orbit keys (left arrow, right arrow) to offset the observer's orientation with respect to the entity's orientation (in Follow mode) or the terrain.

If the observer is attached to an entity in the 3D projection and you switch to the 2D projection, the attach mode is mapped to either Follow or Tether modes, as shown in Table 16-1. If you then switch back to the 3D projection the original attach mode is restored.

Table 16-1: 3D to 2D attachment mode mapping

3D Attach Mode	2D Attach Mode
Follow	Follow
Mimic	Follow
Mimic Track	Follow
Mimic Location Track	Follow
Space Follow	Follow
Tether	Tether
Tether Track	Tether
Tether Location Track	Tether
Track	Tether

16.4. Attaching the Observer at Startup

There may be times when you want to attach the observer to an entity as soon as VR-Vantage starts up. For example, suppose you are using VR-Vantage IG as a flight simulator. You might want it to start up and immediately display the cockpit view for the aircraft. VR-Vantage has a command-line argument, `--bindObservers`, that lets you bind the observer to an entity using saved views.

To start VR-Vantage with the observer bound to an entity:

1. Run a live or recorded simulation.
2. Attach the observer to the entity that you want it to be attached to when you start VR-Vantage. If you have multiple observers, create a saved view for each observer that you want to attach automatically.
3. Save the views.
4. If you have multiple observers, save the display engine configuration.
5. Start VR-Vantage with the `--bindObservers` command-line option and use your saved views file as the argument to `--bindObservers`. If you need to specify more than one observer, use `--dispSetting` to specify the display engine configuration file that defines the observers. You must also specify the terrain file to load.:

   ```
   vrvIG --bindObservers "observer1,...,
     observern,../appData/settings/vantageIG/
     saved_views.osrx" --dispSetting displayconfig.dcx
     terrain_file.mtf
   ```

   If the simulation is running, VR-Vantage attaches the observer to the entity in the saved view. If the simulation is not running, VR-Vantage attaches the observer when the entity is next discovered.



If your display engine configuration has more than one observer, you can save multiple views and automatically bind the observers to the views. The observers are bound to views on a one-to-one basis in the order that the observers are listed in the command and the saved views are listed in the file. If you only have one observer, it is automatically bound to the first view in the file.

16.4.1. Observer Attachment Tutorial

This tutorial implements the procedure described in the previous section using files provided with VR-Vantage and MAK Data Logger. It attaches the observer to one of the aircraft in the *Raid2013-viewControl-DIS.lgr* recording.

To attach the observer to an entity at startup:

1. Start VR-Vantage IG.
2. Load the *Ala Moana.mtf* terrain.
3. Start MAK Data Logger (DIS).
4. Load *Raid2013-viewControl-DIS.lgr*.
5. Connect VR-Vantage IG to the simulation network.
6. Choose **Settings** → **View Controls** to disable view controls.
7. Run the *Raid2013-viewControl-DIS.lgr* recording.
8. Attach the observer in Mimic mode to UAV-Predat1.
9. If the Observer Views Panel is not open, choose **View** → **Observer Views Panel**.
10. On the Observer Views Panel, click the Add button (✚). The view is listed in the panel window.
11. Save the view to a file, for example *UAV-Predat1.osrx*.
12. Shut down VR-Vantage IG.
13. Start VR-Vantage IG with the following command line:

```
vrvIG --bindObservers "Observer 1,../appData/
  settings/vantageIG/UAV-Predat1.osrx"
  "../userData/terrains/Ala Moana.mtf"
```

In this example, it is not strictly necessary to specify the observer because there is only one observer. The observer name is included to illustrate the complete syntax of the command. If you had more than one observer in your display engine configuration and multiple views to attach them to, you could use a command like the following:

```
vrvIG --bindObservers "Observer 1,Observer A,../appData/settings/
  vantageIG/UAV-Predat1.osrx" --dispSetting displayConfig.dcx
  "../userData/terrains/Ala Moana.mtf"
```

You could also do this:

```
vrvIG --bindObservers "Observer A,Observer 1,../appData/settings/
  vantageIG/UAV-Predat1.osrx" --dispSetting displayConfig.dcx
  "../userData/terrains/Ala Moana.mtf"
```

The observers get bound to views in the order in which they are listed.

VR-Vantage includes an example saved view, *UAV-Predat1.osrx*, that you can use to try out this example.

16.4.2. Binding Multiple Observers Tutorial

This tutorial shows how to bind multiple observer views to multiple windows. The tutorial does the following:

- Creates three windows.
- Saves the display engine configuration.
- Saves different views for each window.
- Loads the terrain, saved views, and display engine configuration from the command line.

VR-Vantage loads the terrain, opens the three windows in the display engine configuration, and assigns a saved view to each window.

The tutorial uses VR-Vantage Stealth and both 2D and 3D windows. However, the same procedure would apply to applications that support just 2D or just 3D windows.

Create the Windows

To bind multiple observer views to multiple windows:

1. Start VR-Vantage Stealth.
2. Load *Ala Moana.mtf*.
3. Choose **View** → **Display Engine Configuration Editor Panel**. The panel shows one window with one channel. The observer assigned to the channel is named Observer 1 (Figure 16-10).

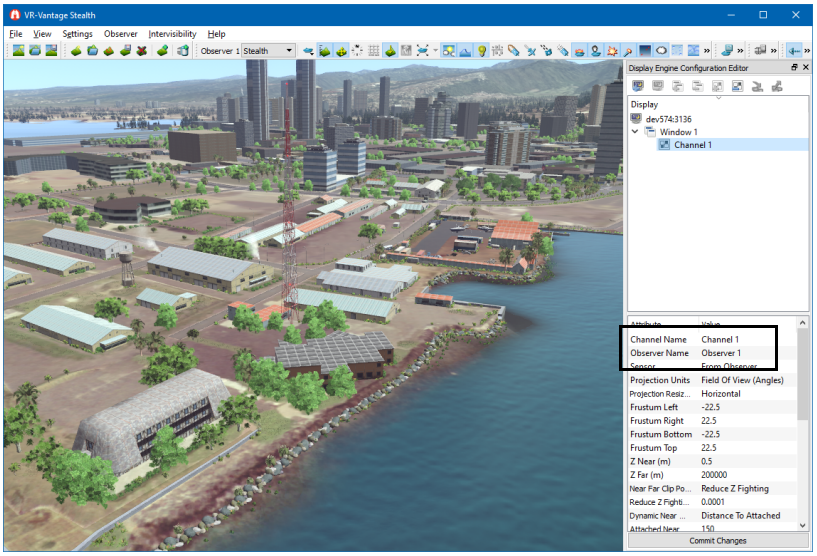


Figure 16-10. Observer 1

4. Choose **File** → **New Stealth Window**. A window titled Stealth Inset 1 opens. It shows the same view as the main window. A window called Stealth Inset 1 is added to the display engine configuration. The channel properties show that the window uses Observer 2 (Figure 16-11).

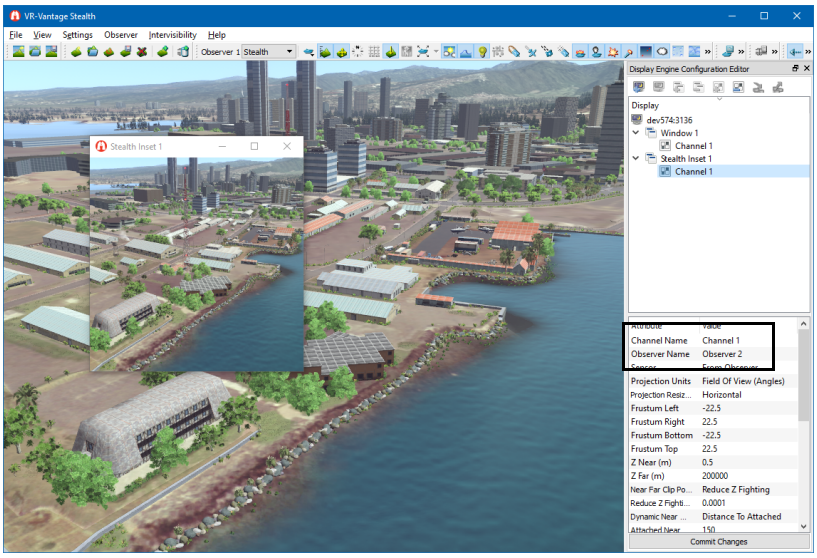


Figure 16-11. Observer 2

5. Choose **File** → **New Plan View Window**. A window titled Plan View Inset 1 opens. It shows a portion of the view in the main window. A window called Plan View Inset 1 is added to the display engine configuration. The channel properties show that the window uses Observer 3 (Figure 16-12).

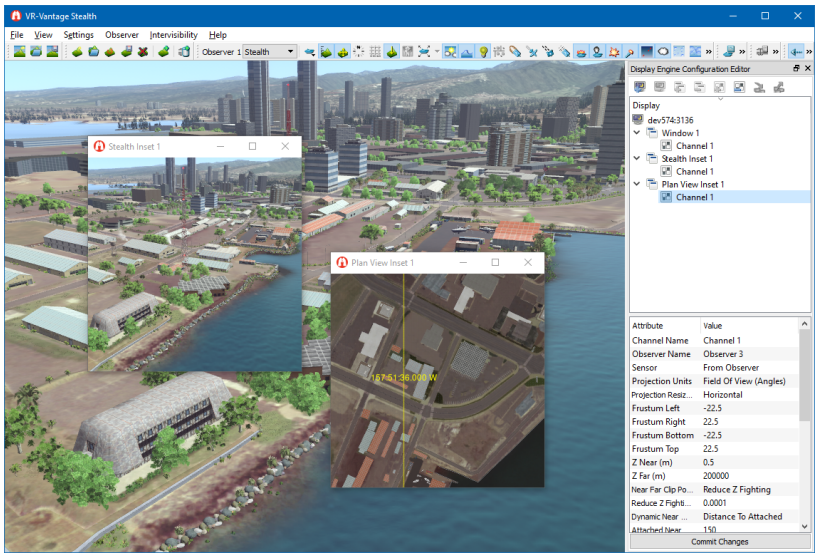


Figure 16-12. Observer 3

Save the Display Engine Configuration

To save the display engine configuration:

1. In the Display Configuration Editor Panel, select the display engine (the top line).
2. Click the Save Display Engine Configuration button ( ). The Save Display Engine Configuration dialog box opens.
3. Type a name for the configuration, for example, *BindObserversTutorial.dcx*.
4. Click Save.

Create and Save the Views

To create observer views for each observer and save them:

1. If the Observer Views Panel is not open, choose **View** → **Observer Views Panel**.
2. Click in the main window to give Observer 1 focus. Confirm this by looking at the observer name displayed on the Observer Settings Toolbar.
3. On the Observer Views Panel, click the Add button ( ). A view called 1 - Absolute is added.
4. Double-click the name of the view to make it editable. Change the name to Default View. (This view will actually be identical to the AlaMoana Port view that is associated with the terrain.)
5. Click in Stealth Inset 1. Make sure that the Observer Settings Toolbar shows that the active observer is Observer 2.
6. Drag the view of the terrain to show a different location, such as the Diamondhead crater.
7. Add a new view to the Observer Views Panel.
8. Change the name of the view to Location Two.
9. Click in Plan View Inset 1. Make sure that the Observer Settings Toolbar shows that the active observer is Observer 3.
10. Zoom out to see all of the island of Oahu. The various windows should now look something like (Figure 16-13).

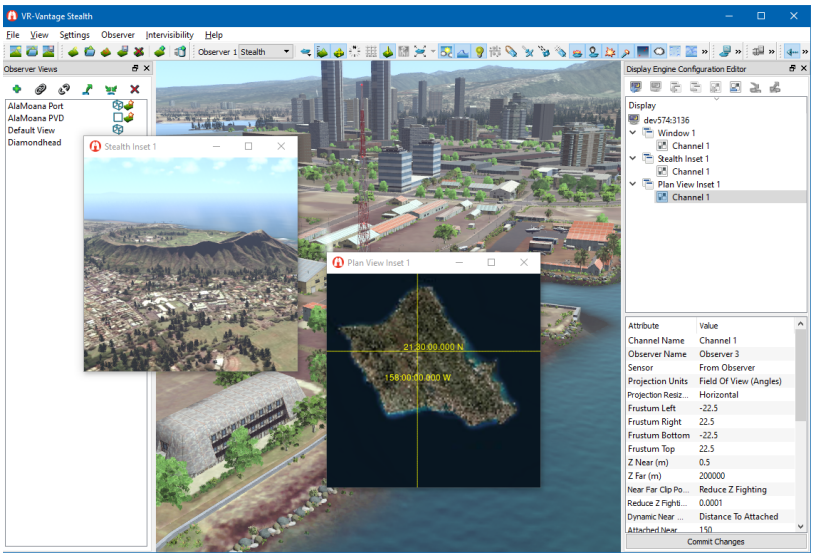


Figure 16-13. VR-Vantage Stealth with saved views assigned

11. Add a new view to the Observer Views Panel.
12. Change the name of the view to Oahu.
13. Click the Save Observer Views button (). The Save Observer Views List prompt opens.
14. Click Save As. The Save Observer Views File dialog box opens.
15. Type a name for the saved views, for example, *BindObserversTutorialViews.osrx*.
16. Click Save.
17. Shut down VR-Vantage.

Start VR-Vantage with the Views and Display Engine Configuration

➤ To load the display engine configuration and bind observers to the saved views, start VR-Vantage Stealth with the following command:

```
C:\MAK\vrvantage2.3\bin64\vrvStealth.exe
  --dispSetting ..\appData\settings\stealth\BindObserversTutorial.dcx
  --bindObservers "Observer 1,Observer 2,Observer 3,
  ..\appData\settings\stealth\BindObserversTutorialViews.osrx"
  ..\userData\terrains\VR-TheWorldOnline.mtf
```

VR-Vantage Stealth starts up, loads the display engine configuration, opens the windows, and assigns the saved views to the windows (Figure 16-13).



17. Inset Views, Sensor FOV, and HUDs

This chapter describes special types of inset windows – inset views, instrument panels, and sensor fields of view, and head up displays.

Inset Views 17-2
Displaying Cockpit Displays 17-3
Displaying Instrument Panels 17-6
Configuring Entities to Use Instrument Panels 17-7
Displaying Sensor Views 17-8
Displaying a Sensor Field of View Cone 17-9

17.1. Inset Views

You can display inset views of individual simulation objects. Figure 17-1 shows a helicopter hovering over the Ala Moana area in the *MAK Earth (online).mtf* terrain with an inset view of a car driving through the town.



Figure 17-1. Inset view

! By default, when you create an inset view, the observer in the inset view is attached to the simulation object in Mimic mode. This has the side effect of causing the 3D model in the main window to disappear. (The reverse occurs if you attach the observer in the main window to the simulation object in Mimic mode.) To make the model reappear, select the inset view and change the attach mode or detach the observer.

You can change the observer position and orientation in an inset view using keyboard controls. You can also edit window and channel properties for the window in the Display Engine Configuration Editor. (For information about the Display Engine Configuration Editor, please see Chapter 6, *Windows, Channels, and Display Engines*.)

When you create an inset view, the title bar contains the simulation object's name. However, the window is not actually tied to the simulation object and you can change the view in the inset view and view other simulation objects. The title bar will continue to display the name of the simulation object for which the window was created.

To open an inset view:

1. Right-click a simulation object (or its entry in the Objects List Panel). A menu opens.
2. On the menu, choose **Create Inset View**.

17.2. Displaying Cockpit Displays

VR-Vantage can display cockpit displays created with GL Studio. It includes cockpits for fixed-wing, rotary-wing, and ground vehicle entities. As with other entity effects, cockpits have schemas and model definitions that you map to entity types. For details about cockpit display models, please see "Adding New Cockpit Display Models," on page 35-10.

VR-Vantage supports 2D cockpits (Figure 17-2) and 3D cockpits (Figure 17-3). A 2D cockpit is fixed to the center of the screen and you cannot zoom in or out on the instrument panel. A 3D cockpit is aligned with the front of the vehicle. If you move the eyepoint, the instrument panel moves in or out of view as you would expect. You can zoom in and out on the instruments.



Figure 17-2. 2D cockpit



Figure 17-3. 3D cockpit

- Cockpits are displayed only when the observer is attached to an entity in Mimic mode.
- VR-Vantage includes one 3D cockpit, for the F-16 fighter jet.
- Cockpits are not supported on Linux.

To display cockpits:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Cockpit Display Settings page (Figure 17-4).

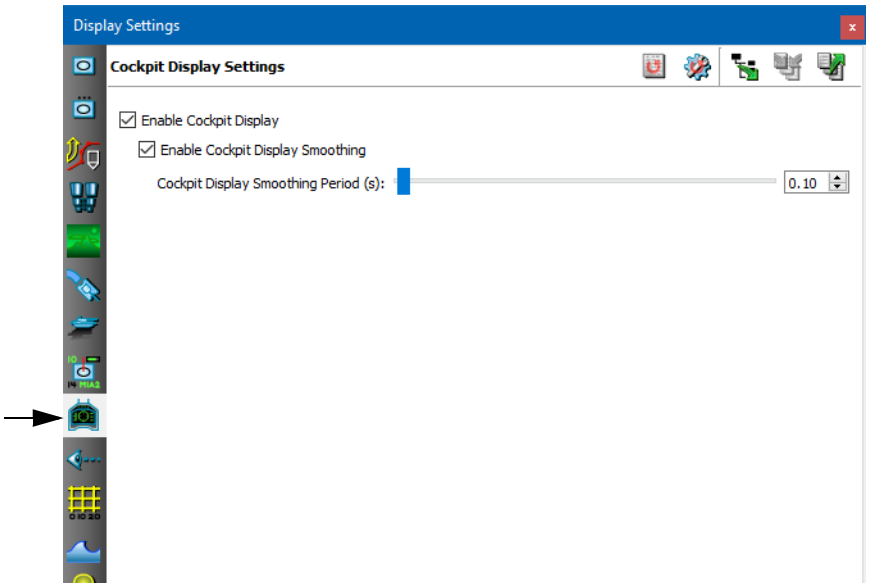


Figure 17-4. Cockpit display settings

3. To enable or disable cockpit displays, select or clear the Enable Cockpit Display check box.
4. Attach the observer to an entity in Mimic mode.



You can also enable and disable cockpit displays by choosing **Settings** → **Cockpit Displays**.

17.3. Displaying Instrument Panels

Instrument panels are inset windows that display a vehicle instrument such as an altimeter or artificial horizon. Unlike a cockpit display, which is a fixed display that operates in Mimic mode, an instrument panel is an independent window (like an inset view) that you can place anywhere on the screen (Figure 17-5). It does not affect the observer mode. You can organize multiple instrument panels to create your own virtual cockpit.



Figure 17-5. Instrument panels

When you create an instrument panel, the title bar of the window lists the entity for which you created it.

- Instrument panels are GL Studio controls.
- Instrument panels are not supported on Linux.
- If the source of the instrument panel is removed from the exercise, for example when a Logger recording rewinds, the instrument panels are removed from the inset windows. To see them again you must open new instrument panels. Because the relationship between an instrument panel window and an entity is not persistent across multiple runs of an exercise, you cannot save an instrument panel layout for an entity. (You could save the layout of the windows to a display configuration, but they would not display instrument panels when you loaded the configuration.)
- If you want to put multiple instrument panels in one window, like the six-pack panel provided with VR-Vantage, you must use GL-Studio.

To open an instrument panel:

1. Select the entity for which you want to display an instrument panel.
2. On its right-click menu, choose **Create Instrument Panel**. A submenu is displayed.
3. Choose the instrument panel you want to open.

17.3.1. Configuring Entities to Use Instrument Panels

VR-Vantage includes a set of instrument panels for fixed-wing aircraft, although many of them may be appropriate for rotary-wing aircraft, and even for ground vehicles. (For example, you might want to show a compass for a ground vehicle to get its heading or an altimeter to show its height above sea level.) The availability of instrument panels to an entity is determined by the instrumentPanelCategory parameter in its model definition.

The instrument panels provided with VR-Vantage have the instrumentPanelCategory set to FixedWing. All fixed-wing entities have this parameter set in their model definition. None of the other entities do. If you want other entities to display instrument panels provided with VR-Vantage, you can add this parameter to their model definitions. (You could also copy an instrument panel model definition and change the category to some other value if you want to. Regardless, to use an instrument panel, an entity's model definition must include the instrumentPanelCategory parameter and set it to a valid value.)

If you create new instrument panels, you must add instrument panel model definitions for them and specify a value for the instrumentPanelCategory parameter. Then you can add the parameter to the appropriate entity model definitions.

17.4. Displaying Sensor Views

VR-Vantage can display the view from gimbaled visual sensors simulated by VR-Forces, such as a camera on a UAV (3D only). The view is displayed in an inset window that has information about the observer mode and area being viewed (Figure 17-6). The window has its own observer and you can change the observer mode in the view.

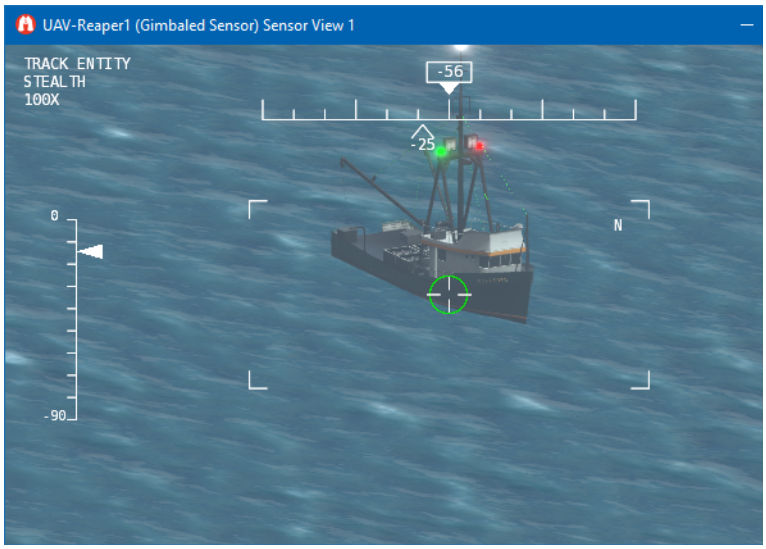


Figure 17-6. Sensor View

To display a sensor view:

1. Select an entity that has a gimbaled sensor.
2. Right-click the entity and on the right-click menu choose **Create Sensor View** → *sensor*, where *sensor* is one of the sensors on the entity.

17.4.1. Displaying a Sensor Field of View Cone

You can display translucent cones that show the area on the terrain that a gimbaled sensor is observing. Each cone is a different color. Figure 17-7 shows two sensor field of view cones. One is focused on an entity and is zoomed in for a tight focus. The other is focused on a point on the terrain.



Figure 17-7. Sensor field of view cones

To display sensor field of view cones:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 14-2).
3. In the list of settings, select the Sensor Fields of View check box.



You can also enable and disable sensor field of view cones by choosing **Observer** → **Sensor Fields of View** or by clicking the Sensor Fields of View button () on the Observer Settings Toolbar.



18. Exaggerating Models and Terrain

This chapter describes terrain scaling and model scaling – features primarily used in XR observer mode.

Exaggerating the Scale of 3D Entity Models 18-2
Exaggerating the Terrain Scale 18-5
Enabling Terrain Scaling 18-6
Scaling the Terrain 18-7
Track Histories and Terrain Scaling 18-8

18.1. Exaggerating the Scale of 3D Entity Models



Model scaling and XR mode is only supported by VR-Vantage Stealth.

VR-Vantage supports the following types of model scaling:

- XR scaling. XR scaling tries to give all entities approximately equal visibility. It does not preserve size relationships between entities. The visual effect of model scaling depends on an entity's size and the observer's distance from the entity. When the observer is close to an entity, scaling models may not make any noticeable difference in its size.
- Linear scaling. Linear scaling scales all entities by the current model scale factor. The distance from the observer is not taken into account. A model scale factor of 10 scales all entities to be 10 times normal size.

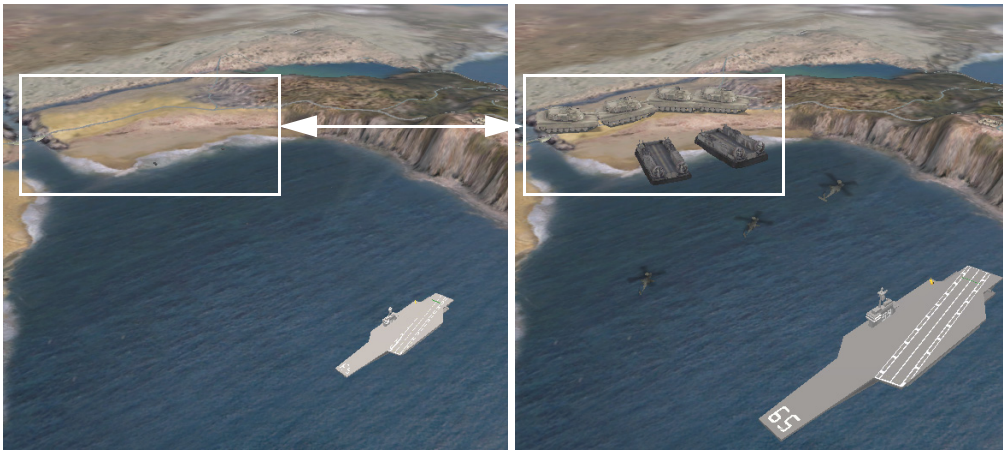


Figure 18-1. Model scaling off and on (Stealth observer mode)

Another way to quickly change the visibility of entities is to switch to XR observer mode. In XR mode, entities are immediately scaled and are represented by colorized 3D models to enhance their visibility.



Figure 18-2. XR mode

To scale entity models:

1. Choose **Observer** → **Advanced Settings**. The Display Settings dialog box opens to the Observer Settings page (Figure 18-3).

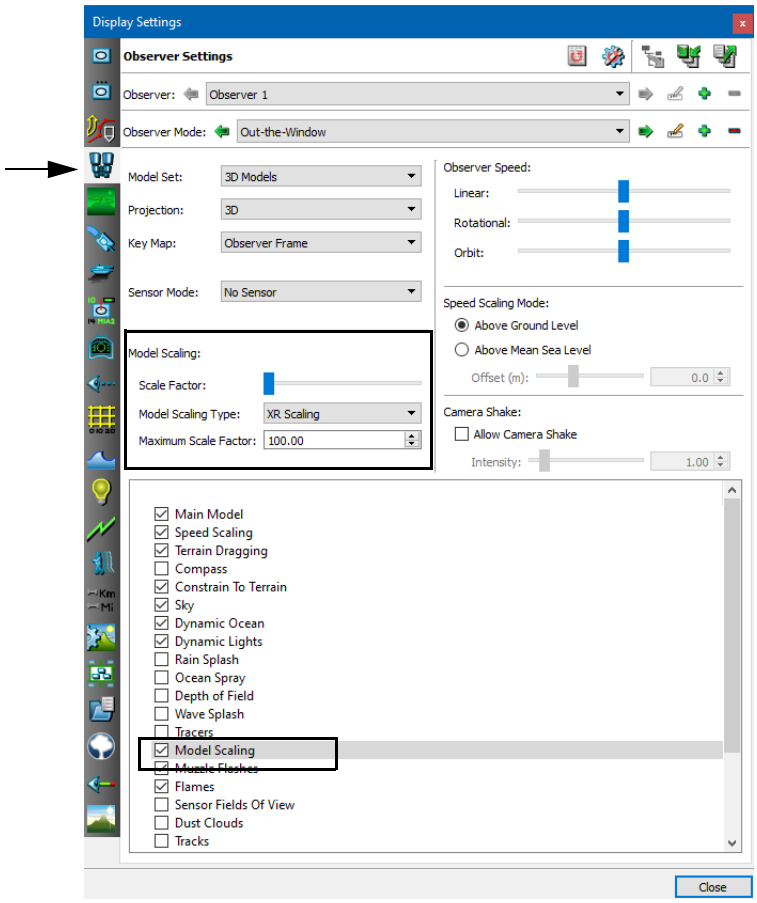


Figure 18-3. Observer Settings page

2. In the list of observer attributes, select Model Scaling.
3. In the Model Scaling Type list, select the type of scaling you want to use – XR or linear.
4. Optionally, change the Maximum Scale Factor.
5. Adjust the Scale Factor slider to the amount of scaling that you want.

You can also scale models from the Observer Settings Toolbar.

To scale models from the Observer Settings Toolbar:

1. If model scaling is not enabled, click the Model Scaling button ( ) on the Observer Settings Toolbar.
2. Click the down arrow on the button. The Scale Factor slider is displayed (Figure 18-4).



Figure 18-4. Scale Factor slider

3. Adjust the slider to scale the models.

The Scale Factor slider is a dockable widget that you can undock and leave visible as long as model scaling is enabled.

18.2. Exaggerating the Terrain Scale

In XR mode, you can exaggerate the terrain scale to improve your sense of line-of-sight issues and other terrain factors that can affect an exercise. Figure 18-5 and Figure 18-6 illustrate terrain from the same observer location with and without terrain scaling. Notice how much easier it is to understand the relationship of the aircraft to the ground in the scaled view.



Figure 18-5. No terrain scaling

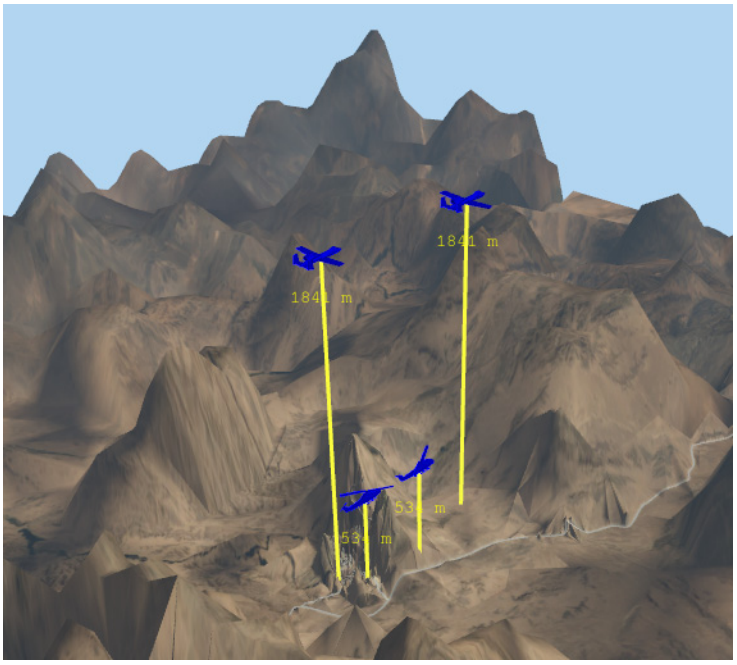


Figure 18-6. Terrain scaling

When you scale the terrain, terrain, props, and SpeedTrees all get scaled. Simulation objects and entity labels get positioned so that they are in the correct position relative to the scaled terrain. However, the text on entity labels and other information panels displays their correct, unscaled location.

When you enable or disable terrain scaling or change the scale factor, track histories get cleared. You can specify that track histories not be cleared when you scale the terrain.

! Terrain scaling is not supported on geocentric terrains.

18.2.1. Enabling Terrain Scaling

➤ To enable or disable terrain scaling, choose **Settings** → **Terrain Scaling**, or click the Terrain Scaling button (▾) on the Display Settings Toolbar.

---

i You can also enable and disable terrain scaling on the Display Settings dialog box, Render Settings page.

18.2.2. Scaling the Terrain

To scale the terrain

1. Enable terrain scaling. (Choose **Settings** → **Terrain Scaling**, or click the Terrain Scaling button ( ) on the Display Settings Toolbar.)
2. Click the arrow next to the Terrain Scaling button. The Terrain Scale Factor slider is displayed (Figure 18-7).



Figure 18-7. Terrain Scale Factor slider

3. Adjust the slider.

You can also scale the terrain on the Display Settings dialog box, Render Settings page.

18.2.3. Track Histories and Terrain Scaling

By default, when you enable or disable terrain scaling or change the scale factor, track histories get cleared and restarted. This is because track histories are not scaled and they would probably misrepresent the simulation object's track after the terrain was scaled. However, you can specify that track histories not be cleared when you scale the terrain.

To specify whether or not track histories are cleared when the terrain is scaled:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 18-8).

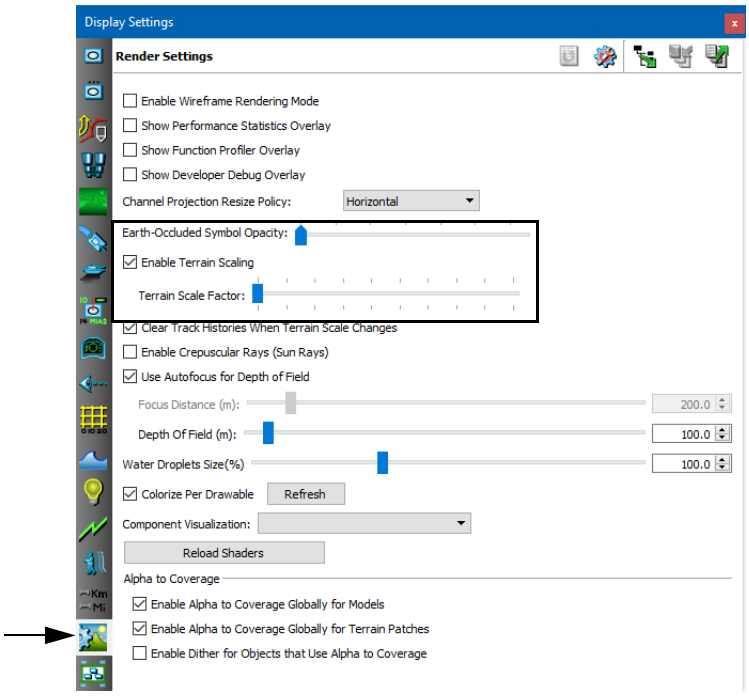


Figure 18-8. Render Settings page

3. Select or clear the Clear Track Histories when Terrain Scale Changes check box.



V. Viewing Simulation Objects

The chapters in this section explain how to manage the display of simulation objects (entities and units) and the information about them that is available.



19. Viewing Simulation Objects

This chapter describes how to select entities and units, and manage how they are viewed.

Selecting Simulation Objects 19-2
Displaying Entity Labels 19-3
Displaying Text Labels for 3D Models 19-5
Turning Entity Labels On and Off 19-6
Pinning 3D Text Labels to the Window 19-6
Displaying Labels for 2D Icons 19-7
Configuring Shadowing and Background Color for 2D Labels 19-9
Inset Views 17-2
Changing Force Colors 19-9
Displaying HAT Lines for Entities (3D Only) 19-10
Changing the Model Set 19-11
Simulation Object Icons 19-11
Changing the Size of 2D Icons 19-12
Rotating 2D Icons to a Simulation Object's Heading 19-13
Viewing Occluded 2D Icons 19-14
Hiding 3D Models 19-15
Display of Embarked Entities 19-15
Viewing Object Counts 19-15
Displaying Models for Remote Observers 19-16
Filtering Simulation Objects Using Interest Management 19-16
Enabling Interest Management 19-17
Configuring Interest Management 19-17

19.1. Selecting Simulation Objects

Many procedures require you to select one or more simulation objects. When a 3D model is selected, a box is drawn around it (Figure 19-1). When a 2D icon is selected, it is highlighted. In either case, the simulation object is highlighted in the Objects List Panel.

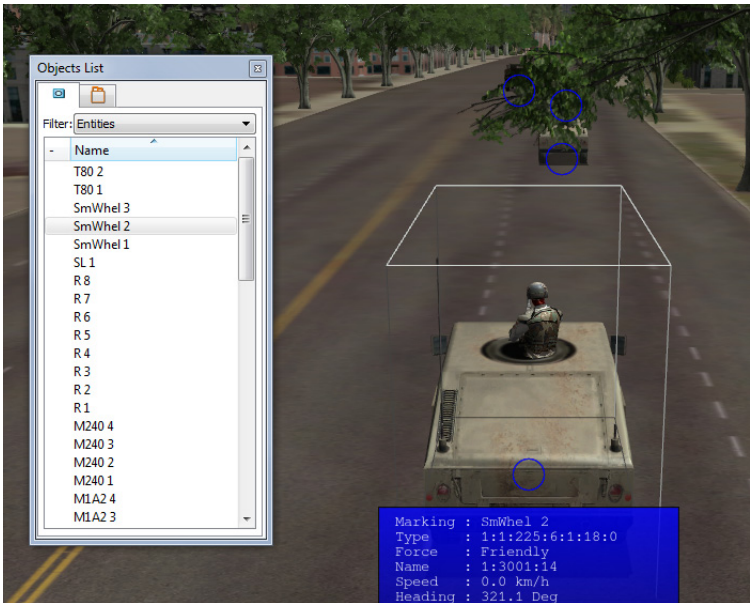


Figure 19-1. Selected 3D model

➤ To select a simulation object, click it in the window, or select it in the Objects List Panel.

For multiple selection, standard conventions for **`Shift`**+click and **`Ctrl`**+click apply.

19.2. Displaying Entity Labels

For 3D models, VR-Vantage displays labels that help you locate and identify entities (Figure 19-2). The label consists of a locator circle, which is always visible when labels are enabled, and a text label that is displayed when you mouse over an entity. All entities within the view have locator circles, whether the model is visible or not. The color of an entity's label represents its force – blue for friendly, red for opposing, and green for neutral.

A 3D text label displays the entity's:

- Marking text
- Entity type enumeration
- Force
- Name
- Speed
- Heading
- Latitude and longitude
- Altitude.



You can configure the attributes of the entity label, such as font size, and the height and width of the background by editing the entityInfo model definition in the Visual Models Editor. For details about editing model definitions, please see "Creating and Editing Model Definitions," on page 32-2.

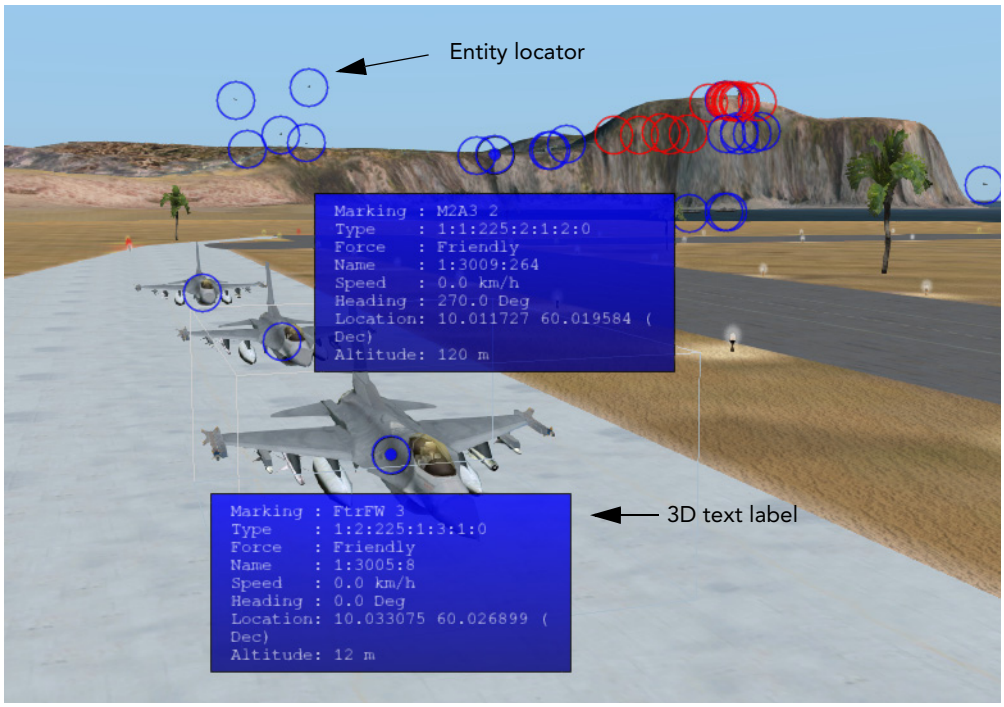


Figure 19-2. Entity labels for 3D models

For 2D icons (VR-Vantage PVD and VR-Vantage Stealth), labels are a set of individually enabled text labels, called symbol decorations (Figure 19-3). In VR-Vantage Stealth, entity labels for 2D icons are supported both in the 2D view and the 3D view. If you resize 2D icons, the labels also get resized. 2D icons can display the simulation object's:

- Acceleration.
- Altitude above mean sea level.
- Altitude above ground level.
- Heading Indicator.
- Label.
- Location.
- Name.
- Orientation.
- Velocity.

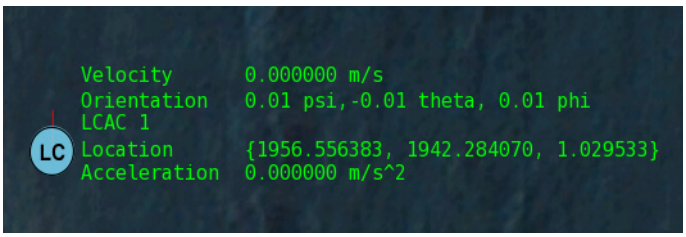


Figure 19-3. Label for a 2D icon

19.2.1. Displaying Text Labels for 3D Models

To view a 3D text label, use one of the following methods:

- Move the cursor over the circle that represents the entity.
- On the Display Settings dialog box, Entity Display Settings page, select Show Information Only For Selected Entities (Figure 19-4). Then select the entity whose label you want to see.

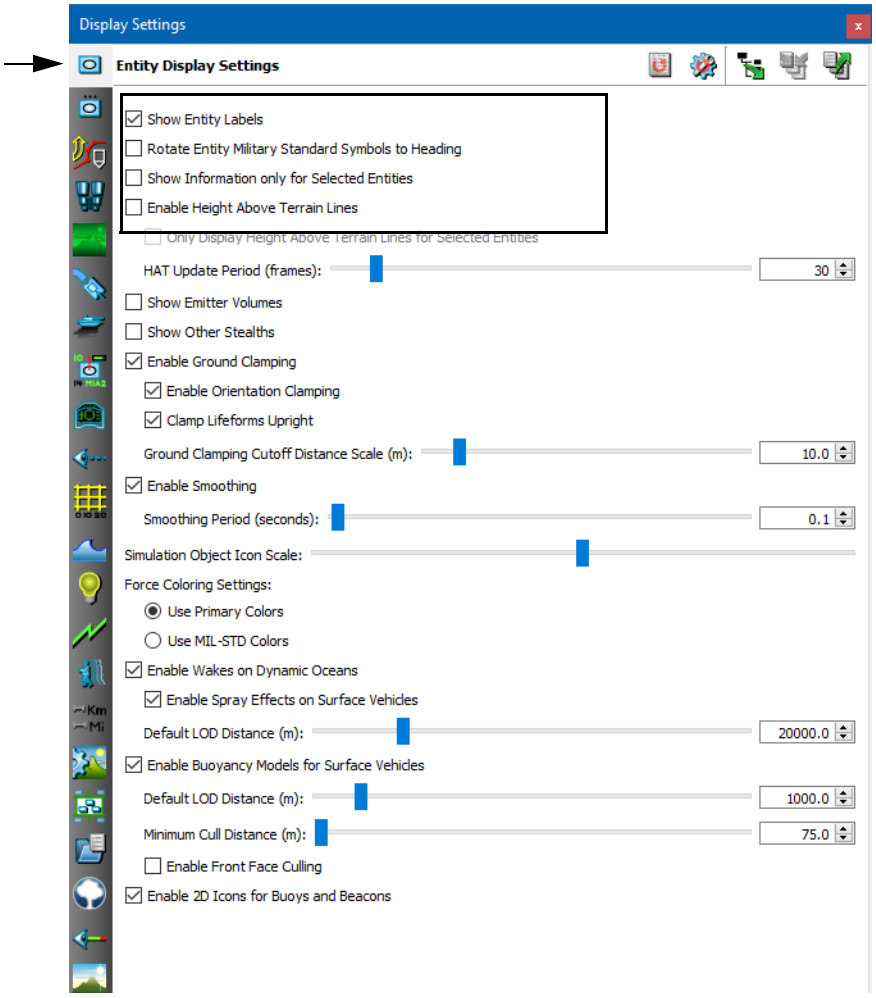


Figure 19-4. Display Settings, Entity Display Settings page

19.2.2. Turning Entity Labels On and Off

To turn the display of entity labels on or off:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page (Figure 19-4).
3. Select or clear the Show Entity Labels check box.



You can also enable and disable entity labels by choosing **Settings** → **Entity Labels**, pressing **L**, or by clicking the Entity Labels button () on the Display Settings Toolbar.

19.2.3. Pinning 3D Text Labels to the Window

You can pin 3D text labels to the window, making them stay visible after you move the cursor away from the entity.

To pin a 3D text label to the window or hide a pinned label:

1. Right-click an entity's locator circle. A menu is displayed.
2. Choose **Toggle Pinned Label**.

Pinning Multiple 3D Text Labels

You can display multiple text labels at one time.

To pin multiple text labels use either of the following methods:

- Right-click and choose **Toggle Pinned Label** for each entity whose label you want to pin.
- On the Display Settings dialog box, Entity Display Settings page, select Show Information Only For Selected Entities (Figure 19-4). Then, in the Objects List Panel, multi-select the entities whose labels you want to see.

19.2.4. Displaying Labels for 2D Icons

You can enable and disable the display of labels for 2D icons by label type. This allows you to control how much screen space is taken up by the labels. You can control the display of labels in the Display Settings dialog box or on the Display Settings Toolbar.

To enable and disable display of 2D icon labels:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Symbol Decoration Settings page (Figure 19-5).

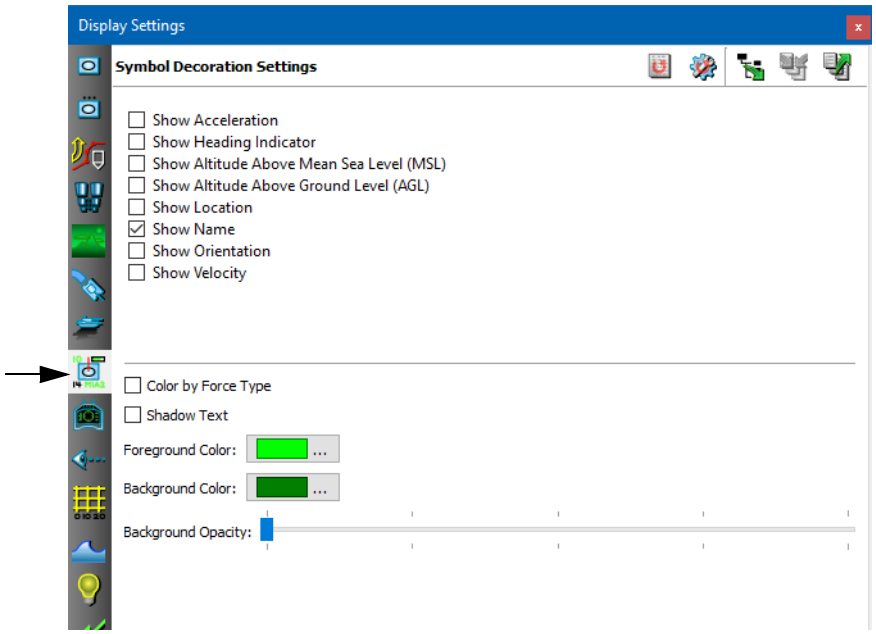


Figure 19-5. Symbol Decoration Settings page

3. Select the check box for each label that you want to display. Clear the check box for each label that you want to hide.

Changing the Color for 2D Labels

The labels for 2D icons use the color specified by the Foreground Color setting on the Symbol Decoration Settings page. You can also specify that labels use the force color.

To change the color of 2D labels:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Symbol Decoration Settings page (Figure 19-5).
3. To set the color to any color:
   a. Click the Foreground Color color swatch. A color picker dialog box opens.
   b. Select the color you want to use.
   c. Click OK.
4. To use the force color for labels, select the Color by Force Type check box. This setting overrides the Foreground Color setting.

Enabling 2D Labels on the Display Settings Toolbar

The Entity Labels button on the Display Settings Toolbar lets you open a panel that lists the symbol decorations that you can show or hide. It also lets you color labels by force type.

To enable or disable 2D labels on the Display Settings Toolbar:

1. Click the arrow on the Entity Labels button (). A small panel is displayed. It is a persistent window.

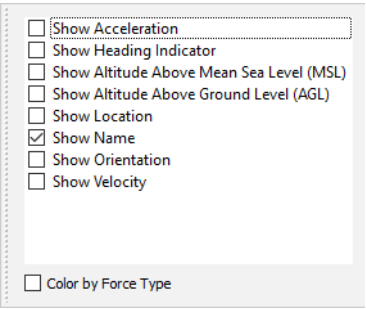


Figure 19-6. Symbol Decoration Panel

2. Select the symbol decorations that you want to display. Clear the ones you want to hide.
3. To hide the Symbol Decorations Panel, click the arrow on the Entity Labels button.

19.2.5. Configuring Shadowing and Background Color for 2D Labels

2D labels are drawn on a background that, by default, is transparent. However, you can increase the opacity of the background to make it visible and can change the background color.

You can add shadowing to the text of 2D labels. This may improve visibility when simulation objects are on light colored terrain.

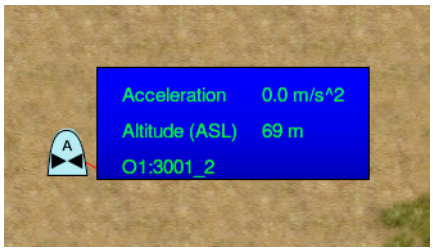


Figure 19-7. 2D label with background

To set the color and opacity of the 2D label background:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Symbol Decoration Settings page (Figure 19-5).
3. To change the opacity of the background, adjust the Background Opacity slider.
4. To change the color of the background:
   a. Click the Background Color color swatch. A color picker opens.
   b. Select the color you want to use for the background.
   c. Click OK.

To configure shadows for 2D labels:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Symbol Decoration Settings page (Figure 19-5).
3. Select or clear the Shadow Text check box.

19.3. Changing Force Colors

By default, labels, detonation lines, and other force-related simulation object graphics (including the 3D Colorized Model model set in VR-Vantage Stealth) are displayed using primary colors (red, blue, and green). You can change the force coloring to use MIL-STD 2525B colors (light blue, light green, and salmon).

To change force coloring:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page (Figure 21-1).
3. Select the Force Coloring Settings option that you want.

19.4. Displaying HAT Lines for Entities (3D Only)

Height-above-terrain (HAT) lines are vertical lines that run from an entity to the ground and display the altitude of the entity (using whatever units are configured on the Display Units Settings page) (Figure 19-8). They help you understand the relationship of the entity to the terrain.

HAT lines are shown only for entities that could be above or below the ground, such as fixed and rotary-wing vehicles. You can show and hide these lines globally and you can specify that they be shown only for the selected entities.

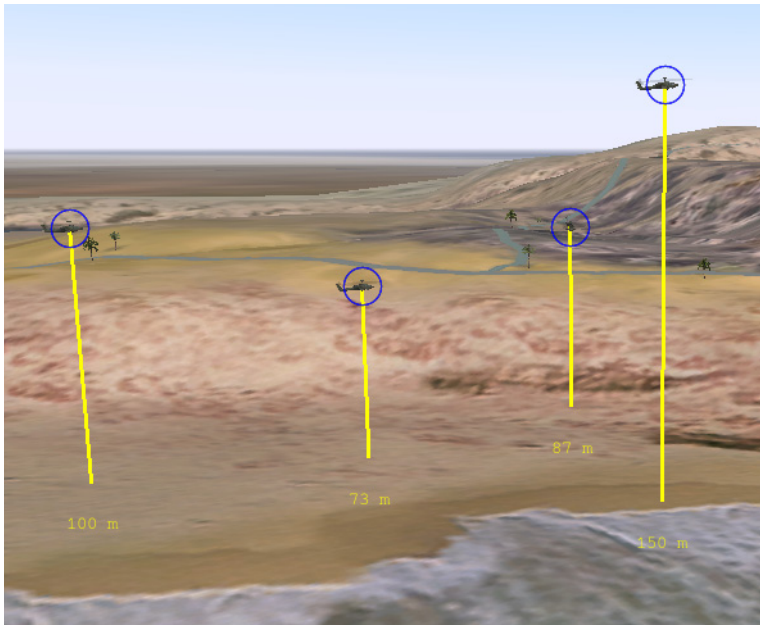


Figure 19-8. Entity height-above-terrain lines

Displaying height above terrain lines can reduce the frame rate.

To show or hide height-above-terrain lines:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page (Figure 19-4).
3. To display lines, select the Enable Height Above Terrain Lines check box.
4. To display lines only for selected entities, select the Only Display Height Above Terrain Lines For Selected Entities check box.

In VR-Vantage Stealth, you can also enable and disable entity HAT lines by choosing **Settings** → **Entity Height Above Terrain Lines**.

19.5. Changing the Model Set

Each observer mode specifies a model set. You can change the model set associated with an observer mode by editing it (for details, please see "Editing an Observer Mode," on page 14-5), but you can also dynamically change the model set being used.

➤ To change the model set, choose **Observer** → **Set Model Set** → *model_set*, where *model_set* is one of the options on the menu, or select a model set from the Set Model Set button () on the Observer Settings Toolbar. (The button changes to show the currently selected model set.)

19.6. Simulation Object Icons

The 2D Icons model set uses MIL-STD 2525B icons. Figure 19-9 illustrates basic icon shapes for friendly forces.

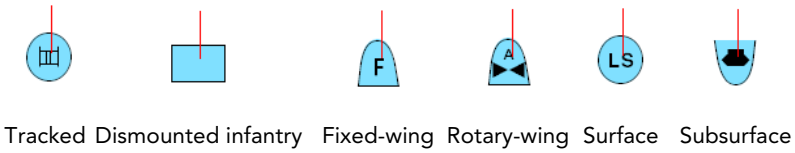


Figure 19-9. MILSTD-2525B icons

The color and shape code for 2D simulation object icons is:

- Light blue – friendly.
- Salmon – hostile.
- Light green – neutral.

Symbols have a black border except as follows:

- If there is no border on the top, the entity domain is subsurface, for example, submarines.
- Selected simulation objects have a white border.

Symbols that represent units have unit identifiers above the symbol, as illustrated in Figure 19-10.

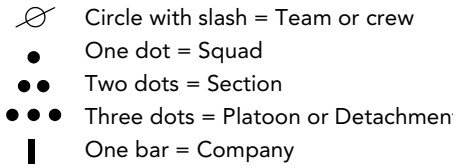


Figure 19-10. Unit echelon indicators

19.6.1. Changing the Size of 2D Icons

The Plan View observer mode (VR-Vantage Stealth and VR-Vantage PVD only) uses the 2D Icons model set for simulation objects. (In VR-Vantage Stealth, you can use these icons in the 3D observer modes too.) You can change the size of 2D icons. When many simulation objects are close together, reducing the size of the icons can reduce overlap.

To change the size of simulation object icons:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page (Figure 19-11).

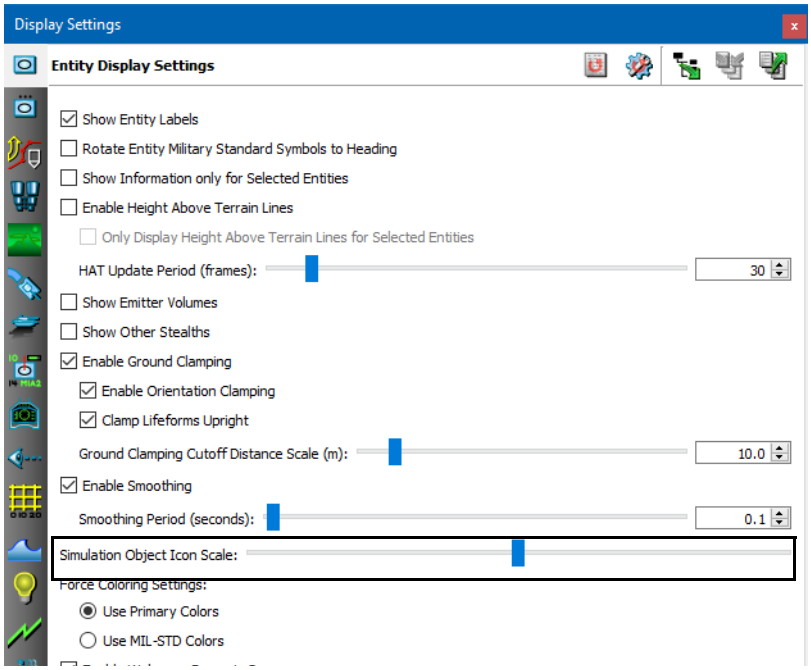


Figure 19-11. Entity Display Settings page

3. Adjust the Simulation Object Icon Scale slider.

19.6.2. Rotating 2D Icons to a Simulation Object's Heading

By default, when a simulation object's heading changes, the heading indicator points to the new heading, but the icon remains oriented towards zero degrees (North). You can specify that icons rotate to the heading of the simulation object.



Figure 19-12. 2D icon rotated to heading

To rotate icons to a simulation object's heading:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page (Figure 19-11).
3. Select the Rotate Entity Military Standard Symbols to Heading check box.

19.7. Viewing Occluded 2D Icons

If you are using VR-TheWorld or a similar terrain in the 2D projection, it is possible that simulation objects are being simulated on the opposite side of the world from the observer (they are occluded from view by the earth). Since 2D icons are displayed on a layer above the terrain, they do not get hidden from view by the terrain. VR-Vantage lets you configure the visibility of occluded entity icons. By default, occluded icons are displayed as 100% transparent. You can vary transparency from 100% transparent to 100% opaque.

To set the transparency of occluded 2D icons:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 19-13).

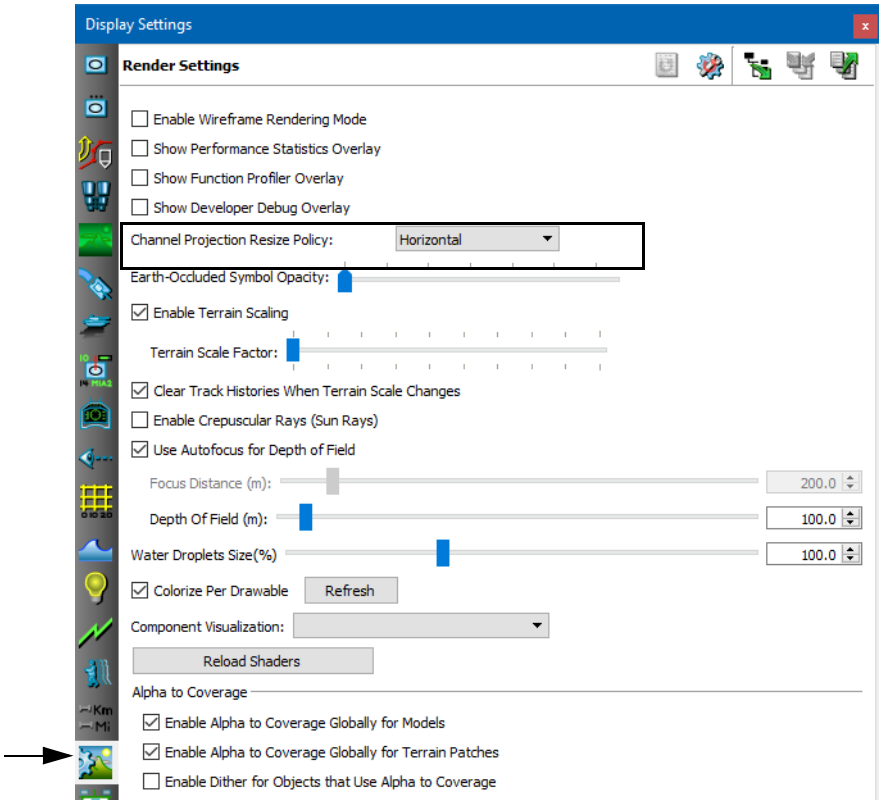


Figure 19-13. Render Settings page

3. Adjust the Earth-Occluded Symbol Opacity slider.

19.8. Hiding 3D Models

You can hide 3D models and 3D colorized models.

To hide 3D models:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 18-3).
3. In the list of observer settings, clear the Main Model check box.

19.9. Display of Embarked Entities

When VR-Vantage renders a scenario generated by VR-Forces, the scenario may include embarked simulation objects. In the 3D projection, embarked entities are displayed just like any other object. In the 2D projection, VR-Vantage does not display icons for embarked entities. However, they are still listed in the Objects List Panel.

19.10. Viewing Object Counts

You can display the number of live and destroyed simulation objects in the simulation you are viewing and the number of VR-Forces control objects, by type.

➤ To display the object counts, choose **View** → **Objects Count Panel**. The Objects Count Panel is added to the display (Figure 19-14).

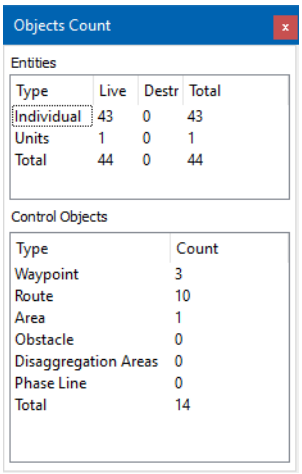


Figure 19-14. Objects Count Panel

19.11. Displaying Models for Remote Observers

VR-Vantage broadcasts the location and orientation of the observer. By default, it does not display a model for the observer. However, if there are other applications in an exercise that are broadcasting a stealth PDU (such as another VR-Vantage application), you might want to represent them with a model.

To display a model for remote observers:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page (Figure 19-15).

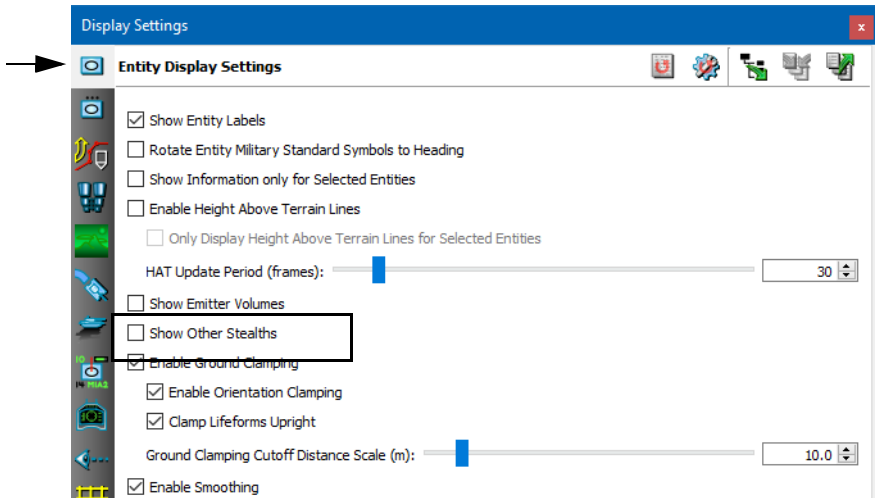


Figure 19-15. Display Settings, Entity Display Settings page

3. Select the Show Other Stealths check box.

19.12. Filtering Simulation Objects Using Interest Management

VR-Vantage can use interest management to improve its performance in simulations that have high simulation object counts. Interest management is an implementation of HLA data distribution management (DDM). When interest management is enabled, VR-Vantage filters out all simulation objects that are more than a specified distance from the observer.

19.12.1. Enabling Interest Management

To use interest management:

- You must use HLA (DIS does not support interest management).
- If you are using the MAK RTI, you must use MAK RTI 3.4a or later.
- You must configure your RTI to support DDM. (If you are using the MAK RTI, you can configure your system to use *ddm-rid.mtl*, which has DDM enabled.)
- When you connect to an exercise, you must enable DDM in the connection settings.

To enable interest management:

1. Choose **Settings** → **Connection**. The Connection Settings dialog box opens.
2. Select the HLA connections page for the version of HLA that you are using.
3. Click Advanced to show all connection parameters.
4. Select the Implicit Geographic DDM check box.
5. Specify any other information required for the simulation.
6. Click Connect.

19.12.2. Configuring Interest Management

Interest management works by filtering out all simulation objects beyond a specified distance from the observer. You can configure this distance (DDM region size) and you can configure the threshold for observer movement (region update threshold) before VR-Vantage recalculates which simulation objects should be displayed.

VR-Vantage uses a geographic DDM scheme. The earth is divided into two dimensional regions, with X pointing north and Y pointing east. When you specify a DDM region size, you specify the length of one side of a region.

The region update threshold is a percentage of the distance from the center of the region to a side. When the observer moves beyond the threshold, the region is recalculated. For example, if the region size is 1000 meters, the distance from the center to a side is 500 meters. If the region update threshold is 50%, when the observer moves more than 250 meters from the center of the region, VR-Vantage recalculates the display region.

DDM regions use a two-dimensional grid, so the altitude of the observer is not considered when filtering simulation objects. However, in 3D observer modes, you usually cannot see entity models once the observer is a certain distance above the terrain (although you can see labels). Therefore, you may want to filter out simulation objects based on the observer's altitude, thereby increasing performance.

To configure interest management:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Interest Management Settings page (Figure 19-16).

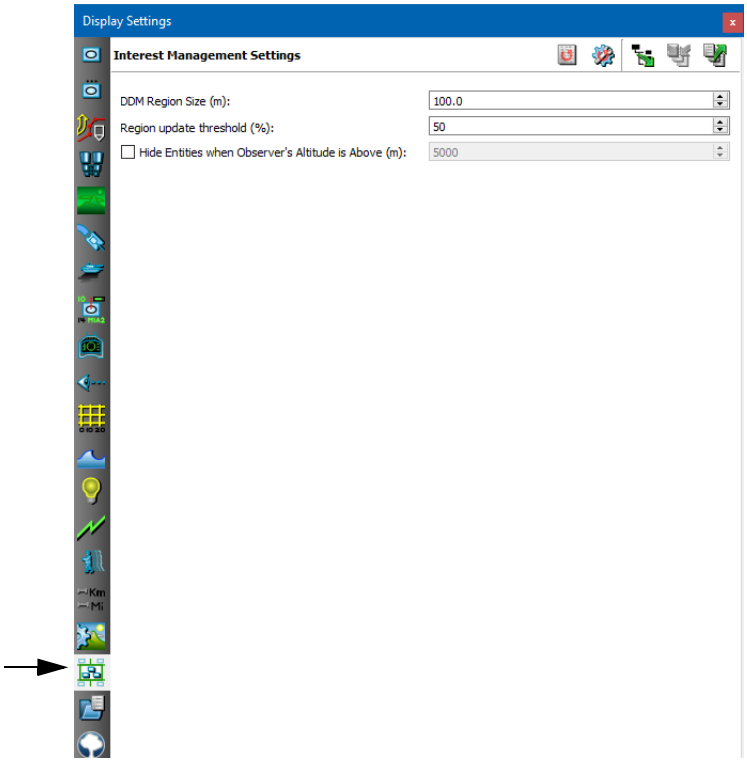


Figure 19-16. Interest Management Settings page

3. In the DDM Region Size box, type a value for the region size.
4. In the Region Update Threshold box, type a value, as a percentage, for the update threshold.
5. Optionally, filter simulation objects based on the observer's altitude, as follows:
   a. Select the Hide Entities When Observer's Altitude is Above check box.
   b. Specify the observer altitude at which to hide simulation objects.
6. Click OK.



20. Displaying Units

If you are using a VR-Vantage application to view a simulation generated by VR-Forces, MAK's computer generated forces application, you can manipulate the display of the units that are in the simulation.

Introduction to Units 20-2
Expanding and Collapsing Units 20-2
Displaying Ghosted Icons 20-5
Displaying Unit Icons and Bounding Volumes 20-8
Specifying the Color Scheme for Unit Bounding Volumes 20-10

20.1. Introduction to Units

Units are simulation objects that represent organizational groupings of entities in a military hierarchy, such as platoons, companies, and brigades. VR-Vantage Stealth and VR-Vantage PVD can represent the units simulated by VR-Forces. VR-Vantage IG has partial support for displaying units. VR-Forces can simulate units in either of the following states:

- Disaggregated: Subordinates are simulated and published as individual entities (Figure 20-1). Members of the unit are displayed using individual models or icons.
- Aggregated: Subordinates are not simulated separately. The unit is represented by one icon.

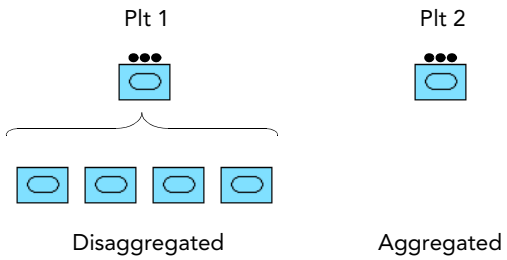


Figure 20-1. Disaggregated and aggregated entities

VR-Vantage can display disaggregated units at the unit level, using a 2D icon (collapsed), or it can display the individual members of a unit (expanded). It can also display translucent volumes that show the bounding volume of the entire unit.

Since aggregated units are sent over the network as one logical entity, VR-Vantage always represents them with one icon.



By default, the Objects List is filtered to just show entities. Therefore, if VR-Vantage is receiving units, there will be unit icons in the display, but no corresponding entry in the Objects List. To see units in the Objects List, change the filter to Units or All.

The rest of this chapter explains how to manage these different display options.

20.2. Expanding and Collapsing Units

You can display disaggregated units in expanded or collapsed mode. When a unit is collapsed, it is displayed as one icon. When a unit is expanded, all the members of the unit are visible as individual entities or subordinate units and the unit icon is hidden. Multi-level units can expand and collapse at each level of aggregation. For example a company could be collapsed to the company level (one icon), the platoon level (an icon for each platoon), or fully expanded.

VR-Vantage IG does not support expanding and collapsing of units because it does not support the 2D Icons model set.

When you Expand or Collapse a unit, you can expand or collapse one level of aggregation or all levels. When you Expand or Expand All, expansion is downward from the current unit and does not affect any sibling units. When you Collapse, contraction is upward in the hierarchy. Collapse All collapses all members of the unit to the topmost level.

Figure 20-2 illustrates these relationships. Assume a company with three platoons, each of which has four vehicles.

- If you Expand the top level unit, it expands to the platoon level. (1)
- If you select Plt C and choose Expand or Expand All, it expands just Plt C. (2)
- If you select the top level unit and expand all, it expands to the leaf level – 12 individual vehicles. (3)
- If the unit is fully expanded and you select one member of Plt C and choose Collapse, just Plt C collapses. (4)
- If you select any member of an expanded unit and select Collapse All, the entire unit collapses. (5)

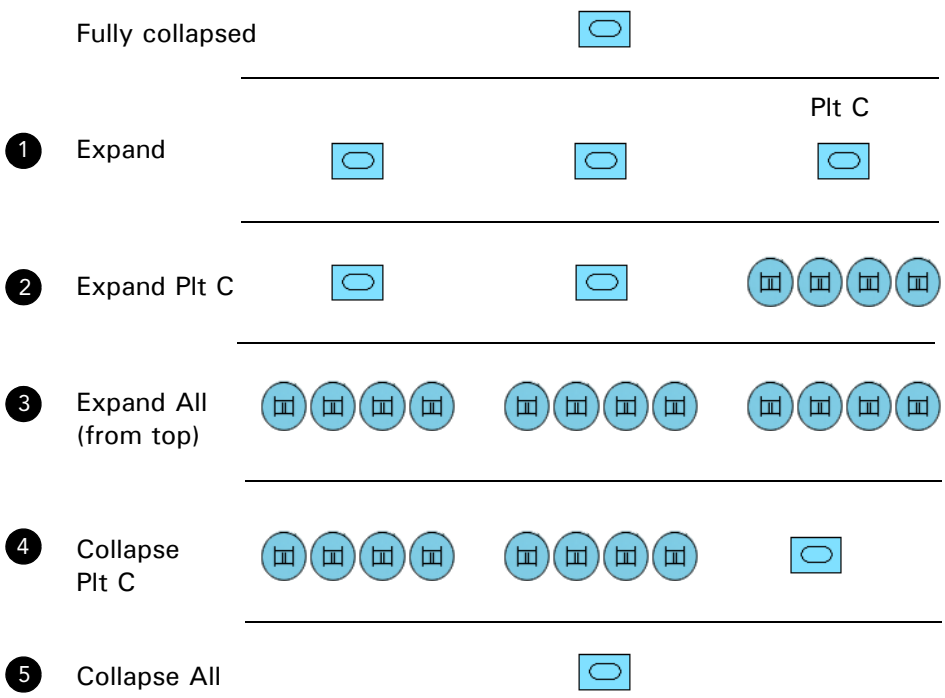


Figure 20-2. Unit expand and collapse

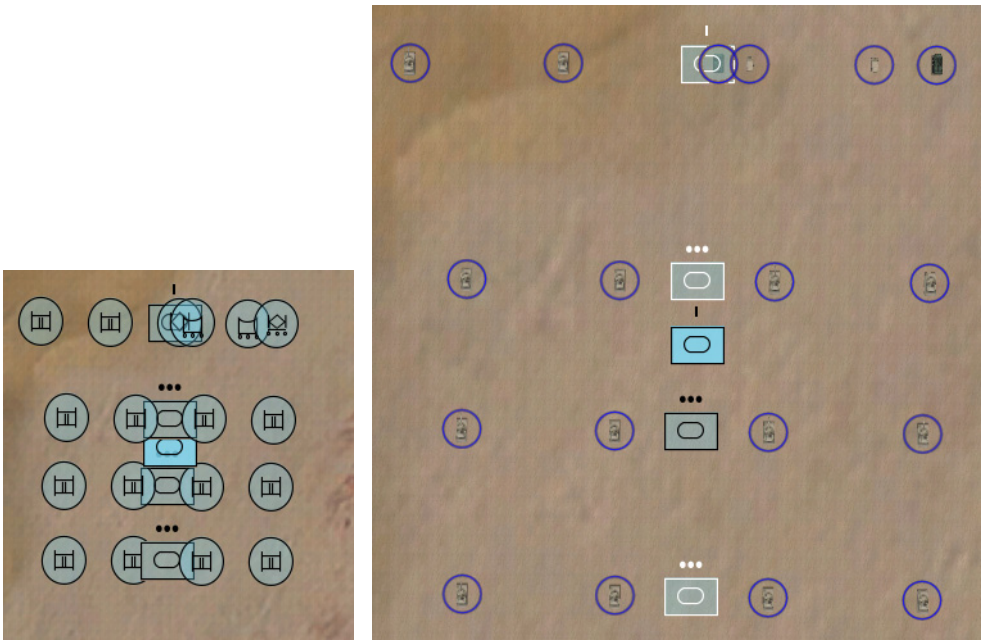
- To expand a unit, right-click the unit icon or its entry in the Objects List View of the Objects List Panel and choose **Expand** on the popup menu.
- To expand a unit and all its subordinate units, right-click the unit icon or its entry in the Objects List View of the Objects List Panel and choose **Expand All** on the popup menu.
- To collapse a unit one level, right-click any member of the unit and choose **Collapse** on the popup menu.
- To collapse a unit to its root level, right-click any member of the unit and choose **Collapse All** on the popup menu.

20.3. Displaying Ghosted Icons

If you want to select units, you need to collapse them so that you can select the unit icon. However, when you do this, you can no longer see where individual subordinate simulation objects are located and if subordinates are destroyed, you cannot see which ones are destroyed and which are functional. VR-Vantage provides the following options for displaying unit icons so that you can work with them in a way that best meets your visualization and simulation object management needs:

- Display ghosted icons for hidden simulation objects.
- Allow ghosted icons to be selected.
- Display all unit members.
- Manually expand and collapse the display.
- Display leaf-level unit members.

In the Plan View observer mode, when you display ghosted units, the unit icon is displayed normally, and the subordinate members are displayed using semi-transparent icons. In 3D observer modes, units that are subordinate to the top level unit are displayed using semi-transparent icons. Leaf-level entities are displayed using the appropriate model for the observer mode (Figure 20-3).



2D 3D

Figure 20-3. Ghosted icons

To configure the display of unit icons:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Unit Display Settings page (Figure 20-4).

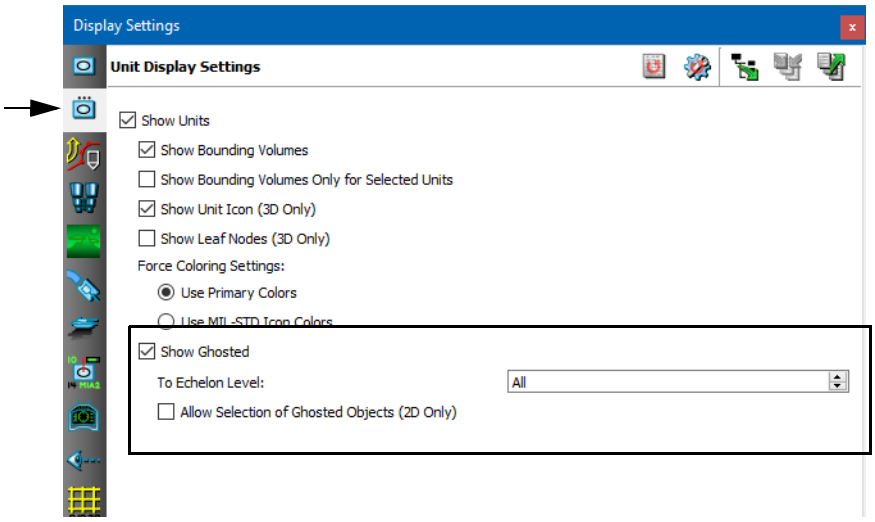


Figure 20-4. Unit Display Settings page

3. To display unit members based on the current level of aggregation only, clear Show Ghosted.
4. To display ghosted icons:
   a. Select Show Ghosted.
   b. In the To Echelon Level box, select the level of aggregation to show ghosted icons for. To show all levels without needing to consider how many levels there are, select All. Figure 20-5 illustrates the effect of changing the echelon level for displaying ghosted icons for the VR-Forces unit hierarchy shown.

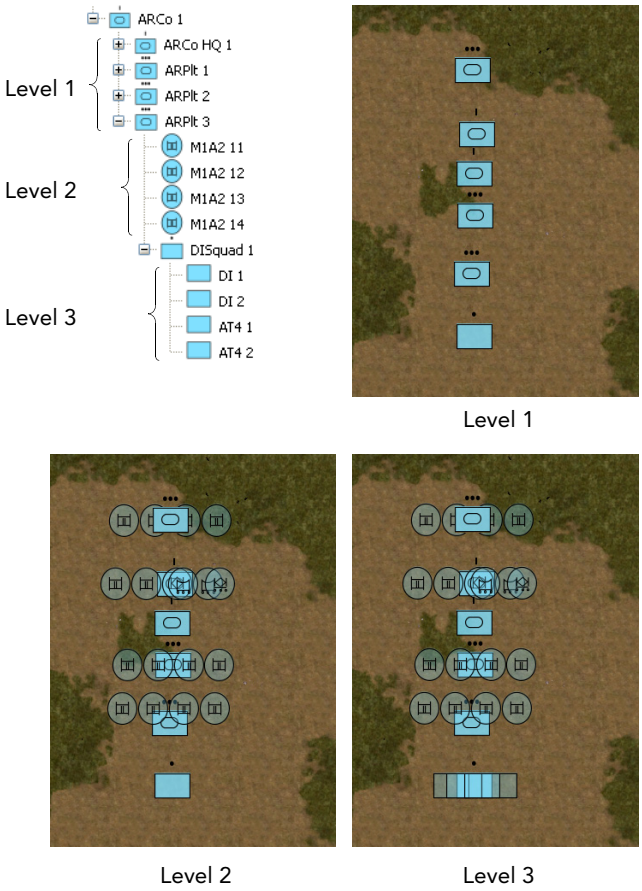


Figure 20-5. Displaying ghosted icons by echelon level

20.4. Displaying Unit Icons and Bounding Volumes

VR-Vantage lets you configure when to display unit icons and bounding volumes. Bounding volumes are translucent polygons that show the extents of the unit. This is useful when the individual entity icons or models are not displayed. You can configure icons and bounding volumes as follows:

- Enable and disable the display of units. When disabled, only the 2D icons or 3D models for leaf level unit members are displayed.
- Enable and disable the display of bounding volumes. Bounding volumes are displayed only for collapsed units. You can restrict the display of bounding volumes to the selected units.
- In 3D observer modes, enable or disable the display of unit icons.

Figure 20-6 shows a unit bounding volume in Stealth observer mode, with leaf nodes displayed.

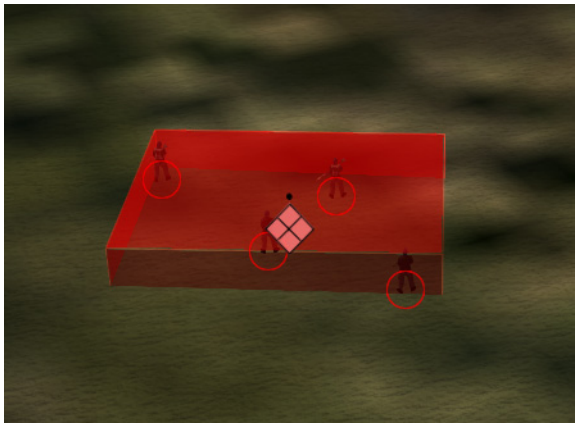


Figure 20-6. 3D bounding volume

Figure 20-7 shows a unit bounding volume in Plan View observer mode, with ghosting enabled.

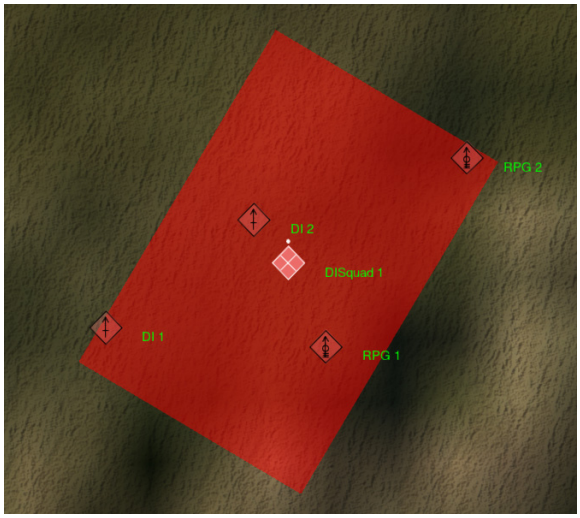


Figure 20-7. 2D bounding volume

You can display unit icons without showing bounding volumes and vice versa.

To configure the display of unit icons and bounding volumes:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Unit Display Settings page (Figure 20-4).
3. To enable the display of unit icons, bounding volumes, or both, select Show Units.
4. To display bounding volumes for collapsed units, select Show Bounding Volumes.
5. To restrict the display of bounding volumes to the selected units, select Show Bounding Volumes Only for Selected Units.
6. To display unit icons in 3D observer modes, select Show Aggregate Icon (3D Only).



You can also enable and disable units by choosing **Settings** → **Units** or by selecting the Units button ( ) on the Display Settings Toolbar.

20.4.1. Specifying the Color Scheme for Unit Bounding Volumes

Unit bounding volumes can be displayed using primary colors or MIL-STD colors.

To specify the color scheme to use for unit bounding volumes:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Unit Display Settings page (Figure 20-4).
3. Select the Force Coloring Settings option that you want (Use Primary Colors or Use MIL-STD Icon Colors).



VI. Visual and Audio Effects

VR-Vantage has many visual effects that add to your understanding of the interactions between simulation objects. These include fire and detonation lines, communications lines, and emitter volumes. You can test for line-of-sight with the intervisibility feature. You can also display many different lighting effects.



21. Viewing Entity Effects

This chapter describes how to display various types of entity-related graphics.

Using Ground Clamping 21-2
Configuring Ground Clamping 21-3
Trajectory Smoothing 21-4
Configuring Trajectory Smoothing 21-5
Displaying Trailing and Decal Effects (3D Only) 21-6
Specifying Models for Trailing Effects and Decal Effects 21-8
Displaying Track Histories 21-8
Configuring Track Histories 21-9
Viewing Fire and Detonation Effects 21-11
Viewing Fire and Detonation Lines 21-12
Displaying Tactical Smoke 21-13
Displaying Electromagnetic Emissions 21-14
Displaying Radio Communication Lines 21-15
Enabling Radio Communications Graphics 21-16
Configuring the Lifetime and Display Mode 21-17
Configuring the Color of Radio Communication Lines 21-18
Using the Depth of Field Effect 21-20
Configuring Depth of Field 21-22
Displaying Wakes and Spray Effects 21-23
Enabling Buoyancy for Surface Entities 21-24
Displaying Laser Designators 21-26

21.1. Using Ground Clamping

Because of differences in terrain databases among exercise participants, entities can sometimes appear to be underground or hovering above the terrain surface. To compensate, VR-Vantage can clamp land and sea entities to the terrain surface. When ground clamping is enabled, VR-Vantage keeps an entity anchored to the terrain surface regardless of the altitude data contained in its state update.

Use of ground clamping can reduce your frame rate. If you correlate the simulation and visualization terrain databases, you can eliminate the need to use ground clamping.

To enable or disable ground clamping:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page (Figure 21-1).

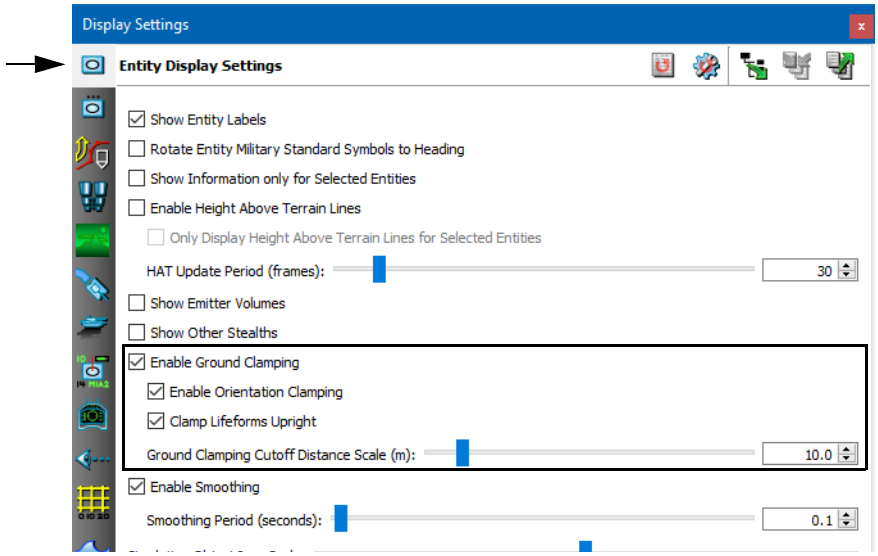


Figure 21-1. Ground clamping settings

3. Select or clear the Enable Ground Clamping check box.

You can also enable and disable ground clamping by choosing **Settings** → **Ground Clamping**.

21.1.1. Configuring Ground Clamping

You can configure the following aspects of ground clamping:

- Enable or disable ground clamping.
- Enable or disable orientation clamping.
- Force lifeforms to remain upright relative to the plane of their location when moving up slopes or steps, rather than being perpendicular to the terrain.
- The distance from the terrain within which entities are clamped.

To configure ground clamping:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page (Figure 21-1).
3. To enable or disable a ground clamping option, select or clear its check box.
4. To specify the distance from the terrain in which entities are ground clamped, adjust the Ground Clamping Cutoff Distance Scale slider or type a value, in meters, in the box.



The setting for the Ground Clamping Cutoff Distance Scale maximizes at 100 meters. If you set it to the maximum, the distance is infinite and all entities are ground-clamped.

21.2. Trajectory Smoothing

From the time VR-Vantage receives a state update message for a simulation object until it receives the next update, it calculates the location of the simulation object based on the heading and acceleration in the most recent state update. This process is called dead-reckoning. When the next state update arrives for the simulation object, VR-Vantage moves it to the new position. If the actual position is significantly different from the dead-reckoned position, the simulation object could appear to jump from the dead-reckoned position to the actual position. VR-Vantage can eliminate these discontinuities by smoothing trajectories and rotation paths.

Figure 21-2 illustrates dead-reckoning and smoothing.

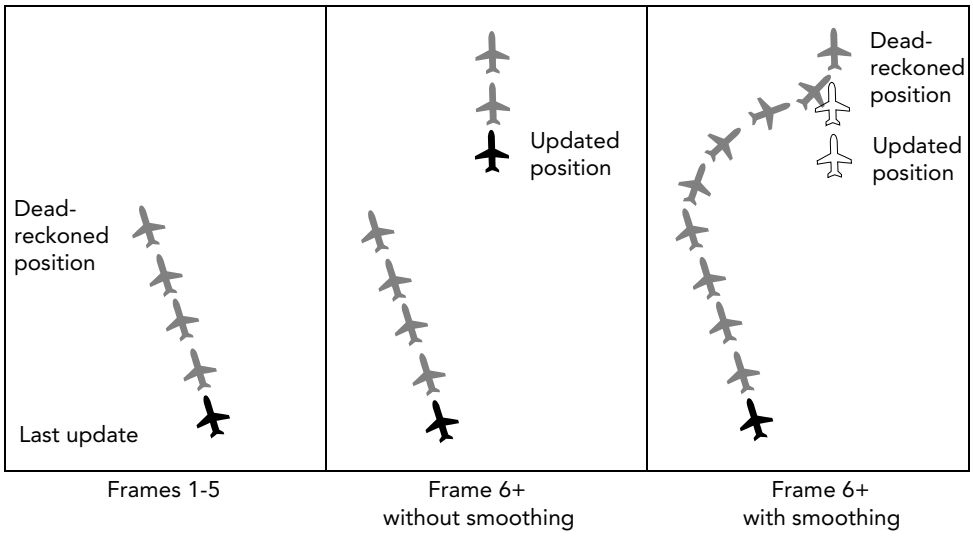


Figure 21-2. Dead-reckoning and smoothing

The smooth period is the period of time over which smoothing takes place. The default period is 1/10 second. If course corrections appear abrupt with this setting, you may want to increase the smooth period.



Smoothing makes fast-moving entities, such as fixed wing aircraft, look better. Slow moving entities, such as dismounted infantry, may look worse. If your simulation is mostly one or the other of these types of entities, enable or disable smoothing as appropriate for the best visualization experience.

21.2.1. Configuring Trajectory Smoothing

To configure trajectory smoothing:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page (Figure 21-3).

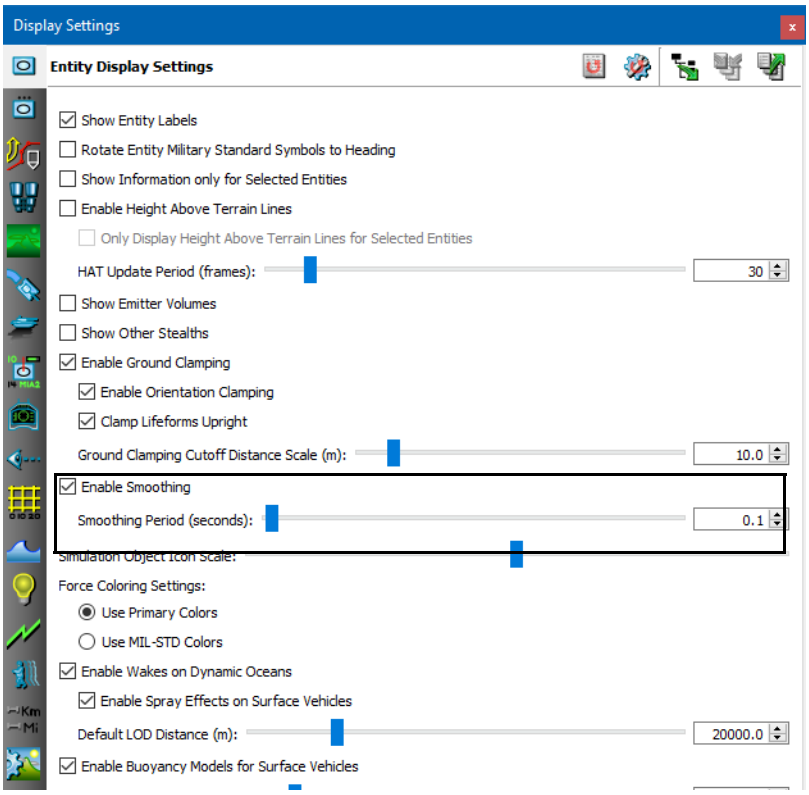


Figure 21-3. Smoothing settings

3. To enable or disable smoothing, select or clear the Enable Smoothing check box.
4. To change the smooth period, adjust the Smoothing Period slider or type a value, in seconds, in the box.

21.3. Displaying Trailing and Decal Effects (3D Only)

VR-Vantage supports trailing effects such as contrails for missiles and aircraft and dust clouds for vehicles. It also supports decal effects, such as tire tracks, footprints, and wakes. In essence, trailing effects are effects that are dispersed through the air, while decal effects are effects that are printed on the ground.

Trailing effects and decal effects are properties of an observer mode. You can enable and disable them on a per-observer basis.

Figure 21-4 illustrates some trailing effects.



Figure 21-4. Tank tracks, dust cloud

To enable or disable trailing effects or decal effects:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 21-5).

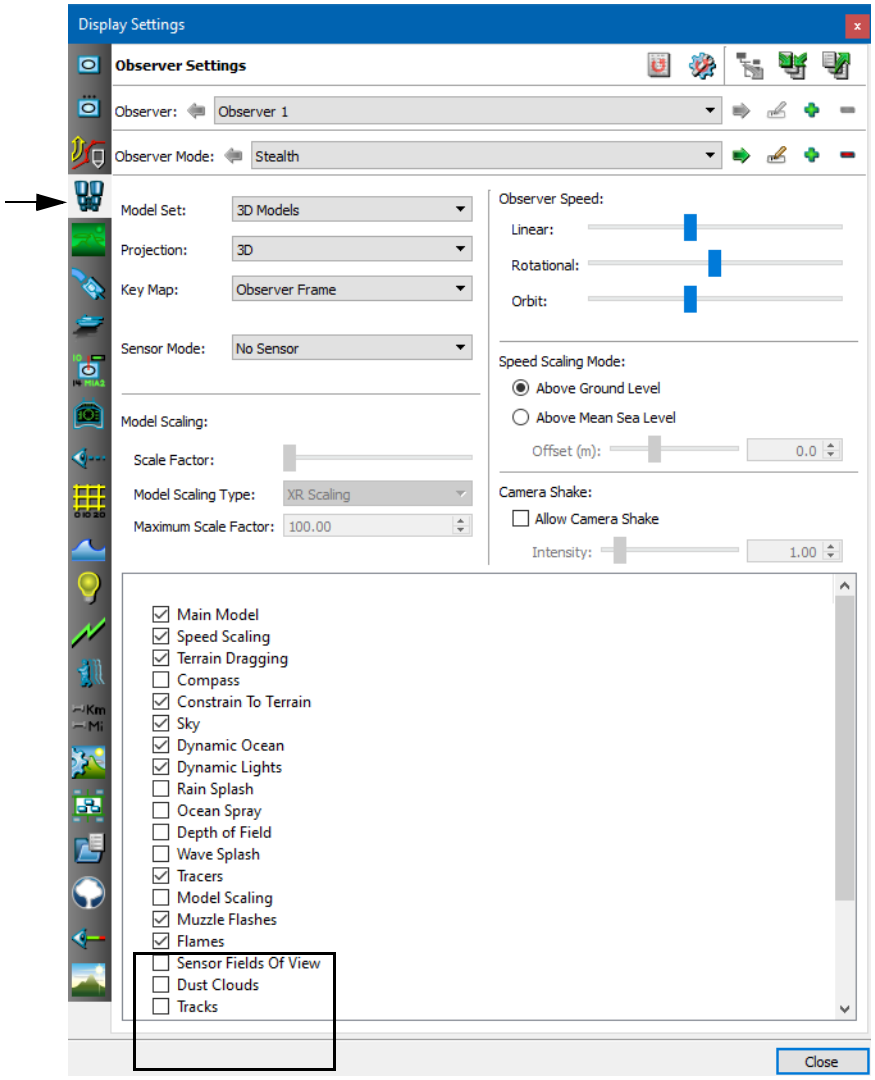


Figure 21-5. Display Settings dialog box, Observer Settings

3. Select or clear the check box for each effect that you want to change.



You can also enable and disable trailing and decal effects by choosing **Observer** → *effect* or clicking the appropriate button on the Observer Settings Toolbar.

21.3.1. Specifying Models for Trailing Effects and Decal Effects

Trailing effects and decal effects are configured through model definitions. (For the purposes of configuration, trailing effects are categorized as decals.) Each effect must have a model definition that uses a decal schema. Then, in an entity's entity definition, you configure a visualizer for the effect. For details about creating model definitions, please see "Creating and Editing Model Definitions," on page 32-2.

21.4. Displaying Track Histories

Track histories allow you to track the paths of simulation objects during an exercise, including the path of munitions as they travel from the shooter to the target. When track histories are enabled, each simulation object leaves a track ribbon. The ribbon is colored based on the simulation object's force. Figure 21-6 illustrates track histories.

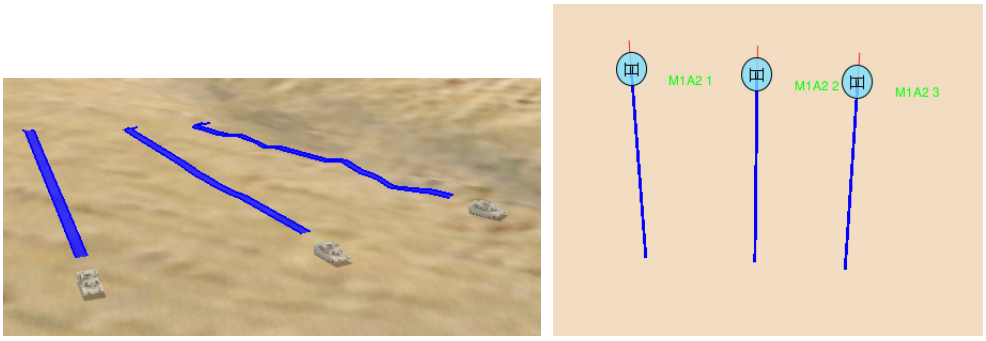


Figure 21-6. Track history

Track histories are a property of an observer mode. You can enable and disable them on a per-observer basis.

> Track histories have a significant performance and memory cost. For simulations with large simulation object counts, turning track histories off can improve performance and memory usage.

To enable or disable display of all track histories:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 21-5).
3. Select the observer mode for which you want to change the Track Histories setting.
4. Select or clear the Track Histories check box.



You can also enable or disable track histories by choosing **Observer** → **Track Histories** or clicking the Track Histories button ( ) on the Observer Settings Toolbar.

21.4.1. Configuring Track Histories

You can configure how long track history ribbons are drawn. A track history is composed of a series of segments, joined together to form a line. You can configure how long each segment is and how many segments at a time can be displayed. Drawing shorter segments improves the accuracy of the track history, but can degrade performance. Increasing the maximum number of segments provides a longer history, but can clutter up the screen if there are many simulation objects and can also affect performance. Figure 21-7 illustrates the visual result of drastically changing the segment length. The left track history uses the default segment length of 0.5 meters. The right track history has a segment length of 5000 meters.



0.5 M segment length

5000 M segment length

Figure 21-7. Track history segment length

When a track history reaches the maximum number of segments configured, it starts dropping segments from the beginning of the history (the oldest segments).

You can specify that track histories have an infinite length. In this case they never drop segments.

If you specify infinite length for track histories, it is possible that you could eventually use up all the memory in your computer and the program would crash.

To configure track histories:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Track History Settings page (Figure 21-8).

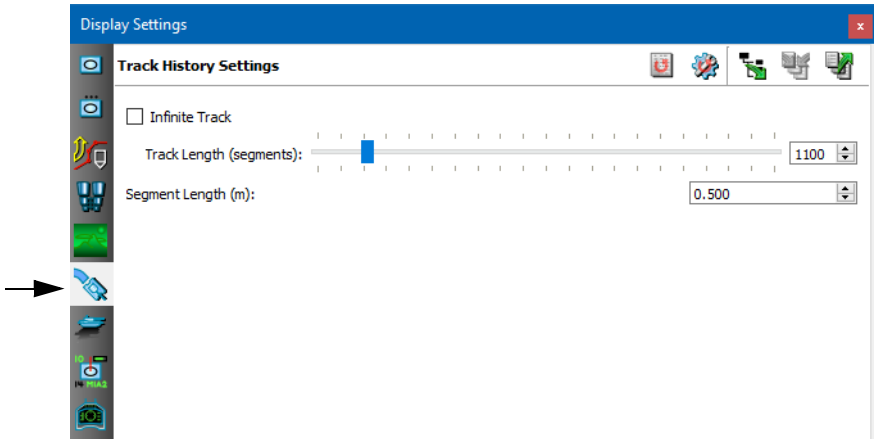


Figure 21-8. Track History Settings page

3. If you want track histories to have an indefinite length, select the Infinite Track check box.
4. To change the number of segments that can be displayed for a track history, adjust the Track Length slider, or enter a value in the box.
5. To specify the length of each track history segment, enter a value in the Segment Length box.

21.5. Viewing Fire and Detonation Effects

VR-Vantage can display the following graphic effects when munitions are fired and detonated:

- Flaming effects.
- Smoke plumes.
- Detonations.
- Muzzle flashes.
- Tracers. Tracers are displayed if the munition in a fire interaction is a tracer munition and the munition is mapped to a tracer model. Tracers are not supported in 2D observer modes.

To enable fire and detonation effects:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 21-5).
3. Select or clear the appropriate check boxes.

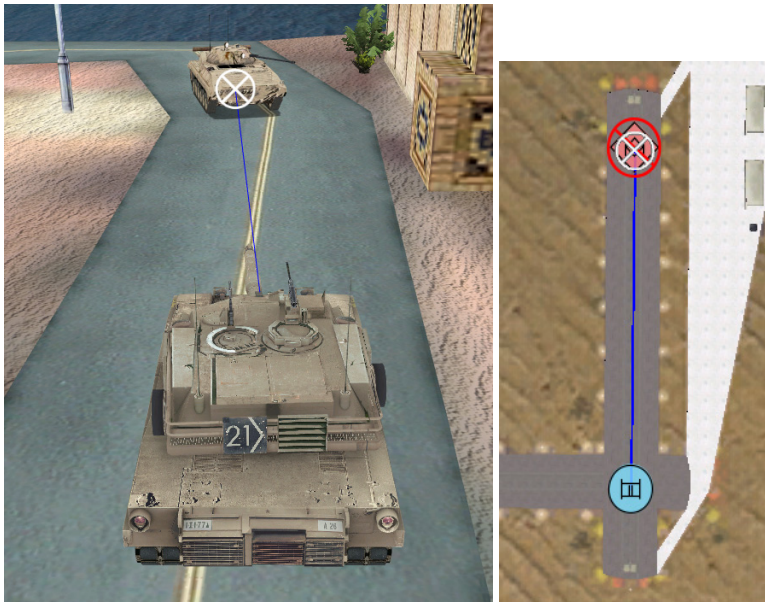


You can also enable or disable display of a fire or detonation effect by choosing **Observer** → *effect*, where *effect* is Flames, Smoke Plumes, Detonations, Muzzle Flashes, or Tracers, or click the appropriate button on the Observer Settings toolbar.

21.6. Viewing Fire and Detonation Lines

If fire and detonation lines are enabled, when an entity fires a munition, a line is drawn between the entity and the location of the detonation. The line is colored for the force of the shooter (Figure 21-9).

Fire and detonation lines are not supported in VR-Vantage IG.



3D 2D

Figure 21-9. Fire line and detonation

A detonation is identified by a circle with an X through it. The detonation color indicates the result of the detonation, as follows:

- Entity impact – white.
- Ground impact – gray.
- Airburst – black.
- Other – white.

➤ To enable or disable display of fire and detonation lines, choose **Settings** → **Fire / Detonate Lines** or click the Fire/Detonate Lines button () on the Display Settings Toolbar.

21.7. Displaying Tactical Smoke

VR-Vantage can display tactical smoke munitions fired by entities. You may want to disable tactical smoke if it obscures the observer's view. Also, if many entities are generating smoke, it could affect performance.



Do not confuse tactical smoke and smoke plumes. Smoke plumes are a visual effect displayed when an entity is on fire. Tactical smoke visualizes a smoke munition.

To enable tactical smoke:

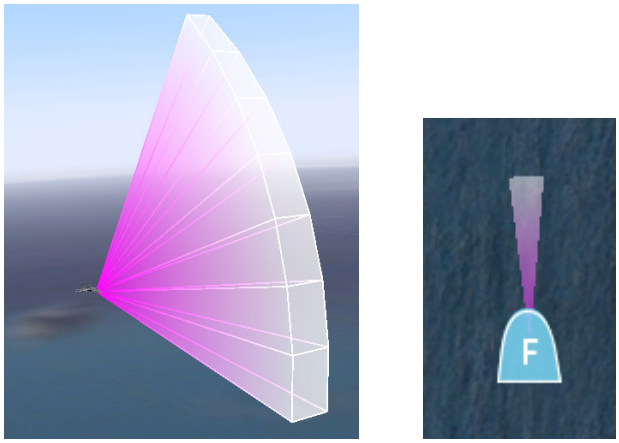
1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 21-5).
3. Select or clear the Tactical Smoke check box.



You can also enable or disable display of a fire or detonation effect by choosing **Observer** → **Tactical Smoke**, or clicking the Tactical Smoke button ( )on the Observer Settings toolbar.

21.8. Displaying Electromagnetic Emissions

VR-Vantage can display electromagnetic emissions (emitter volumes) for entities that have emitter systems (Figure 21-10). You can configure the colors used to represent emitter frequencies and you can configure the size of the emitter volumes.



3D 2D

Figure 21-10. Emitter volumes

To enable or disable display of electromagnetic emissions:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page (Figure 21-11).

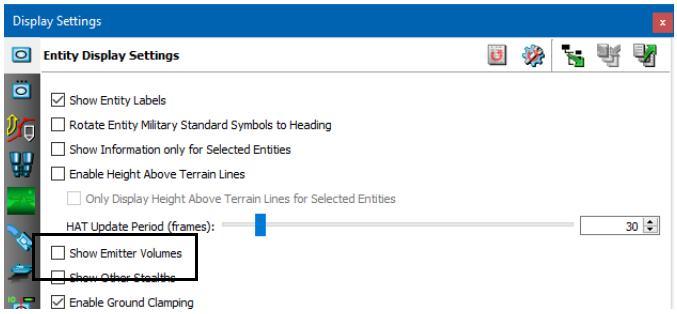


Figure 21-11. Entity Display Settings page

3. Select or clear the Show Emitter Volumes check box.



You can also enable or disable emitter volumes by choosing **Settings** → **Emitter Volumes**.

21.9. Displaying Radio Communication Lines

VR-Vantage can display squawk indicators when a simulation object sends a signal interaction. It can also display communication lines between simulation objects when a VR-Forces simulation object (or a simulation object in an application built with the appropriate MAK libraries) sends a radio message to another simulation object. You can configure a communication line's color and lifetime. Figure 21-12 displays squawk indicators and a line-of-sight communication line.



Figure 21-12. Radio communication line and squawk indicators

For communications over large distances, a line-of-sight communication line would travel through the earth. Therefore, you can also display communication lines drawn with the curvature of the earth (Figure 21-13).

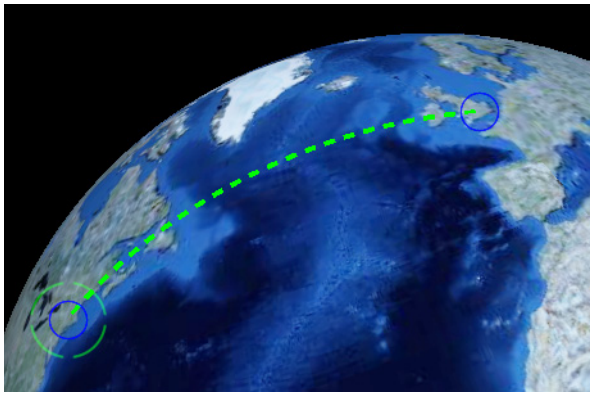


Figure 21-13. Curvature of the earth communication line

21.9.1. Enabling Radio Communications Graphics

You can globally enable or disable the ability to display radio communication lines. When they are enabled, you can enable and disable communication lines and squawk indicators separately.

To enable or disable radio communications graphics:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Radio Communications Settings page (Figure 21-14).

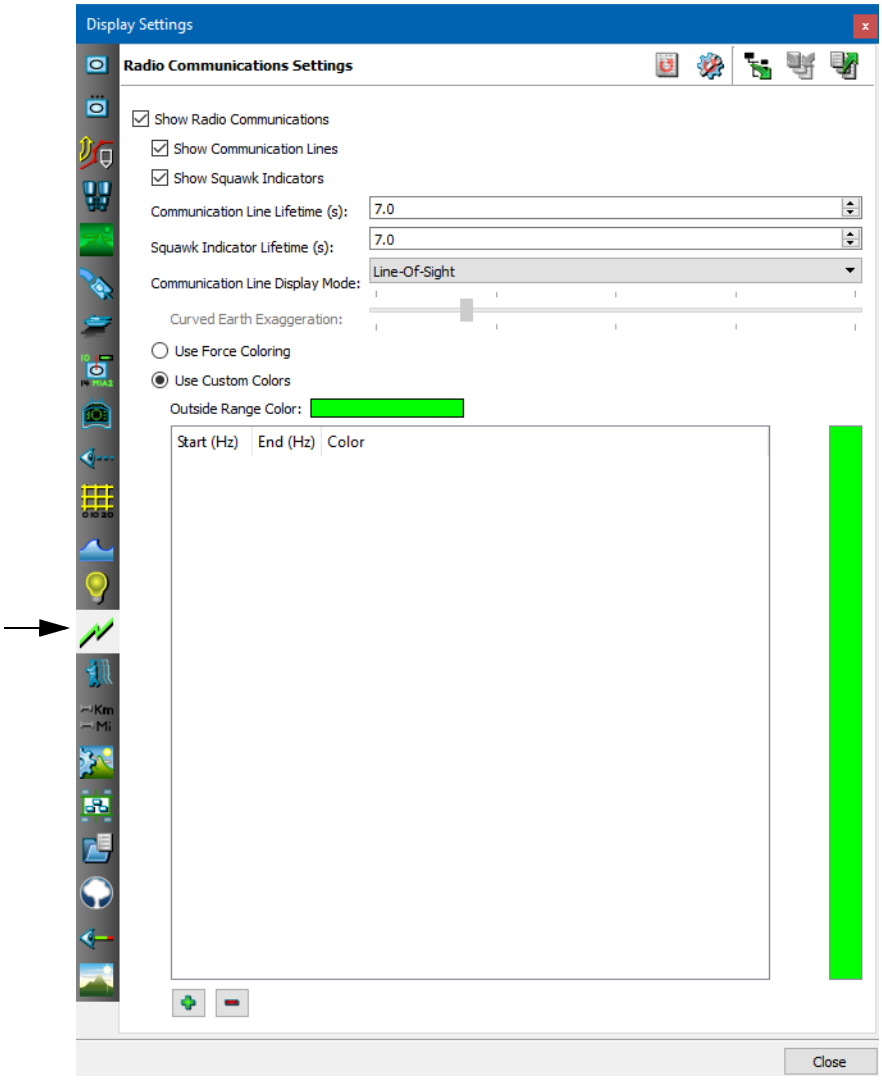


Figure 21-14. Radio Communications Settings page

3. Select or clear the Show Radio Communications check box.



You can also enable and disable communication lines by choosing **Settings → Radio Communication Display**.

To enable or disable communication lines:

1. Enable radio communication lines.
2. Select or clear the Show Communication Lines check box.

To enable or disable squawk indicators:

1. Enable radio communication lines.
2. Select or clear the Show Squawk Indicators check box.

21.9.2. Configuring the Lifetime and Display Mode

You can configure how long a radio communication line is displayed. You can also configure whether it is displayed using line-of-sight or curvature of the earth.

To configure the lifetime and display mode for radio communication lines:

1. Choose **Settings → Display**. The Display Settings dialog box opens.
2. Select the Radio Communications Settings page (Figure 21-14).
3. To specify how long a communication line or squawk indicator should be displayed, enter a value, in seconds, in the Communication Line Lifetime and Squawk Indicator Lifetime fields.
4. To specify the display mode, select an option in the Communication Line Display Mode list.
5. If you select Curvature of the Earth, adjust the Curved Earth Exaggeration slider to change the degree to which the curvature is exaggerated.

21.9.3. Configuring the Color of Radio Communication Lines

By default, radio communication lines are green. You can configure communication lines to use force coloring or to be colored based on the frequency of the message. When you color by frequency, you specify colors for the frequency ranges you are interested in and a color to use for all messages outside the specified frequencies.

To configure the color of radio communication lines:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Radio Communications Settings page (Figure 21-14).
3. To color radio communication lines using the force color, select the Use Force Coloring option.
4. To color radio communication lines by frequency:
   a. Select the Use Custom Colors option.
   b. Click the Add button (). A range is added to the frequency list with a default color (Figure 21-15).

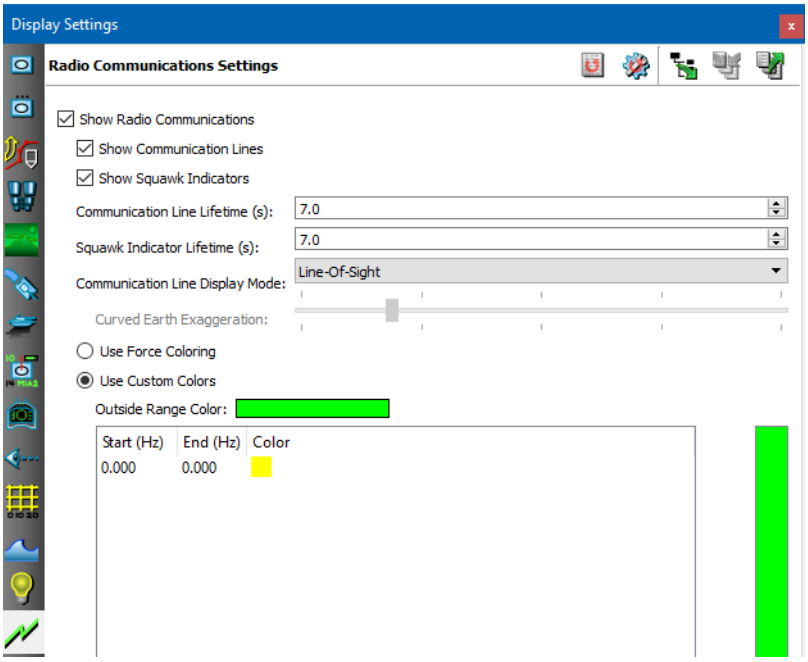


Figure 21-15. New frequency point

   c. Double-click the starting value of the range to make it editable (Figure 21-16).

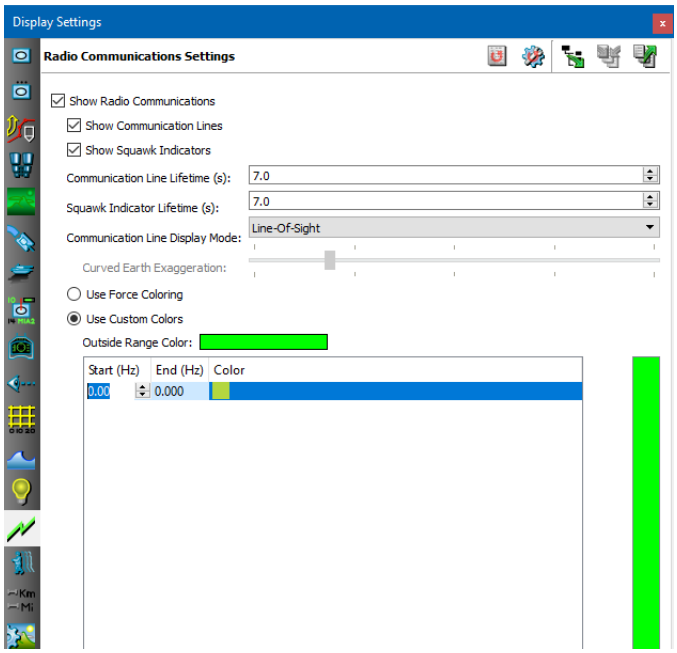


Figure 21-16. Editable frequency point

d. Type the starting frequency you want to use.

e. Edit the end frequency.

f. Double-click the color swatch and select a color for this range.

g. Optionally, add points for different frequency ranges.

h. Optionally, click the Outside Range Color color swatch and specify a color for all frequencies not specified in the frequency list.



If you configure multiple frequency ranges, you cannot add a range whose values are within those of an existing range.

21.10. Using the Depth of Field Effect

Depth of field controls the area in the scene that is in focus. Depth of field is calculated based on a focal depth and a focal range. The focal depth is the distance from the observer that is the center (focal point) of the area in focus. The focal range is the distance from the focal point that is in focus.

In Figure 21-17, the focal depth is about 10 meters, fairly close to the observer, and the focal range is also small, so that the scene is out of focus just past the car (note that the first large tree is out of focus). In Figure 21-18, the focal depth is the same, but the range is much greater, so the scene is in focus to the back of the parking lot.

In Figure 21-19 the focal depth is centered at the second large tree. The focal range is, again, fairly narrow, so that the person and car in the foreground are out of focus.

VR-Vantage lets you set the focal depth automatically or manually. In either case, you can adjust the focal range.



Figure 21-17. Depth of field (close)



Figure 21-18. Depth of field (far)



Figure 21-19. Depth of field, middle distance

To enable depth of field:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 21-5).
3. Select the observer mode for which you want to enable depth of field.
4. Select or clear the Depth of Field check box.

21.10.1. Configuring Depth of Field

You can enable and disable automatic depth of field and you can adjust the focal depth and focal range.

To configure depth of field:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 21-20).

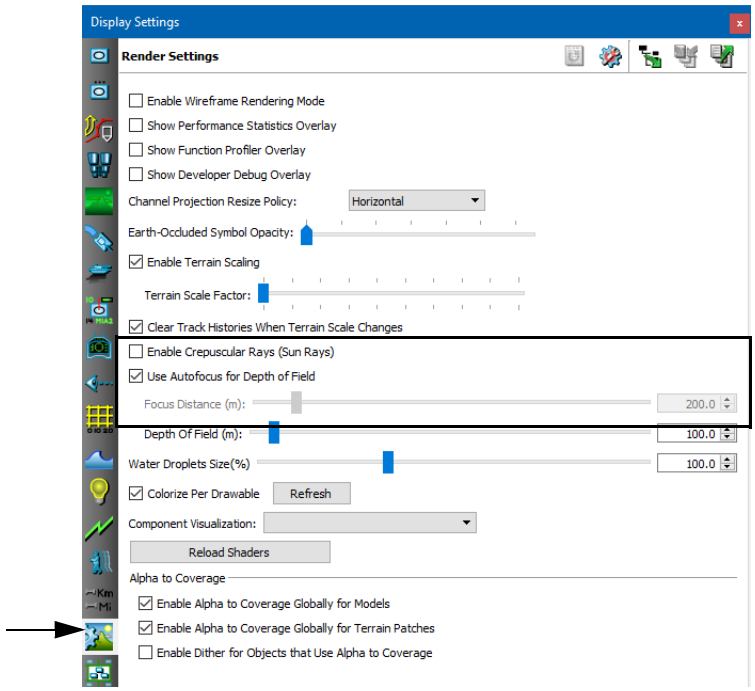


Figure 21-20. Depth of Field settings

3. To use autofocus for focal depth, select the Use Autofocus for Depth of Field check box.
4. To set the focal depth manually, clear the Use Autofocus for Depth of Field check box and adjust the Focus Distance slider or specify a value in the box.
5. To adjust the focal range, adjust the Depth of Field slider or specify a value in the box.

21.11. Displaying Wakes and Spray Effects

When dynamic ocean is enabled, surface entities display wakes and spray effects. However, these effects affect performance. You can disable them if you need to emphasize performance over visual effects.

Wakes have their own schema, with many attributes, including length, offset, the amount of time over which they fade out, and LOD distance. You can edit these attributes for individual entities in the Visual Model Editors dialog box.

To enable or disable wakes and spray effects:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page (Figure 21-21).

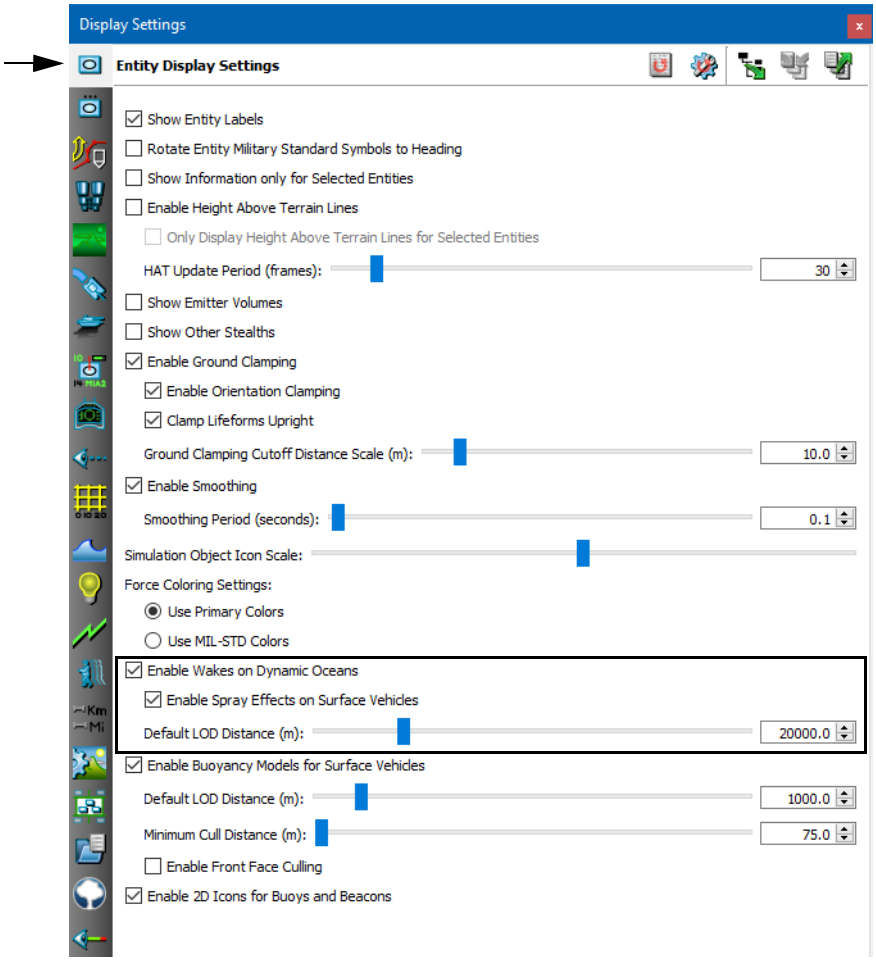


Figure 21-21. Entity Display Settings page

3. To enable or disable wakes, select or clear the Enable Wakes on Dynamic Oceans check box, or toggle the Wake button ( ) on the Display Settings Toolbar.

4. If wakes are enabled, select Enable Spray Effects on Surface Vehicles to enable spray effects. Clear the check box to disable them. Disabling wakes also disables spray effects.

21.12. Enabling Buoyancy for Surface Entities



When dynamic ocean is enabled, surface entities can bob up and down with wave action. However, the buoyancy models affect performance. You can disable buoyancy if you need to emphasize performance over visual effects. You can also control the area in which buoyancy models are calculated.

If buoyancy is enabled, VR-Vantage calculates buoyancy only for models within certain distances of the observer, as follows:

- Buoyancy models are always calculated for entities within the minimum cull distance. This ensures that if the observer turns, entities that are relatively near will show buoyancy immediately.
- Buoyancy models are not calculated for entities that are beyond the maximum buoyancy LOD range. You can set this value on a per-entity basis in its entity model definition. If it is not set in the model definition, VR-Vantage uses the default LOD distance specified on the Entity Display Settings page. If the default distance is set to 0.0, then entities that do not have an LOD range set in their model definitions will have an infinite buoyancy range (although buoyancy still will not be calculated past the far clipping plane).
- Buoyancy models are not calculated for entities beyond the far clipping plane or nearer than the near clipping plane.



VR-Vantage uses the nearer of the far clipping plane or the maximum buoyancy LOD range to determine the distance at which it stops calculating buoyancy models.

When buoyancy is enabled, you can specify that VR-Vantage use face front culling. VR-Vantage calculates buoyancy models for all entities within the minimum range and all entities between the near clipping plane and the far clipping plane or maximum buoyancy LOD range, whichever is closest. In Figure 21-22, the shaded area is the area in which buoyancy models are calculated.

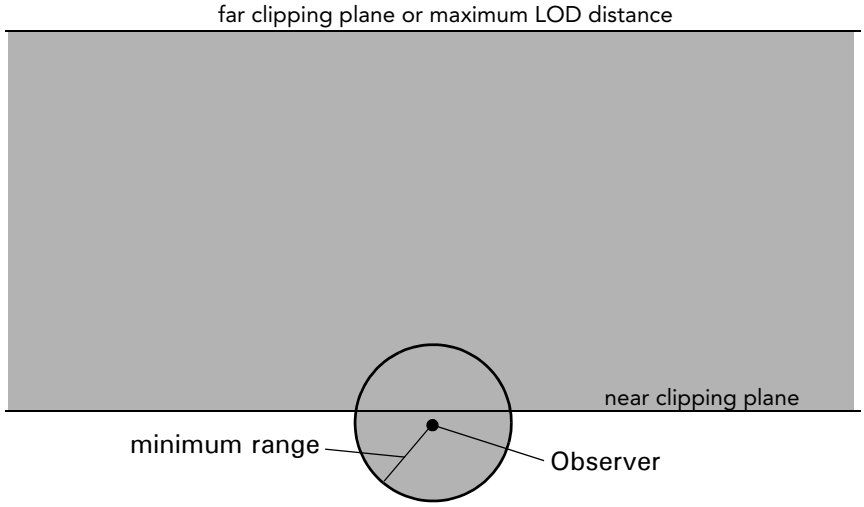


Figure 21-22. Face front culling

By default, face front culling is disabled and only distance culling is used.

To enable or disable the buoyancy models:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page (Figure 21-21).
3. Select or clear the Enable Buoyancy Models for Surface Vehicles check box.
4. To specify the distance beyond which buoyancy models are not calculated, adjust the Default LOD Distance slider or change the value in the box. (If the default distance is set to 0, then entities that do not have an LOD range set in their model definitions will have an infinite buoyancy range. However, buoyancy will not be calculated past the far clipping plane.)
5. To specify the radius within which all entities have their buoyancy models calculated, adjust the Minimum Cull Distance slider or change the value in the box.
6. Optionally, specify use of face front culling.

21.13. Displaying Laser Designators

You can display laser designation information if it is available. VR-Vantage displays a green circle and a range (on top of the entity icon or model) (Figure 21-23).

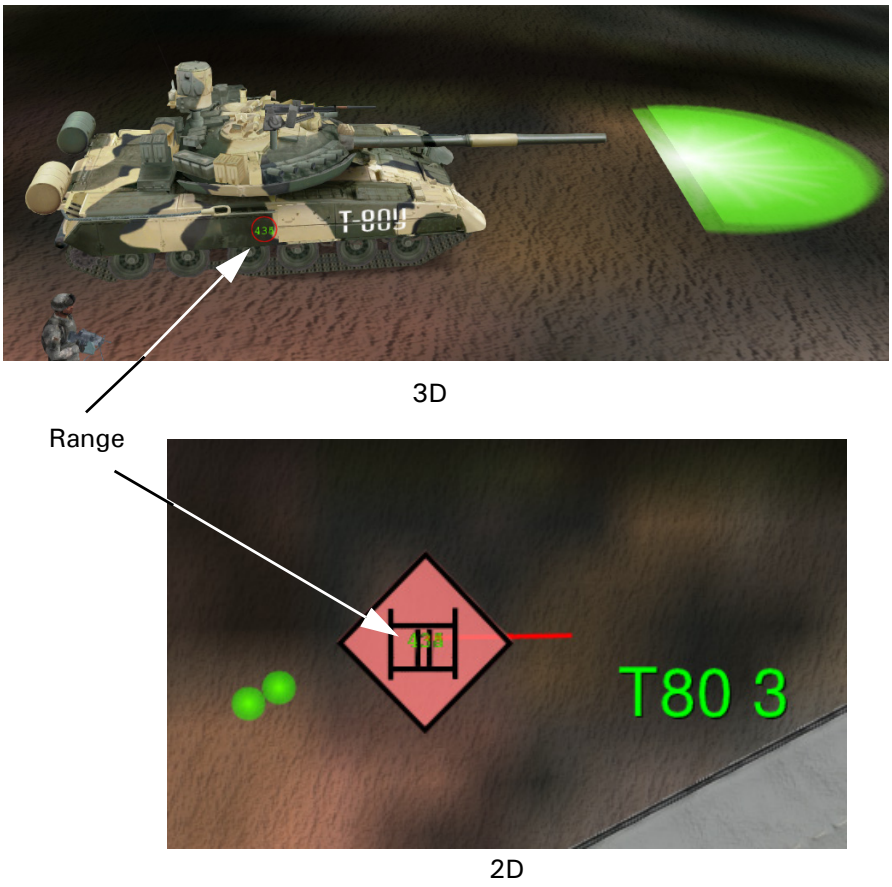


Figure 21-23. Laser designators

➤ To display laser designators, choose **Settings** → **Laser Designators**, or on the Display Settings Toolbar, click the Laser Designator button ( ).



22. Intervisibility Objects

Intervisibility objects show line-of-sight between entities or objects in the simulation.

Intervisibility (Line-of-Sight) 22-2
Types of Intervisibility 22-3
Point-to-Point Intervisibility 22-3
Intervisibility Fan 22-4
Entity Intervisibility 22-4
Intervisibility Objects can be Transient or Permanent 22-5
Creating a Transient Intervisibility Object 22-6
Creating a Permanent Intervisibility Object 22-7
Editing an Intervisibility Object 22-8
Configuring Intervisibility Lines and Fans 22-9
Pinning an Intervisibility Object to a Simulation Object 22-10
Showing and Hiding Intervisibility Lines 22-10
Displaying Intervisibility Line Information 22-11
Deleting Intervisibility Objects 22-12

22.1. Intervisibility (Line-of-Sight)

You can display intervisibility (line-of-sight or LOS) between points, within areas, and between a point and all simulation objects in a defined area. Intervisibility displays can be transient or permanent. Permanent intervisibility displays are objects. You can resize them and move them around the terrain. Intervisibility objects can be tied to a particular simulation object.

VR-Vantage IG does not support intervisibility.

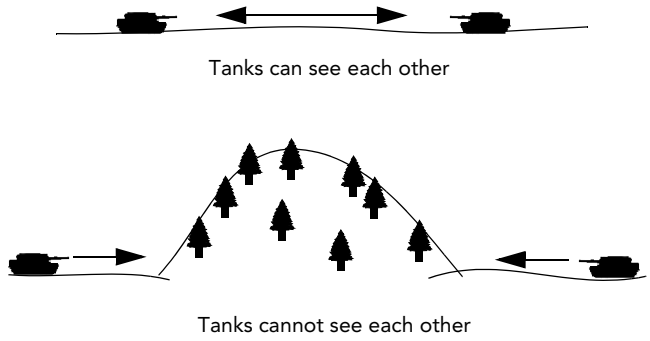


Figure 22-1. Intervisibility

You can configure parameters for intervisibility objects, such as the height above the terrain from which intervisibility is calculated. You can also change the colors used to display intervisibility.

- By default, intervisibility between simulation objects is calculated from the center of their bounding volumes. It is possible that this may not match the line-of-sight calculations used by the simulation application whose output you are viewing. If so, simulation objects may shoot at targets for which they do not appear to have line-of-sight.
- Linear and areal features do not block line-of-sight. Only point features block LOS.

22.2. Types of Intervisibility

VR-Vantage supports the following types of intervisibility:

- Point-to-point (intervisibility line).
- Intervisibility fan.
- Entity intervisibility.

The intervisibility fan and entity intervisibility are displayed as a circular area. However, intervisibility calculations are made within a cylindrical area extending above and below the terrain, so that intervisibility calculations can be made for air platform entities and subsurface entities.

In the 2D projection, intervisibility lines are fully visible on the terrain. In the 3D projection, they may pass through the terrain.

22.2.1. Point-to-Point Intervisibility

Point-to-point intervisibility (Figure 22-2) shows the line-of-sight between any two points. The portion of the line that is visible from the starting point is green. The portion that is not visible is red. You can change the colors used.

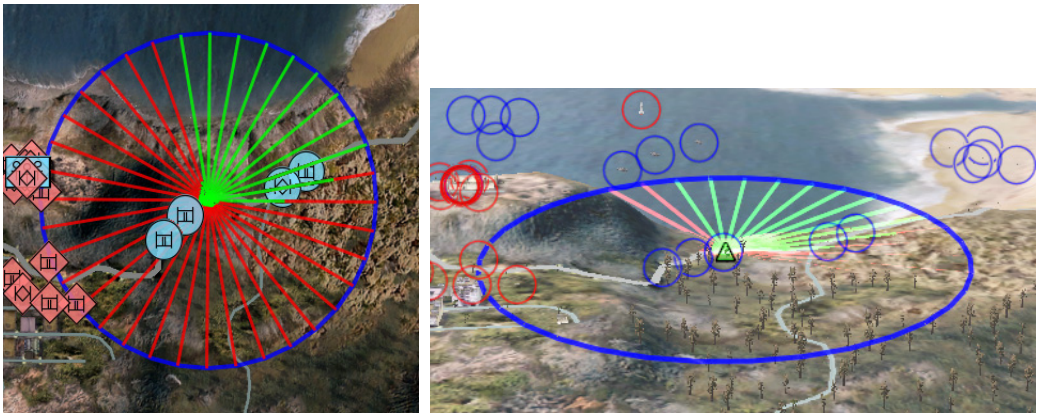


2D 3D

Figure 22-2. Point-to-point intervisibility

22.2.2. Intervisibility Fan

An intervisibility fan (Figure 22-3) shows a series of lines radiating from the center of a circle to its outer edge. Each ray shows intervisibility along its path from the origin to the border. You can configure the distance between rays. (For details, please see "Configuring Intervisibility Lines and Fans," on page 22-9.)



2D 3D

Figure 22-3. Intervisibility fan

22.2.3. Entity Intervisibility

Entity intervisibility (Figure 22-4) shows which simulation objects are visible from the center of a defined area and whether or not the simulation objects can see the center point. If you center the intervisibility circle on a point on the terrain, you can see which simulation objects can see that point. If you lock the circle to a simulation object, you can see which simulation objects can see that object and which it can see. (For details, please see "Configuring Intervisibility Lines and Fans," on page 22-9.)



2D 3D

Figure 22-4. Entity intervisibility

22.3. Intervisibility Objects can be Transient or Permanent

You can configure intervisibility objects to be transient or permanent. (For details, please see "Configuring Intervisibility Lines and Fans," on page 22-9.) A transient intervisibility object displays intervisibility from the origin point (defined by a mouse click) to wherever you move the cursor. If you click the mouse again, you reset the origin. You cannot use the mouse for other actions until you exit intervisibility mode (by right-clicking). This mode is convenient when you want to quickly check intervisibility, but do not want to keep intervisibility objects in the display.

A permanent intervisibility object stays in the display after you create it. You can resize the object, edit drawing options, and move it around the display.

22.3.1. Creating a Transient Intervisibility Object

Transient intervisibility objects do not stay in the display after you quit intervisibility mode.

To display transient intervisibility:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Intervisibility Settings page (Figure 22-5).

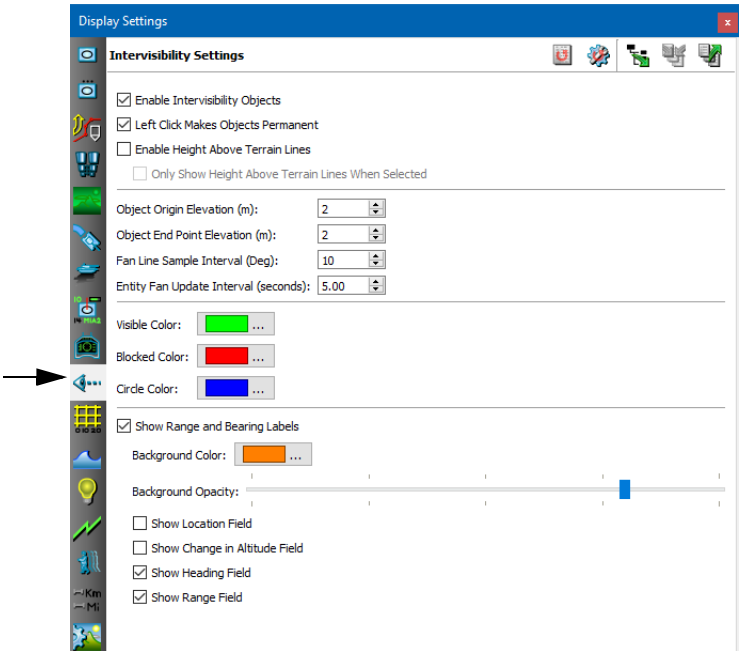


Figure 22-5. Intervisibility Settings page

3. Clear the Left Click Makes Objects Permanent check box.
4. Choose **Intervisibility** → **Create** *intervisibility_type*, where *intervisibility_type* is a type of intervisibility object, or click one of the buttons on the Intervisibility Toolbar.
5. Click on the terrain to place the origin of the intervisibility object.
6. Move the cursor around the terrain to expand and contract the object. For point-to-point intervisibility, moving the cursor also changes the direction of the line.
7. To change the origin of the intervisibility display, click another location on the terrain.
8. Right-click to cancel intervisibility mode.

22.3.2. Creating a Permanent Intervisibility Object

Permanent intervisibility objects remain in the display after you create them.

To create a permanent intervisibility object:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Intervisibility Settings page (Figure 22-5).
3. Select Enable Intervisibility Objects.
4. Select Left Click Makes Objects Permanent.
5. Choose **Intervisibility** → **Create** *intervisibility_type*, where *intervisibility_type* is a type of intervisibility object, or click one of the buttons on the Intervisibility Toolbar.
6. Click on a simulation object or on a location on the terrain where you want the origin of the intervisibility object to be located. An intervisibility object is displayed. As you move the cursor closer to the origin, or farther away from it, the circle or line contracts and expands. If you are displaying point-to-point, you can rotate the line around the origin before you specify the end point.
7. Click on the terrain where you want to fix the end point of a line or border of a circle.

22.4. Editing an Intervisibility Object

You can change the following aspects of an intervisibility object:

- The location of either end point of an intervisibility line, to change its length and direction.
- The origin of an intervisibility fan, thereby moving the entire fan.
- The diameter of an intervisibility fan.

If an intervisibility object is anchored to a simulation object, you cannot change the origin.

To change the length or diameter of an intervisibility object:

1. To edit a line, double-click its end point.

   To edit a fan, select the object, then double-click the outer ring of the fan.

   The cursor changes (✣).
2. Drag the end point or circle towards or away from the origin. If you are editing an intervisibility line, you can move the end point to any location.
3. Click to exit edit mode.

To revert to the original size of the intervisibility object, right-click while you are still in edit mode.

To move an intervisibility fan or line:

1. Double-click a fan's center point or any place on a line. The cursor changes.
2. Drag the object to a new location.
3. Click to set the location.

To revert to the original location of the intervisibility object, right-click while you are still in edit mode.

22.5. Configuring Intervisibility Lines and Fans

You can configure the following attributes of intervisibility objects:

- The elevation of the origin and end point.
- The sampling rate for intervisibility fans.
- The update rate for entity intervisibility fans.
- Display of height above terrain lines (3D) and height labels (3D and 2D).
- The colors used to display intervisibility objects.
- Display of range and bearing labels. For more information about range and bearing labels, please see "Displaying Intervisibility Line Information," on page 22-11.

When you change colors or the sampling rate, changes affect all existing intervisibility objects. Changes to the origin and end point elevation only affect newly created objects.

To configure intervisibility objects:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Intervisibility Settings page (Figure 22-5).
3. Change the settings as desired. Table 22-1 describes the settings.

Table 22-1: Intervisibility settings

Option	Description
Enable Height Above Terrain Lines	Enables or disables display of height above terrain lines (3D) and height labels (3D and 2D).
Show Height Above Terrain Lines Only for Selected	If selected, HAT lines are displayed only for the selected intervisibility objects.
Object Origin Elevation	The height, in meters, above the terrain at the origin point from which to calculate intervisibility.
Object End Point Elevation	The height, in meters, above the terrain at the end point to use to calculate intervisibility.
Fan Line Sample Interval	The distance, in degrees, between each ray in a intervisibility fan. After changing this value, press **`Enter`** or click away from the field.
Entity Fan Update Interval	How frequently, in seconds, to update the entity intervisibility fan.
Visible Color Blocked Color	The colors to use for the portion of an intervisibility line that a simulation object can see and the portion where visibility is blocked.
Circle Color	The color to use for the outer circle of an intervisibility fan.
Show Range and Bearing Labels	Enable or disable display of range and bearing labels.

Table 22-1: Intervisibility settings

Option	Description
Background Color	Specify the background color for range and bearing labels.
Background Opacity	Specify the opacity of the background for range and bearing labels.
Show Location Field	If range and bearing labels are displayed, show the location of the endpoints.
Show Change in Altitude Field	If range and bearing labels are displayed, show the difference in altitude between the endpoints.
Show Heading Field	If range and bearing labels are displayed, show the heading of each endpoint relative to the other point.
Show Range Field	If range and bearing labels are displayed, show the length of the intervisibility line.

22.6. Pinning an Intervisibility Object to a Simulation Object

When you pin an intervisibility object to a simulation object, the intervisibility object moves with the simulation object and you know that its origin is centered on the simulation object.

To pin an intervisibility object to a simulation object:

1. Select the simulation object to which you want to pin the intervisibility object.
2. Choose **Intervisibility** → **Create** *intervisibility_type*, where *intervisibility_type* is a type of intervisibility object, or click one of the buttons on the Intervisibility Toolbar. The origin is placed on the simulation object icon or model.
3. Specify the end point for the intervisibility object.

22.7. Showing and Hiding Intervisibility Lines

You can toggle the display of intervisibility lines on and off.

To enable or disable the display of intervisibility lines:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Intervisibility Settings page (Figure 22-5).
3. Select or clear the Enable Intervisibility Objects check box.



You can also enable and disable display of intervisibility objects by choosing **Intervisibility** → **Enable/Disable Intervisibility**, or by clicking the Enable/Disable Intervisibility button () on the Intervisibility Toolbar.

22.8. Displaying Intervisibility Line Information

You can display the following information about an intervisibility line:

- Location of each endpoint.
- Heading of the line from each endpoint.
- Altitude of each endpoint.
- Length of the line (range).



Figure 22-6. Intervisibility line information (2D)

The information label has a background. You can adjust the color and opacity of the background.

To display information about an intervisibility line:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Intervisibility Settings page (Figure 22-5).
3. Select or clear the Show Range and Bearing Labels check box.
4. Select the check boxes for the type of information you want to display. For descriptions of the check box options, please see Table 22-1.
5. To change the background color for the label:
   a. Click the Background Color button. A color picker opens.
   b. Select the color you want to use for the background.
   c. Click OK.
6. To change the opacity of the background, adjust the Background Opacity slider.

22.9. Deleting Intervisibility Objects

➤ To delete an intervisibility object, select the object and choose **Intervisibility** → **Delete**.



23. Sound Effects

This chapter explains how to enable and configure sound effects.

Introduction to Sound Effects 23-2
Enabling Sound Effects 23-2
Setting the Sound Volume 23-3
Specifying the Range in which Sounds are Heard 23-3
Associating Sound Files with Entities and Events 23-4
Mapping a Sound File to an Entity 23-5
Mapping Sound Files to Fire and Detonation Events 23-8

23.1. Introduction to Sound Effects

VR-Vantage can associate sound effects with entities, fire events, and detonation events. You can enable and disable use of sound effects and you can specify that sounds will be heard only when the observer is within a certain distance of the entity or event that generates them.

Sound effects are not supported on Linux.

23.2. Enabling Sound Effects

To enable sound effects:

1. Choose **Settings** → **Audio Settings**. The Audio Settings dialog box opens.
2. Select the Audio Settings page (Figure 23-1).

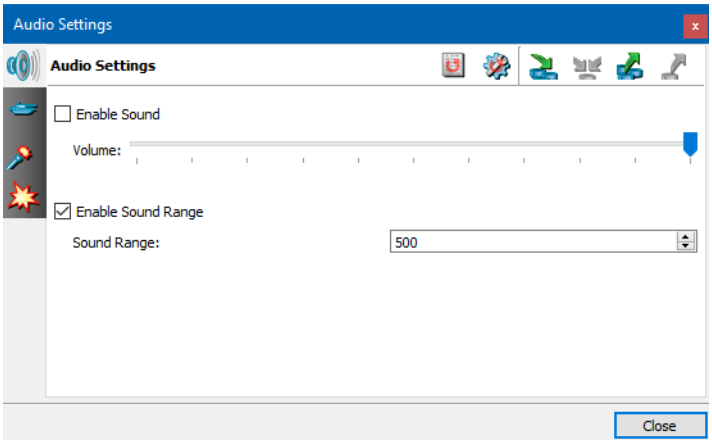


Figure 23-1. Audio Settings page

3. Select the Enable Sound check box.

You can also enable and disable sound effects by clicking the Audio button ( ) on the Display Settings toolbar.

23.2.1. Setting the Sound Volume

You can adjust the volume of sound effects.

To set the sound volume:

1. Choose **Settings** → **Audio Settings**. The Audio Settings dialog box opens.
2. Select the Audio Settings page (Figure 23-1).
3. Adjust the Volume slider.

> You can also change the volume by clicking the arrow on the Audio button ( ) (on the Display Settings toolbar) and adjusting the Audio Volume slider.

23.2.2. Specifying the Range in which Sounds are Heard

You can specify that sound effects be played only if the observer is within a certain distance of the entities that can emit sounds. If sound range is enabled, as soon as the observer moves beyond the range from an entity, sound effects are shut off. If this option is disabled, sound effects are played for all entities in the scenario regardless of their distance from the observer. However, even if sound range is not enabled, as the distance between the observer and entities increases, sound levels drop off.

To specify the range in which sounds can be heard:

1. Choose **Settings** → **Audio Settings**. The Audio Settings dialog box opens.
2. Select the Audio Settings page (Figure 23-1).
3. Select the Enable Sound Range check box.
4. Specify a value in the Sound Range text box.

23.3. Associating Sound Files with Entities and Events

VR-Vantage includes a set of sound files (WAV format) for generic entity platforms and for fire and detonation events. The files are in *./data/Audio/Sounds*. Sound files are associated with entity type enumerations. You can change the mappings and add new mappings.

You add and edit mappings between entity types, fire events, or detonation events and sound files on pages in the Audio Settings dialog box. You can add or edit mappings using mapping dialog boxes or directly in the mappings windows. The mapping dialog boxes behave differently depending on the state of the mapping window when you open the dialog box, as follows:

- If you click the Add Mapping button and have not selected a mapping in the Entity Mapping Settings window, you are presented with a default entity type (all wildcards).
- If you click the Add Mapping button and have selected an existing mapping, the dialog box displays the mapping you selected. From here, you can edit the existing mapping or create a new one.

When you edit a mapping, the result depends on whether you edit the entity or event enumeration or the sound file. The differences are based on the requirements for uniqueness in a mapping. The uniqueness of an entity type or fire event mapping is based on the entity type enumeration. The uniqueness of a detonation event mapping is based on the entity type enumeration and the detonation result value. Table 23-1 lists the results of different editing actions.

Table 23-1: Editing sound mapping files

Action	Result
Edit enumeration in dialog box.	New mapping is added. The previous mapping is not changed.
Edit enumeration in place.	New mapping replaces previous mapping.
Edit sound file in dialog box or in window.	Mapping to current entity or event is updated.
Edit channel in dialog box or in window.	Mapping to current entity or event is updated.
Edit detonation result in dialog box.	New mapping is added. The previous mapping is not changed.
Edit detonation result in window.	New mapping replaces previous mapping.

23.3.1. Mapping a Sound File to an Entity

An entity type enumeration can only have one sound file mapped to it. A sound file can be mapped to as many entity types as you want.

Adding a New Entity Sound Mapping

To add a new entity sound mapping:

1. Choose **Settings** → **Audio Settings**. The Audio Settings dialog box opens.
2. Select the Entity Sound Editor page (Figure 23-2).

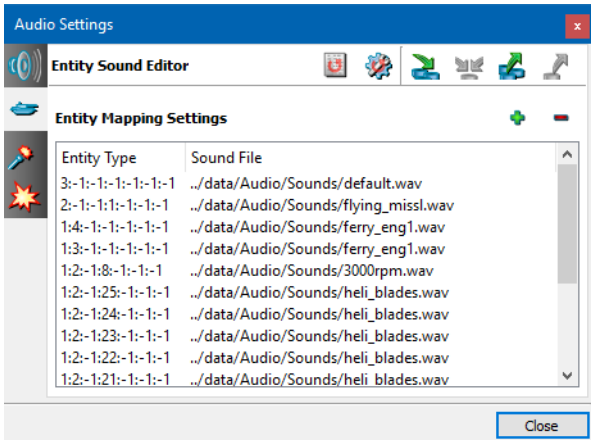


Figure 23-2. Entity Sound Editor page

3. Click the Add button (). The Create New Entity Type Sound Mapping dialog box opens (Figure 23-3).

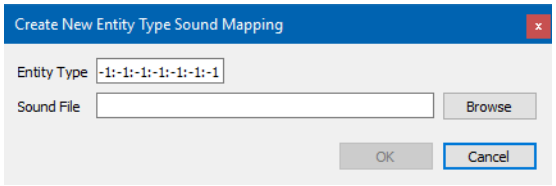


Figure 23-3. Create New Entity Type Sound Mapping dialog box



You cannot open the Create New Entity Type Sound Mapping dialog box with default options if one of the existing mappings is selected. If you have selected a mapping and want to return to the unselected state, click on white space in the mappings window or select one of the other tabs and then select the Entity Sound Editor tab again.

4. In the Entity Type box, type the enumeration for this mapping.

5. In the Sound File text box, type the pathname of a sound file or click Browse and select a file in the file chooser dialog box.
6. Click OK.



The OK button does not become enabled until you specify a valid entity type and sound file name.

Creating an Entity Sound Mapping from an Existing Mapping

When you edit a mapping, you can either change the sound file associated with the current entity type or create a new mapping with a different entity type.

To create a sound mapping from an existing mapping:

1. Choose **Settings** → **Audio Settings**. The Audio Settings dialog box opens.
2. Select the Entity Sound Editor page (Figure 23-2).
3. Select the mapping that you want to use as the starting point.
4. Click the Add button (). The Create Entity Type Sound Mapping from Existing Mapping dialog box opens (Figure 23-4).

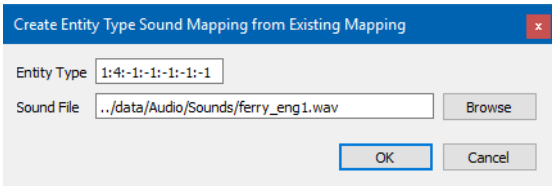


Figure 23-4. Create Entity Type Sound Mapping from Existing Mapping dialog box

5. To change the sound file associated with the entity type, type a new pathname or click Browse and select a sound file.
6. To create a new mapping, change the Entity Type enumeration.
7. Click OK. If you just changed the sound file, the mapping is updated. If you changed the entity type, a new mapping is added to the list.

Editing the Entity Type in the Mappings Window

If you edit the entity type in the mappings window (that is, without using a dialog box), it replaces the previous mapping.

To edit the entity type in the window:

1. Choose **Settings** → **Audio Settings**. The Audio Settings dialog box opens.
2. Select the Entity Sound Editor page (Figure 23-2).
3. Double-click the entity type you want to edit. It becomes editable (Figure 23-5).

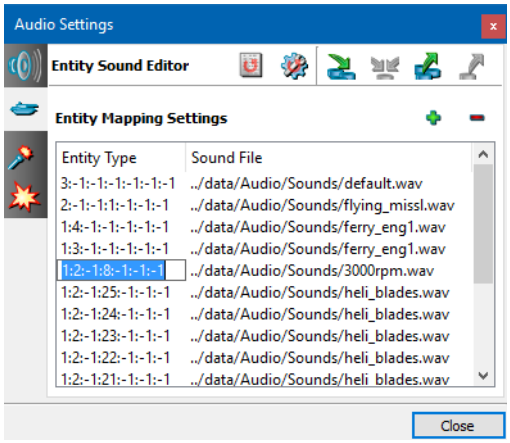


Figure 23-5. Editable entity type

4. Change the entity type enumeration.
5. Press **`Enter`**.

Editing the Sound Mapping in the Mappings Window

When you edit the sound mapping in the mappings window, the change does not affect any other mapping.

To edit the sound mapping in the window:

1. Choose **Settings** → **Audio Settings**. The Audio Settings dialog box opens.
2. Select the Entity Sound Editor page (Figure 23-2).
3. Double-click the sound mapping that you want to change. The Sound File dialog box opens.
4. Select the new sound file.
5. Click Open.

23.3.2. Mapping Sound Files to Fire and Detonation Events

When you map a sound file to a weapon fire or a detonation event, you can specify a channel for the sound. When you map a detonation sound, you can specify the detonation result for the mapping.

Specifying a channel for a sound event lets you control the number of sounds that can play concurrently. Sounds that are mapped to channel 0 can all play at the same time. Sounds mapped to any other channel can play only if no other event is using the channel. For example, suppose you map two weapon fire events to channel 1. One of the weapon type fires and the sound is generated. Then the second type of weapon event occurs. Since the first event is using channel 1, the second sound is not generated.

The SISO Enumerations document specifies a list of detonation results, such as Entity Impact, Ground Impact, and Air Hit. The detonation result is part of the detonation message sent over the network. You can specify that a detonation sound be played only for a certain type of detonation result. If you set the result to "any" it plays for any type of detonation. If, for a particular detonation type, you map one sound to "any" and another sound to a specific detonation result, the specific detonation result takes precedence.



You can change fire and detonation mapping settings directly, rather than using the sound mapping dialog box, just like you can change entity type sound mappings. For details, see the sub-procedures under "Mapping a Sound File to an Entity," on page 23-5.

To map a sound file to a weapon fire or detonation:

1. Choose **Settings** → **Audio Settings**. The Audio Settings dialog box opens.
2. Select the Fire Sound Editor page or Detonation Sound Editor page (Figure 23-6), as appropriate.

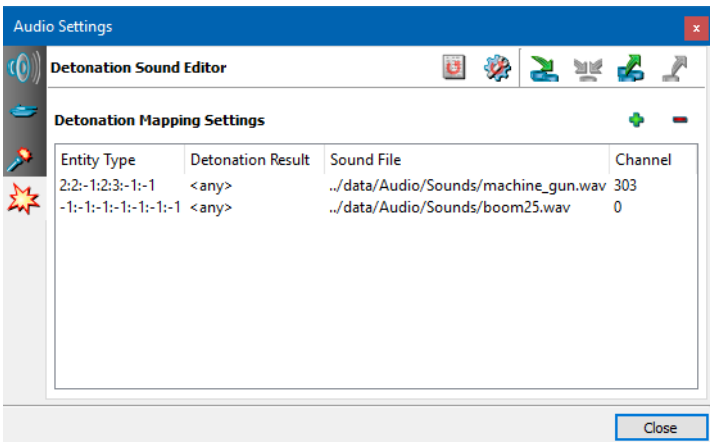


Figure 23-6. Detonation Sound Editor page

3. Click the Add button (+). The Create New Detonation (or Weapon Fire) Sound Mapping dialog box opens (Figure 23-7).

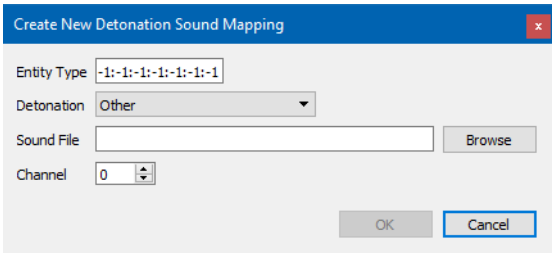


Figure 23-7. Create New Detonation Sound Mapping dialog box

When you click the Add button, if you have selected a mapping in the list, the Create New Sound Mapping from Existing dialog box displays the values for the selected mapping. If you change the entity type or detonation result, a new mapping is added with the new values.

4. In the Entity Type box, type the enumeration for this mapping.
5. Optionally, for detonation sounds, select an option on the Detonation list.
6. In the Sound File text box, type a pathname for the sound file you want to use or click Browse and select a file in the file choose dialog box.
7. Optionally, specify a channel in the Channel box.
8. Click OK.



VII. Terrain

A terrain is a representation of the earth or a portion of it on which simulated entities move about. VR-Vantage allows you to build your terrain at runtime using a variety of database and vector formats. It can load OpenFlight, DTED, CTDB, CDB, MetaFlight, and GDB (MAK terrain format) databases. It can load feature data from shapefiles and GDB files. You can also superimpose geospatial images such as GeoTiffs or other raster image files on the terrain. You can save these composed terrains in the VR-Vantage MTF file format.

VII.1. Terrain Databases Provided with VR-Vantage

Table 23-1 describes the terrain databases provided with VR-Vantage. Some have variants, which indicate the coordinate system or other variations. Terrains with "(online)" in the title connect to VR-TheWorld Server or another online server and require an internet connection. Some terrains have variations that emphasize a particular set of features or other data. VR-Vantage also has a supplemental data package that includes additional terrains. Please contact your MAK salesperson for download details.

Table 23-1: Terrain databases shipped with VR-Vantage (Sheet 1 of 2)

Variation	Description
Ala Moana.mtf Ala Moana - Experimental.mtf	A high-resolution area of Honolulu, Hawaii, USA, set into a raster map of the world. This terrain is stored locally and does not require an internet connection. Ala Moana - Experimental.mtf has additional features that may reduce performance.
Bakersfield Topo Map (online).mtf	VR-TheWorldOnline with topographical maps for the area around Bakersfield, California, USA. This terrain contains some feature data that simulation objects in aggregate-level scenarios can take advantage of.
Brooklyn.mtf Brooklyn (online).mtf	A high resolution city center surrounded by an extruded cityscape. Located geographically in Brooklyn, N.Y., USA, but not an actual location in that city.
California (online).mtf California Air (online).mtf	A procedurally generated earth merged with high resolution imagery and elevation. California Air is optimized for viewing from altitude. It should perform a little better than California.
DestructibleTerrain.mtf	A small terrain for demonstrating destructible terrain based on switch nodes.
Ground-db.mtf	UTM. OpenFlight. A generic relatively flat ground terrain.
LittlePond.mtf LittlePond (online).mtf	Flat earth. DTED. Shape data. Imagery. A detailed suburban inset in a larger, less detailed terrain covering Boston, Massachusetts, U.S.A. and points west.
MAK Earth - Base (online).mtf	The base VR-TheWorld Server terrain with a high resolution inset for Honolulu, Hawaii, USA and the surrounding area. Does not include streaming feature data. Includes streaming feature data.

Table 23-1: Terrain databases shipped with VR-Vantage (Sheet 2 of 2)

Variation	Description
MAK Earth (online).mtf MAK Earth - Experimental (online).mtf	VR-TheWorld Server terrain with a high resolution inset for Honolulu, Hawaii, USA and other cities. Includes streaming feature data. MAK Earth - Experimental has additional features that may reduce performance.
Makland.mtf	UTM. OpenFlight. A fictional terrain created by MAK. It includes a variety of terrain and feature data, including underwater features. It is placed in the Indian Ocean.
ProceduralEarth (online).mtf ProceduralEarth - OpenStreetMap (online).mtf	Procedurally generated terrains. Images are generated based on land use data, resulting in high quality visual imagery without the overhead of high resolution static images. Terrains may include airport data for all airports in the world, street data from OpenStreetMap.
RhoneValley (online).mtf	A procedurally generated view of the Rhone Valley in France.
Sadr City (online).mtf Sadr City.mtf	An urban geotypical Middle Eastern terrain in Sadr City, Iraq, a suburb of Baghdad. Ideal for urban scenarios.
SanLuisObispo.mtf	Flat earth. DTED. Imagery. A terrain that covers a portion of California, U.S.A.
SanLuisObispoTopoGdal.mtf	Geocentric. San Luis Obispo in topographic coordinates as local streaming terrain using an earth file.
Test terrains	Multiple terrains used for testing different terrain database capabilities.
VR-Village.mtf	GDB. Feature data. OpenFlight terrain. VR-Village is a fictional database created by VT MAK. It has a highly detailed small town in a larger, sparsely detailed desert environment. The town includes multi-level buildings, underground tunnels, and a road that tunnels through a mountain. The coordinates place it off the west coast of Africa.
WorldFlatEarthNoElevation.mtf	Flat earth. Low resolution projected image of the world.
WorldGeocentric.mtf	Geocentric. Low resolution projection of the world.
WorldGlobalWater.mtf	A terrain with no elevation data for land masses.



24. Changing How the Terrain is Displayed

This chapter explains how to find out information about a terrain by changing how it is displayed.

Displaying Feature Data (2D) 24-2
Displaying a List of Vector Features 24-4
Changing the Colors Used for Feature Data 24-5
Displaying Grid Lines 24-6
Configuring Grid Lines 24-7
Configuring Visible Surfaces (GDB Soil Types) 24-8
Coloring Terrain Based on Elevation 24-9
Configuring Elevation Color Settings 24-10
Displaying Contour Lines 24-11
Configuring Contour Lines 24-12
Displaying Models and Terrain in Wireframe Mode 24-13

24.1. Displaying Feature Data (2D)

If a terrain contains feature data, you can display it in the Plan View observer mode. Figure 24-1 shows a section of the Little Pond terrain with features disabled and enabled.



Figure 24-1. Feature data

This setting only displays features imported from GDB files or shapefiles. It does not display features streamed from a terrain server. If you load features using an earth file, they are drawn (or not drawn) based on the configuration information in the file.



If your terrain has many features, displaying them can take a long time.

To display feature data:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page.
3. In the Observer Mode list, select Plan View (Figure 24-2).

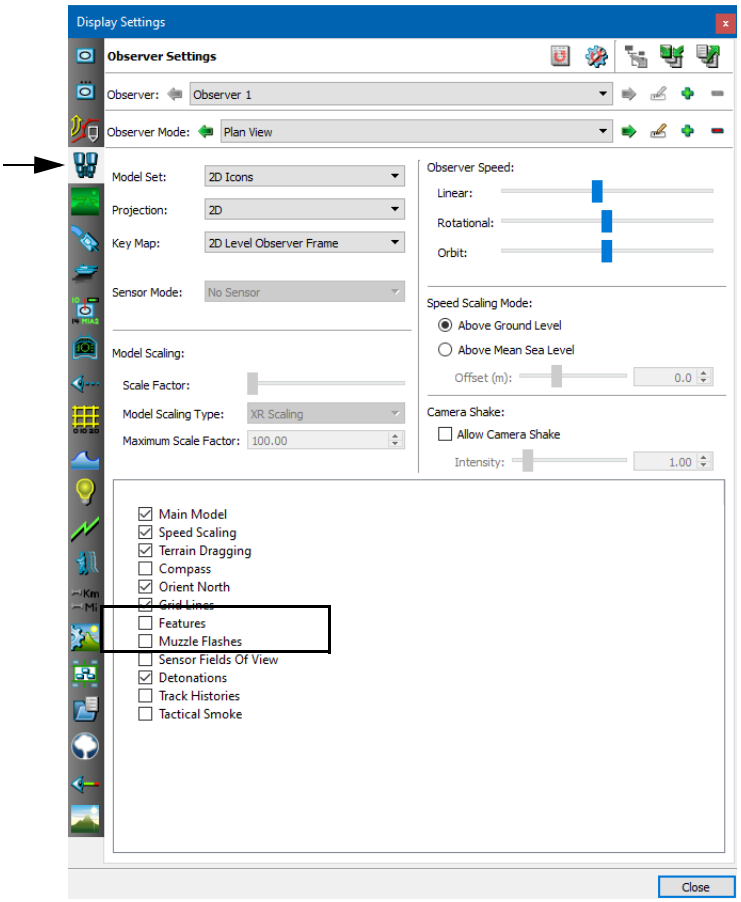


Figure 24-2. Observer Settings page, Plan View observer mode

4. Select or clear the Features check box.



You can also enable or disable display of feature data by choosing **Observer** → **Features Ocean**, or clicking the Feature Data button () on the Observer Settings Toolbar.

24.1.1. Displaying a List of Vector Features

If a terrain has vector features, they are listed on the Vector Network tab of the Terrain Information Panel. If you select a feature in the list, it is highlighted in the display, and vice versa (Figure 24-3).

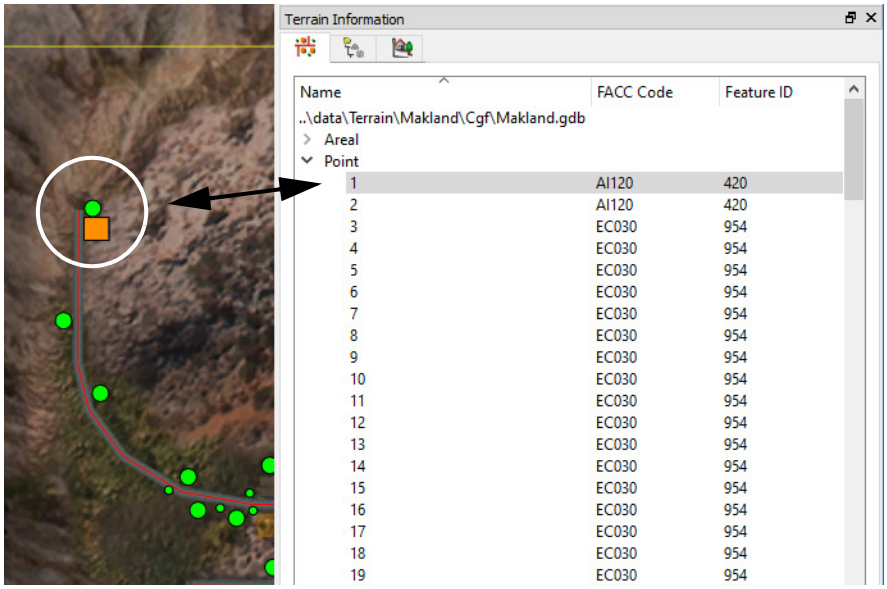


Figure 24-3. Vector feature selected

To display the feature list:

1. Choose **View** → **Terrain Information Panel**. The Terrain Information Panel opens.
2. Select the Vector Network tab (Figure 24-3).
3. Expand the top-level categories to see the lists of individual features.

24.1.2. Changing the Colors Used for Feature Data

If a terrain database contains feature data, you can change the colors used to display it. In Figure 24-4, the Woodland feature in the Little Pond terrain was changed to pink. Compare it to the default color (Figure 24-1).



Figure 24-4. Feature data with color change

To change the color of feature data:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Feature Settings page (Figure 24-5).

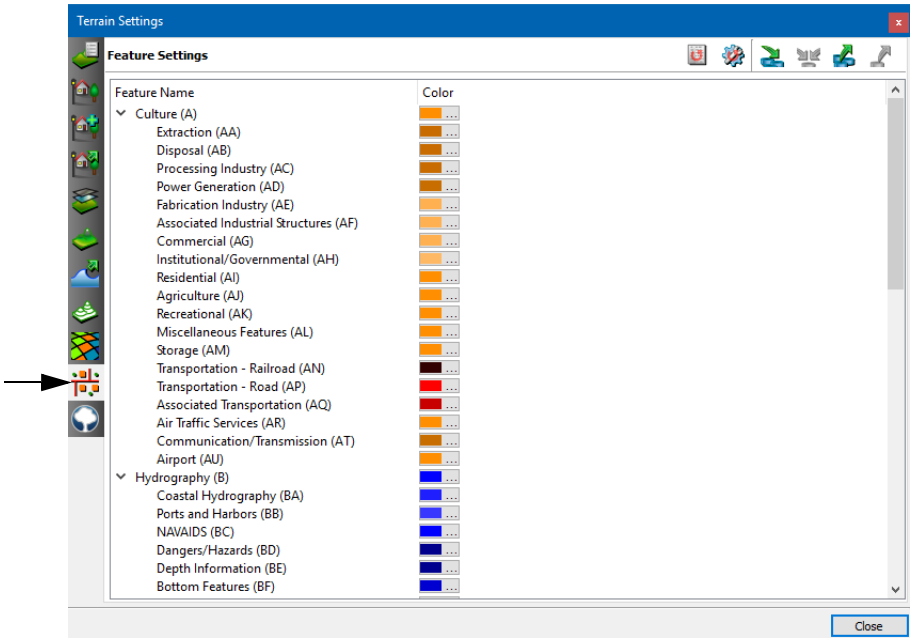


Figure 24-5. Feature Settings page

3. Click the color swatch for the feature that you want to change. A color picker dialog box opens.
4. Select the color that you want to use.
5. Click OK.

24.2. Displaying Grid Lines

VR-Vantage can display grid lines in the 2D view. Grid-lines are an observer-specific setting.

To enable or disable grid lines:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page (Figure 24-6).

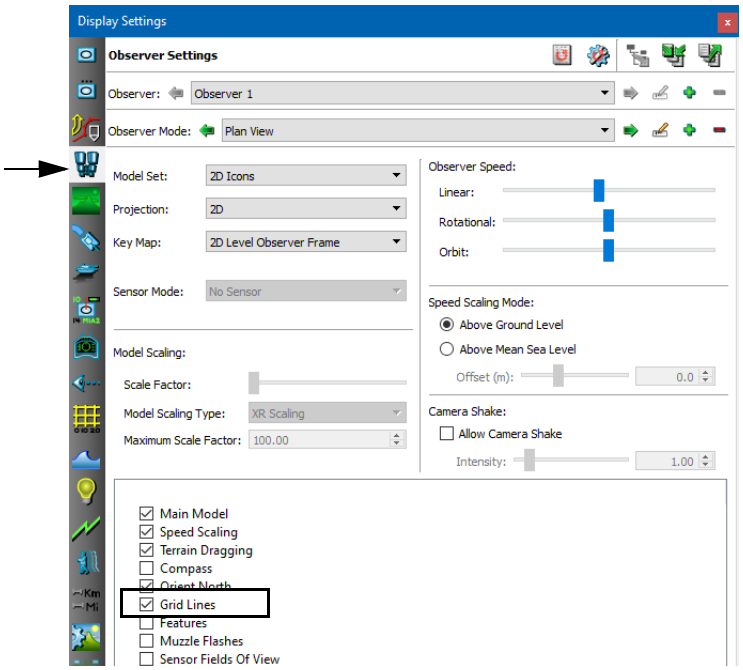


Figure 24-6. Observer Settings page

3. Select Plan View mode (or a locally created 2D observer mode).
4. Select or clear the Grid Lines check box.

> You can also enable or disable grid lines by choosing **Observer** → **Grid Lines** or clicking the Grid Lines button ( ) on the Observer Settings Toolbar.

24.2.1. Configuring Grid Lines

You can configure the coordinate system used for grid lines, the precision, and the spacing. Spacing can be fixed, in meters, or determined by VR-Vantage using a specified number of screen pixels. Fixed spacing can result in a very tight grid and unreadable labels as you zoom out. Auto spacing does not have this problem. However, the grid lines will constantly recalculate as you zoom in and out.

To configure grid lines:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Grid Lines Settings page (Figure 24-7).

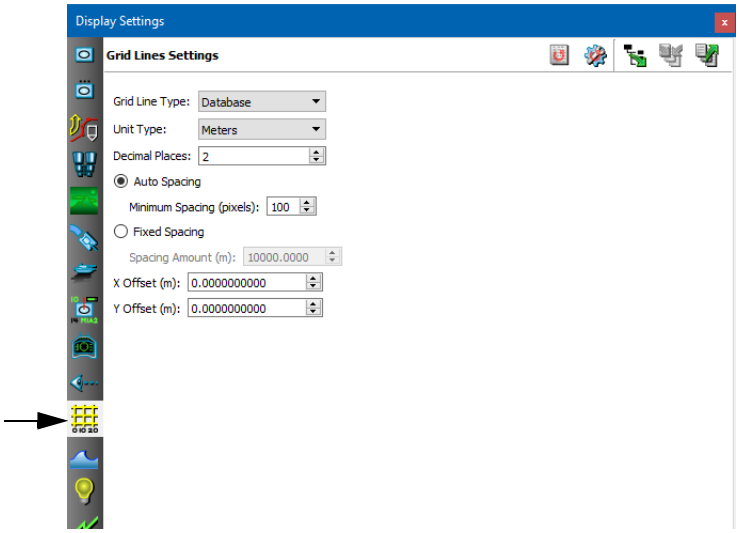


Figure 24-7. Grid Lines Settings page

3. To change the coordinate system used for grid lines, select an option in the Grid Line Type list. The list displays grid line types that are appropriate for the type of database loaded. (If no database is loaded, the coordinate system is set to Flat Earth.)
4. To specify the unit type for the grid line labels, select an option on the Unit Type list.
5. To specify the precision of the grid line labels, type a value in the Decimal Places box.
6. To change the spacing type, select AutoSpacing or Fixed Spacing and specify a value for the chosen spacing type.
7. To specify an offset from the origin, change the values for X Offset and Y Offset.

24.3. Configuring Visible Surfaces (GDB Soil Types)

If a terrain patch was loaded from a GDB file, you can display or hide GDB soil types. Figure 24-8 shows the *makland.gdb* terrain file. The view on the left shows all soil types. The view on the right has Building, Grass, Ocean, and Rock surfaces hidden.

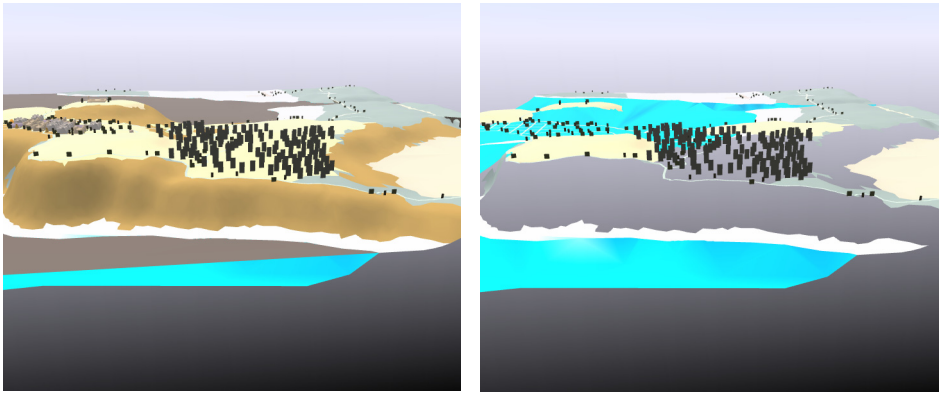


Figure 24-8. Visible surfaces toggled off

To display or hide GDB soil types:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Visible Surfaces page (Figure 24-9).

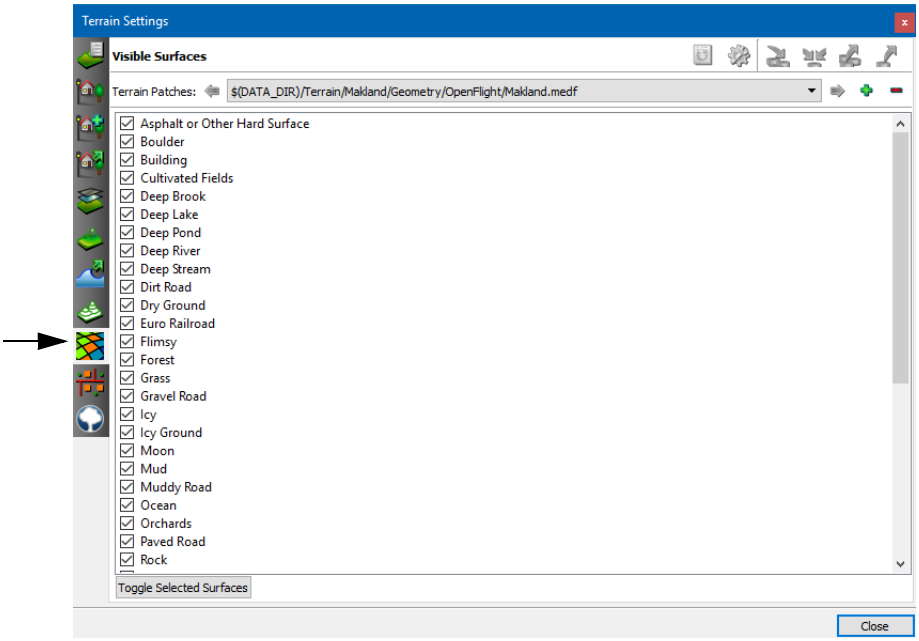


Figure 24-9. Visible Surfaces page

3. In the Terrain Patches list, select the terrain patch that you want to configure.
4. Select the check box for each soil type that you want to be visible.
5. Clear the check box for each soil type that you want to hide.
6. To change the state of multiple soil types at one time:
   a. Select (highlight) the names (not the check boxes) of the soil types whose state you want to change.
   b. Click Toggle Selected Surfaces.

24.4. Coloring Terrain Based on Elevation

You can color the terrain based on elevation.

To enable or disable coloring of terrain by elevation:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Elevation Color page (Figure 24-10).

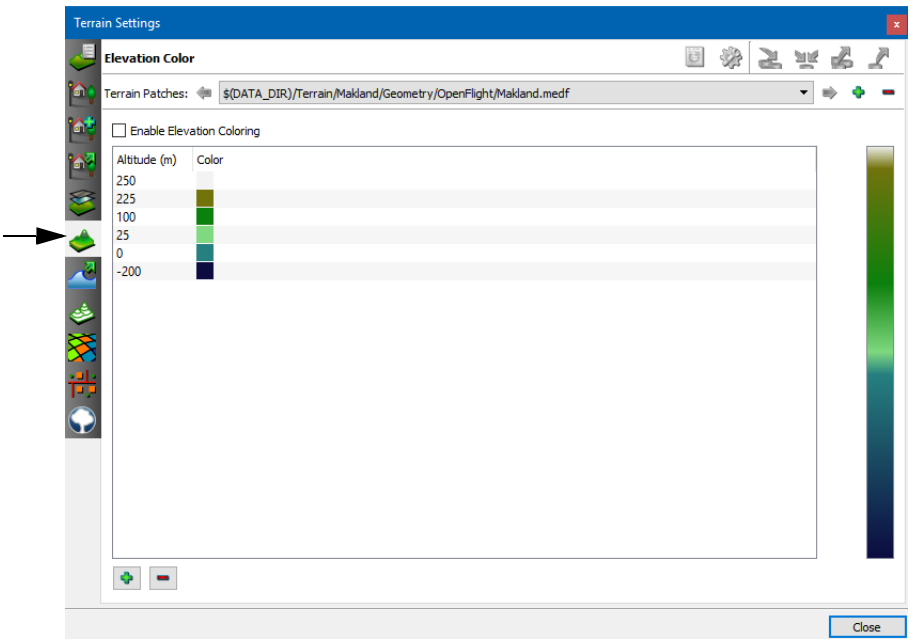


Figure 24-10. Elevation Color page

3. In the Terrain Patches list, select the terrain patch for which you want to enable elevation coloring.
4. Select or clear the Enable Elevation Coloring check box.

24.4.1. Configuring Elevation Color Settings

You can specify the color applied to terrain at a given altitude. You can add altitude points and remove them.

Inserting an Altitude Point

To insert an altitude point:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Elevation Color page (Figure 24-10).
3. In the Terrain Patches list, select the terrain patch for which you want to insert an altitude point.
4. Select an altitude point in the list that is above or below the altitude point you want to add.
5. Click Insert Point. A new point is added with the same altitude as the one you selected.
6. Double-click the new altitude point to make it editable.
7. Set the altitude point to the altitude that you want.
8. Click away from the altitude point to set it.

Deleting an Altitude Point

To delete an altitude point:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Elevation Color page (Figure 24-10).
3. In the Terrain Patches list, select the terrain patch for which you want to delete an altitude point.
4. Select the altitude point that you want to delete.
5. Click Delete Point.

Changing the Color of an Altitude Point

To change the color of an altitude point:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Elevation Color page (Figure 24-10).
3. In the Terrain Patches list, select the terrain patch for which you want to change the color of an altitude point.
4. Double-click the color swatch for the altitude point whose color you want to change. A color chooser dialog box opens.
5. Select or specify a new color.
6. Click OK.

24.5. Displaying Contour Lines

You can display contour lines and intermediate contour lines on the terrain. Intermediate contour lines divide the contour line interval into fifths (Figure 24-11). You can configure the thickness of contour lines and the interval between them.

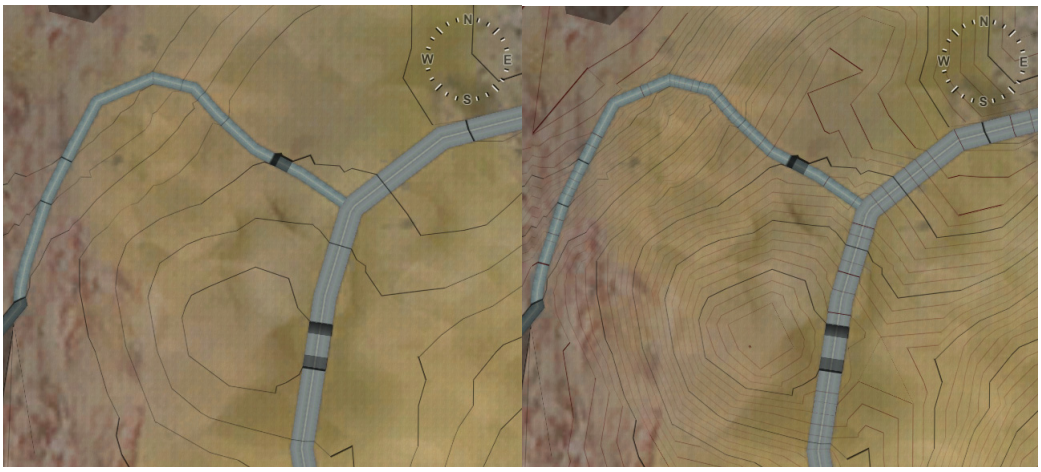


Figure 24-11. Contour lines and intermediate contour lines

To display contour lines:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Contour Lines page (Figure 24-12).

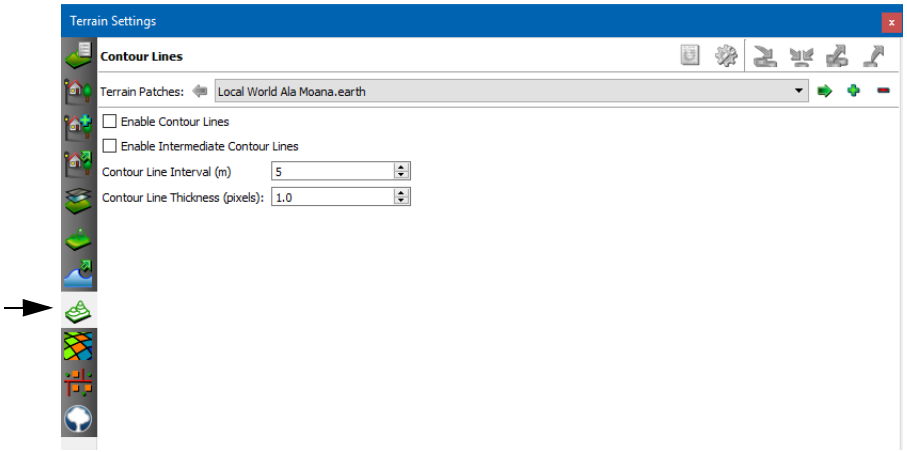


Figure 24-12. Contour Lines page

3. In the Terrain Patches list, select the terrain patch for which you want to enable or disable the display of contour lines.
4. Select Enable Contour Lines.
5. To enable intermediate contour lines, select Enable Intermediate Contour Lines.

24.5.1. Configuring Contour Lines

You can configure the thickness of contour lines and the interval between them.

To configure contour lines:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Contour Lines page (Figure 24-12).
3. In the Terrain Patches list, select the terrain patch for which you want to configure the thickness and interval of contour lines.
4. In the Contour Line Interval text box, specify the distance, in meters, between contour lines.
5. In the Contour Line Thickness text box, specify the thickness, in pixels, for contour lines.

24.6. Displaying Models and Terrain in Wireframe Mode

In wireframe mode, VR-Vantage displays the polygons that make up the terrain and objects (Figure 24-13).



Figure 24-13. Wireframe off and on

To enable or disable wireframe mode:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 24-14).

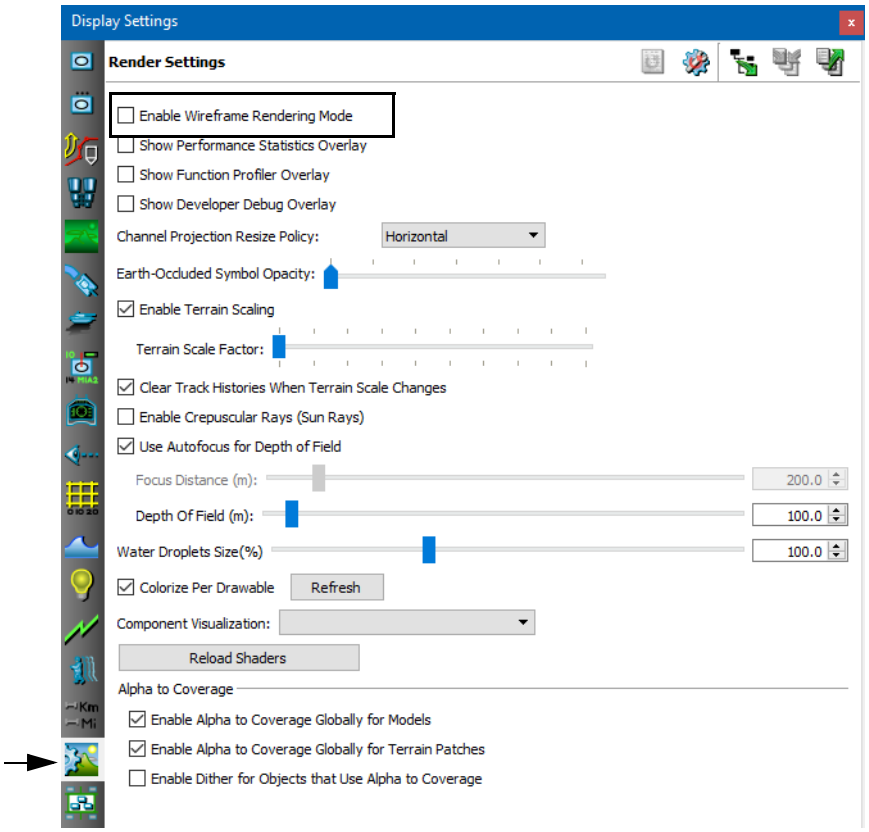


Figure 24-14. Render Settings page

3. Select the Enable Wireframe Rendering Mode check box.



You can also enable and disable wireframe mode by choosing **Settings** → **Wireframe Rendering** or clicking the Wireframe Rendering button ( ) on the Display Settings Toolbar.



25. Composing Terrains

This chapter explains how to combine various types of data to build a terrain.

Terrain Agility and Composability 25-2
Supported File Formats 25-2
Feature Layers and Props 25-4
Creating a Composed Terrain 25-5
Considerations and Limitations for Building Terrains 25-5
Saving a Terrain 25-6
Adding Elevation Data (Terrain Patches) to a Terrain 25-7
Adding Images to a Terrain 25-9
Changing the Display Order of Raster Maps 25-13
Adding a Feature Layer 25-14
Adding a Dynamic Ocean Layer 25-15
Extracting Water Textures and Adding an Ocean Layer 25-16
Adding a Dynamic Ocean Layer Directly 25-17
Setting the Ocean LOD Elevation 25-18
Configuring the Ocean Height Map 25-18
Adding Props to a Terrain 25-21
Extracting Props from a Terrain 30-8
Adding Props from a Feature Layer 25-23
Viewing a List of Props 25-25
Selecting Props 25-25
Setting the Opacity of Props 25-26
Changing a Prop's Type 25-27
Changing a Prop's Position or Orientation 25-27
Changing a Prop's Model Definition 25-27

25.1. Terrain Agility and Composability

VR-Vantage applications are terrain-agile. Rather than being restricted to loading terrains in a limited number of formats that have been created using various terrain-creation software, VR-Vantage lets you compose terrains from a variety of supported data formats. You can combine elevation data (terrain patches), imagery, and feature data to create terrains.

VR-Vantage offers the following basic methods for creating terrains:

- Compose a terrain by loading individual elevation, image, and feature data files. This method is best for relatively small terrains that do not require paging. It supports all of the coordinate systems that VR-Vantage supports. The Makland (*Makland.mtf*) and VR-Village (*VR-Village.mtf*) terrains provided with VR-Vantage are examples of this method. For more information, please see "Creating a Composed Terrain," on page 25-5.
- Specify a set of files to load using an earth file. Use this method to stream terrain data from local or remote sources. This method is best for terrains with large amounts of elevation or image data, which would benefit from paging. For more information, please see "Connecting to Terrain Servers," on page 26-2.

When you compose a terrain, you can save it as a MAK terrain file (MTF), which is essentially a description of all of the data used to visualize the simulated 2D and 3D environment. Later, you can open the MTF file and VR-Vantage will automatically load all of the data that you loaded manually when you first created the terrain.

These two methods of loading terrain data are not mutually exclusive. You can stream terrain using an earth file and also load hand-modeled terrains as terrain patches. *MAK Earth (online).mtf* is an example of combining streamed terrain from VR-theWorld Server, and hand-modeled terrain (the Ala Moana inset in Honolulu, Hawaii, USA).

You can save terrains in a cached format that loads terrain data more quickly than if the terrain is created from the source data. For details, please see "Configuring File Caching," on page 29-2.

25.1.1. Supported File Formats

VR-Vantage can import elevation, terrain, images, models, and feature data. Some types of files can contain more than one type of information. For example GeoTiff files can have elevation data or just have image data. The following sections list the supported formats.



In theory, VR-Vantage can load any format supported by osgEarth if you use the earth file format to load them. However, VR-Vantage does not include drivers for all of the formats supported by osgEarth. Furthermore, when using osgEarth, VR-Vantage only supports geocentric terrains. If you are uncertain about support for a particular format, please contact support@mak.com.

Terrain Data

VR-Vantage imports terrain data directly as terrain patches. It supports the following formats:

- MAK terrain database (*.gdb*).
- MAK encrypted data format (*.medf*).
- MAK Stealth (6.2 and earlier) format (*.mtl*).
- MAK terrain project or terrain descriptor file (*.mtp*, *.mtd*).
- OpenFlight (*.flt*).
- MetaFlight (*.mft*).
- CTDB 4, 7, 7L (*.c4b*, *.c7b*, *.c7l*).
- Shape terrain data (*.shp*).
- TerraPage (*.txp*).
- 3DS.
- OBJ.
- OSG.
- LWO.

Elevation Data

VR-Vantage supports the following types of elevation data:

- DTED (*.dt0*, *.dt1*, *.dt2*, *.dted*).
- GeoTiff.
- DEM.
- Arc/Info Binary Data (*.adf*).
- Other formats supported by osgEarth.

Supported Image Formats

You can import images in the following formats:

- GeoTiff (*.tif*, *.tiff*).
- JPEG (*.jpg*, *.jpeg*).
- PNG.
- RGB.
- Windows bitmap (*.bmp*).
- GIF.
- ECW (Windows only).
- Other formats supported by osgEarth using the GDAL driver. For a list of formats supported by GDAL, go to http://www.gdal.org/ and select the Supported Formats link. For more information about earth files, please see "Connecting to Terrain Servers," on page 26-2 and Chapter 26, *Streaming Data Using Earth Files*.

Supported Feature Data Formats

VR-Vantage supports the following feature data formats:

- Shape files.
- S-57.
- OGR vector formats referenced in earth files.

25.1.2. Feature Layers and Props

A terrain can have objects such as buildings and trees embedded in it. These objects cannot be manipulated in this format. However, VR-Vantage lets you extract these types of objects from the terrain and render them independently as props.

Props are terrain elements that you can manipulate as separate objects. You can add props to a terrain by extracting them from a terrain patch or from a feature layer.

A feature layer is a file that contains feature data such as point features, linear features, and areal features. You can extract point features as props. You can extract props from MAK Terrain Format (GDB) files, shapefiles (SHP), or DFAD data (DFD).

For more information, please see "Adding Props to a Terrain," on page 25-21.

25.2. Creating a Composed Terrain

The most common ways to compose a terrain in VR-Vantage are as follows:

- Load individual terrain components through the VR-Vantage GUI and then save the terrain as an MTF file.
- Create an earth file that specifies all the terrain components to load. Then load the earth file and save it as an MTF file.
- Load an earth file and add local terrain components. Then save the terrain as an MTF file.

This section describes the first option, creating a terrain by loading individual elevation, imagery, and feature files. (Using earth files is the preferred method. For details about using earth files, please see Chapter 26, *Streaming Data Using Earth Files*.) The general procedure for composing a terrain this way is as follows:

1. Choose **File** → **New Terrain**, or click the New Terrain button () on the Terrain Toolbar. If any terrain data is loaded, it is removed.
2. Add elevation data, as described in "Adding Elevation Data (Terrain Patches) to a Terrain," on page 25-7.
3. Optionally, add imagery, as described in "Adding Images to a Terrain," on page 25-9.
4. Optionally, add feature data, as described in "Adding a Feature Layer," on page 25-14.
5. Optionally, add a dynamic ocean layer, as described in "Adding a Dynamic Ocean Layer," on page 25-15.
6. Optionally, add props or extract props, as described in "Adding Props from a Feature Layer," on page 25-23 and "Extracting Props from a Terrain," on page 30-8.
7. Save the terrain, as described in "Saving a Terrain," on page 25-6.

25.2.1. Considerations and Limitations for Building Terrains

As you compose a terrain, consider the following issues:

- When you load DTED as a terrain patch, VR-Vantage supports DTED level 0, 1, and 2, as follows:
  - DTED is loaded using a Flat Earth projection.
  - Level 0 DTED files have 1 KM post spacing. All posts are loaded.
  - Level 1 DTED files have 100 M post spacing. Every fifth post is loaded.
  - Level 2 DTED files have 20 M post spacing. Every fifth post is loaded.

  The maximum cell size you can load is 1 degree.
- The larger a terrain database is (as measured by the number of polygons), the more it will affect performance.

Loading DTED

By default, when you load DTED files as terrain patches, the files are loaded with their altitude referenced to WGS84. If you want the altitude to reference a different vertical datum (most likely EGM96), you must provide a geoid grid file for that vertical datum. If you use a geoid grid file, the altitudes in the DTED file are adjusted based on the information in the grid file.

To use a geoid grid file, set the environment variable MAK_GEOID_GRID to the path of the file.



This section does not apply to terrains loaded using an earth file.

For information about EGM96 and geoid files, please see the following URL:

http://earth-info.nga.mil/GandG/wgs84/gravitymod/egm96/egm96.html

25.2.2. Saving a Terrain

When you save a terrain, VR-Vantage saves information about the terrain patches, imagery, and feature layers that you used to create the terrain. Saving a terrain does not save out data to a file format comparable to OpenFlight or GDB that can be loaded into other applications or converted into other formats. If accelerated file loading is enabled, the data gets saved in a format that can be loaded more quickly than from its native format. (For details about accelerated file loading, please see "Configuring File Caching," on page 29-2.) Terrains get saved with the extension *.mtf*.

To save a terrain:

1. Choose **File** → **Save Terrain**, or click the Save Terrain button () on the Terrain Toolbar. The Save Terrain dialog box opens.
2. Type a name for the terrain.
3. Click Save.

25.3. Adding Elevation Data (Terrain Patches) to a Terrain

To build a terrain database, you load elevation data, imagery, and feature data. Then you save the data collection in MTF format. Elevation data is added as terrain patches. Note the limitations in "Considerations and Limitations for Building Terrains," on page 25-5.

! Each time you add a terrain patch, VR-Vantage resets the origin and coordinate system. Therefore, if you add multiple terrain patches, any patches whose origin and coordinate system do not match the patch that is loaded last will not be usable.

To add a terrain patch to a terrain database:

1. Choose **File** → **Add Terrain Patch**, or click the Add Terrain Patch button (🌿) on the Terrain Toolbar. The Add Terrain Patch dialog box opens and an Open dialog box opens on top of it.

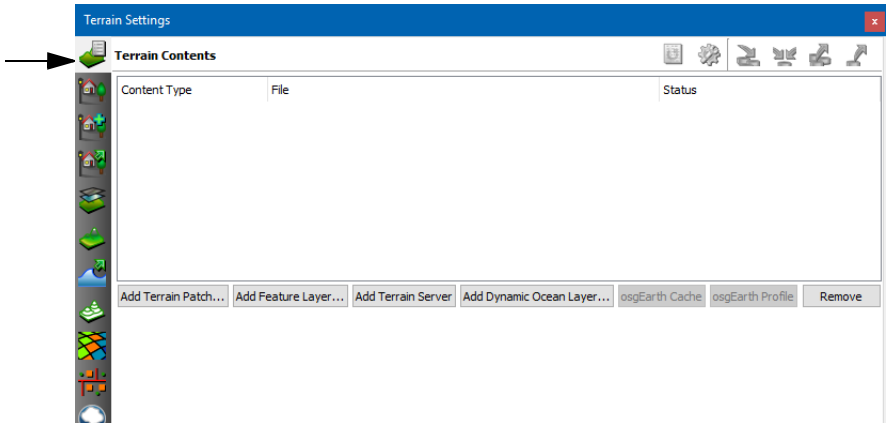


Figure 25-1. Terrain Contents page

2. Select a file in a supported format.
3. Click Open. The Add Terrain Patch dialog box displays details about the terrain patch (Figure 25-2).

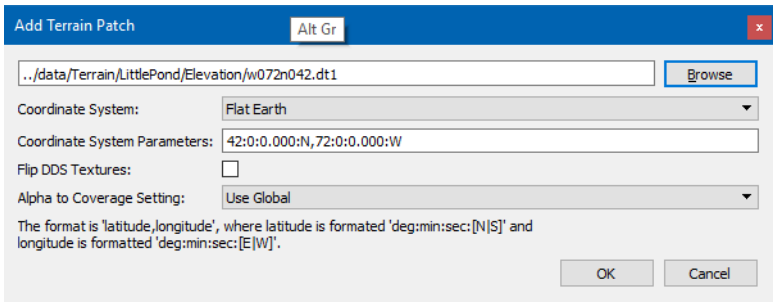


Figure 25-2. Add Terrain Patch dialog box

4. If the terrain patch has DDS textures that are upside down relative to VR-Vantage's expectations, select Flip DDS Textures. This setting does not affect paged terrains. (For information about DDS textures, please see "Displaying DDS Textures Correctly," on page 29-6.)
5. Optionally, change the Alpha to Coverage Setting. The options are:
   - Use Global. Use the setting for alpha to coverage from the Display Settings dialog box, Render Settings page.
   - Force Off. Use the regular rendering way of handling transparent objects.
   - Force On. Turn on alpha to coverage for the entire terrain patch (excluding props since they are handled as models).

   For more information about alpha to coverage, please see "Alpha to Coverage," on page 5-9.
6. Click OK. The terrain patch is added.
7. Optionally, add additional terrain patches.
8. Save the terrain.

25.4. Adding Images to a Terrain

Depending on your terrain source, you may have terrain polygonal data, but no texture data or feature data to provide a realistic 3D display. You can drape images (raster maps) over the terrain to provide a more pleasing display.



Adding very large images (multi-100 GB) may cause VR-Vantage to crash. If you need to load very large images, load them using an earth file and the GDAL driver. The *SanLuisObispo-Imagery.earth* file demonstrates how to load an image using this method. You can also use the GDAL driver to load raster formats that VR-Vantage cannot load directly. For a list of formats supported by GDAL, go to http://www.gdal.org/ and select the Supported Formats link.

To add raster maps to the terrain:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Raster Maps page.
3. Click the Add button (+) that is above the list of raster maps. A line is added to the Raster Maps list (Figure 25-3).

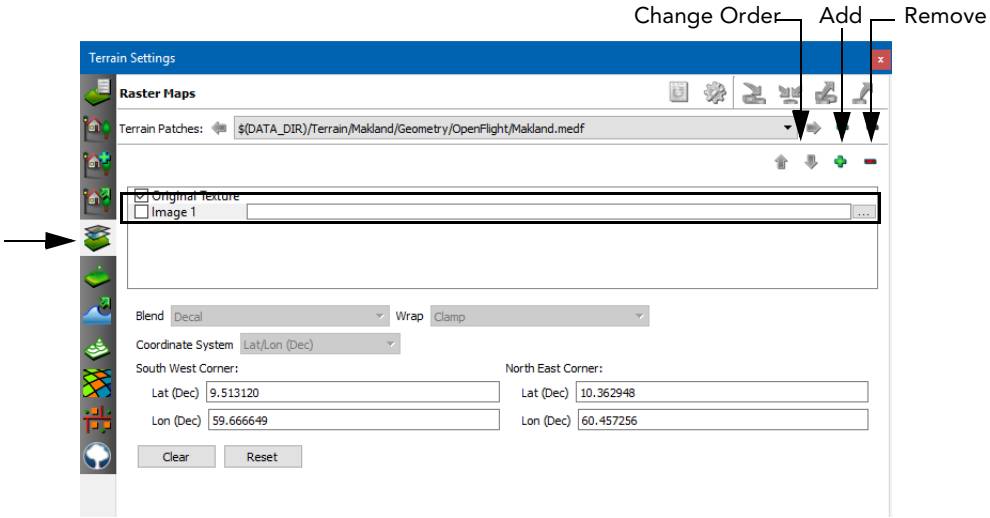


Figure 25-3. Raster Maps page

4. Click the Browse button to the right of the new line. The Image File dialog box opens.
5. Select the raster image that you want to add to the terrain.
6. Click OK. The image is listed in the Raster Maps window. To view it in the main window, select the check box to the left of the entry (Figure 25-4).

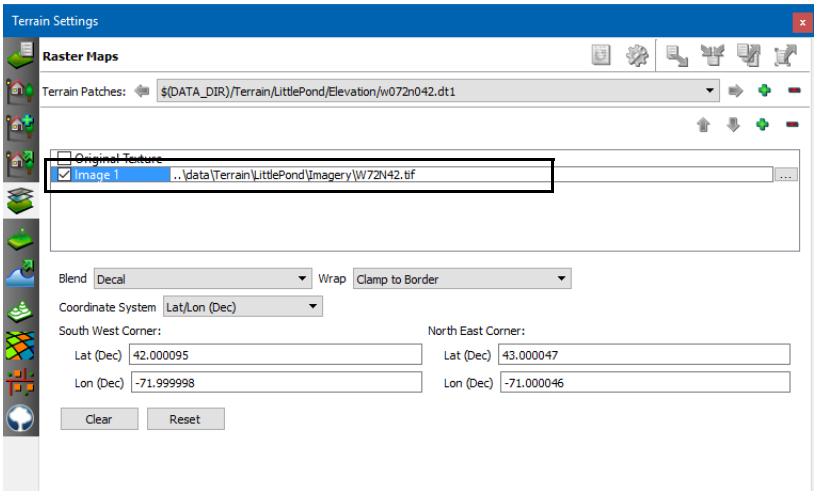


Figure 25-4. Making an image visible

7. Optionally, specify the attributes of the image, as follows:

   – Blend specifies how the image interacts with underlying images, as described in Table 25-1:

Table 25-1: Blend settings

Blend setting	Description
Decal	Covers the polygons with the image. Use it to apply an opaque texture or a texture with transparency. (This is done using an alpha channel.)
Modulate	Color values of the image modify the intensity of the colors of the polygon.
Blend	Blends polygon and texture colors.
Replace	Similar to decal.
Add	Adds the pixel values of the source and base colors.

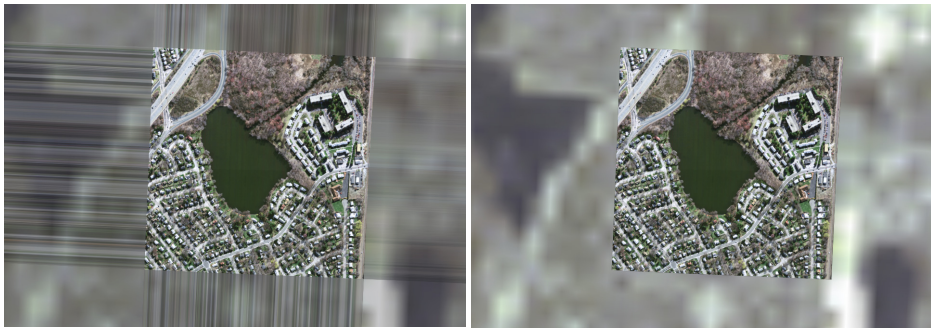
– Wrap specifies whether or not the image is repeated, and if so, how it is repeated, as described in Table 25-2, and as illustrated in Figure 25-5 and Figure 25-6:

Table 25-2: Wrap settings

Wrap setting	Description
Clamp	Extends the border pixels of the image to the edge of the terrain or the next continuous image, blending them with the pixels of other images. It does not obscure underlying images.
Clamp to Edge	Extends the border pixels over the entire terrain, obscuring any other image.
Clamp to Border	Places the image once, at the specified coordinates.
Repeat	Tiles the image onto the terrain.
Mirror	Tiles the image onto the terrain, alternating between the true image and a mirror image.

– Coordinate System specifies the coordinate system to use to place the image. Most options are self-explanatory. The Texture coordinate system specifies UV (X,Y) coordinates for placing textures on the terrain.

– Location specifies the coordinates for the Southwest (SW) and Northeast (NE) corners of the image. If the image is a GeoTIFF, the coordinates are in the image file. If the image file does not have coordinates, VR-Vantage stretches it to cover the terrain patch. If you do not want the image to cover the entire terrain, you can specify the coordinates.



Clamp | Clamp to border

Figure 25-5. Raster image wrap effects (Clamp and Clamp to Border)

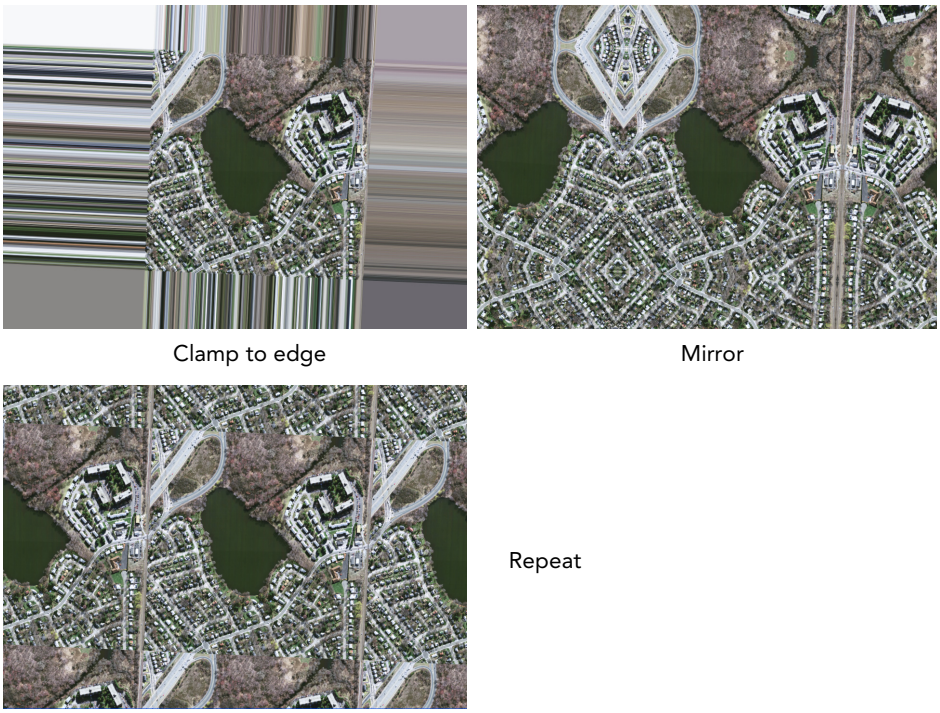


Figure 25-6. Raster image wrap effects (Clamp to Edge, Mirror, Repeat)

25.4.1. Changing the Display Order of Raster Maps

Raster maps are displayed in the order in which they are listed in the Terrain Settings dialog box, Raster Maps page. If they overlay the same portion of the terrain, the topmost image is displayed. You can change the order in which image files are listed, thereby changing which one is displayed in VR-Vantage.

Figure 25-7 shows two views of the detailed portion of the LittlePond terrain, one with the default image order, and one with the order of raster images reversed.



Figure 25-7. Changing the raster map order

To change the order of a raster map.

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Raster Maps page (Figure 25-3).
3. Select the raster image whose order you want to change.
4. Click the Up Arrow or Down Arrow above the upper right corner of the list of raster maps. The image moves in the list and the window changes as appropriate.

25.5. Adding a Feature Layer

A feature layer is a file that contains data about geographic features on the terrain. This includes linear features, such as roads, and point features, such as trees and buildings. You can create props from point features. A feature layer can be a MAK Terrain Format (GDB) file, a shapefile (SHP), or DFAD data (DFD).

When VR-Vantage imports a shapefile, it uses the name of the shapefile to indicate what type of feature it contains. Filenames are mapped to feature types in *./appData/importConfig/filename_shp_map.txt*. For example, the file name *ROADL.shp* is mapped to feature 129, which is a road. If the shapefiles that you need to import are not listed in *filename_shp_map.txt*, you can add them by editing the file.



- Adding a feature layer does not, by itself, make any visible difference to a terrain. To see the feature data, you must extract the data as props and map the props to model definitions. For details, please see "Adding Props to a Terrain," on page 25-21.
- The number of layers you can add depends on your graphics card. If you add a layer and it does not appear, try reducing the number of layers you have.

To add a feature layer:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Terrain Contents page (Figure 25-8).

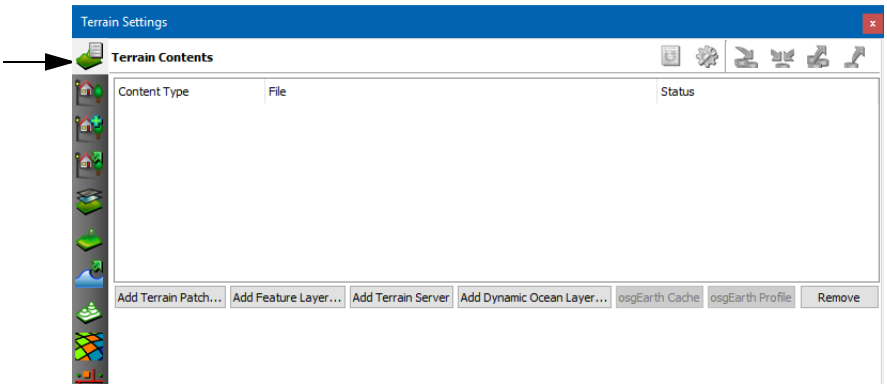


Figure 25-8. Terrain Contents page

3. Click Add Feature Layer. The Select Source File for Feature Layer dialog box opens.
4. In the Select Source File for Feature Layer dialog box, select the file that you want to use for this feature layer.
5. Click Open. The source file is added to the list of terrain contents.

25.6. Adding a Dynamic Ocean Layer

VR-Vantage can simulate dynamic ocean effects, such as waves, swell, and chop. To simulate the ocean, your terrain must have a dynamic ocean layer. You can add an ocean layer directly or you can add one as part of the process of extracting water textures.

If your terrain has textures assigned to water polygons, then you should extract them. They will be replaced by the ocean layer. If you do not extract the water textures and you simply add an ocean layer, the ocean layer and the water texture will interfere with each other in a process called Z-fighting and create an unacceptable display (Figure 25-9). This problem occurs because the ocean layer and the texture are both at the same height (0, or sea level).

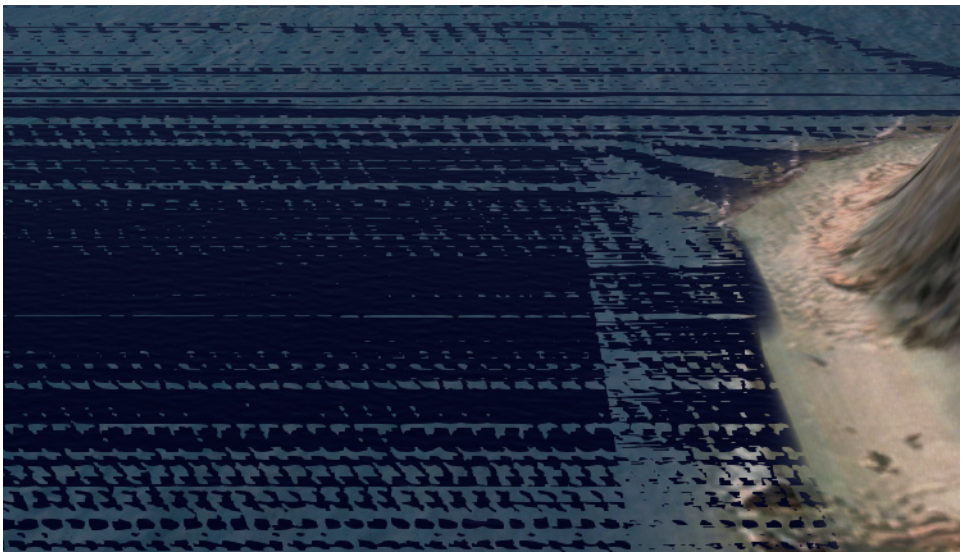


Figure 25-9. Z-fighting

If your terrain does not have a specific water texture assigned to polygons, your only option is to add an ocean layer directly. However, since the ocean layer will be at the same elevation as the terrain texture, you may experience Z-fighting. You may also have to raise the ocean height to actually see the ocean effects.

Figure 25-10 shows the Makland terrain with a water texture and an ocean layer. In the left view, the ocean layer has a height of 0 meters. You cannot see it. In the right view, the ocean layer is set to 1 meter. Now it is visible.

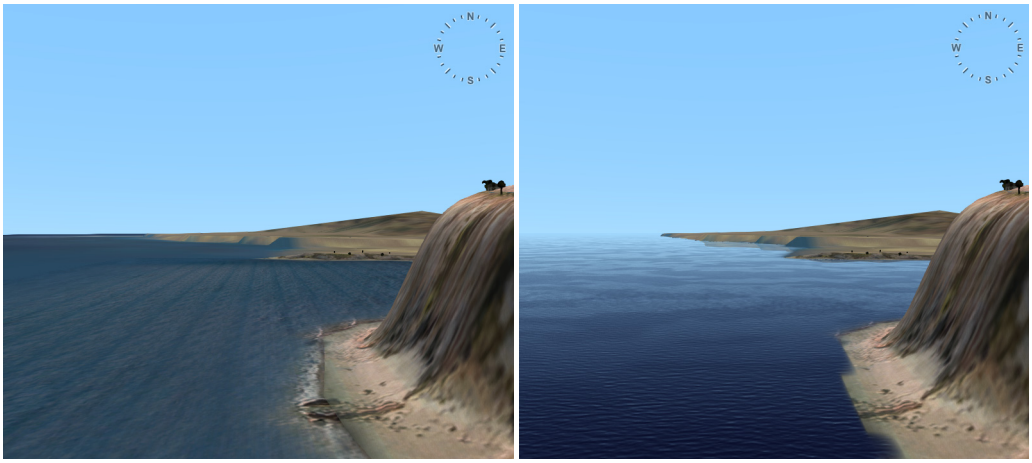


Figure 25-10. Ocean height

25.6.1. Extracting Water Textures and Adding an Ocean Layer

To extract textures and add an ocean layer:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Extract Ocean page (Figure 25-11).

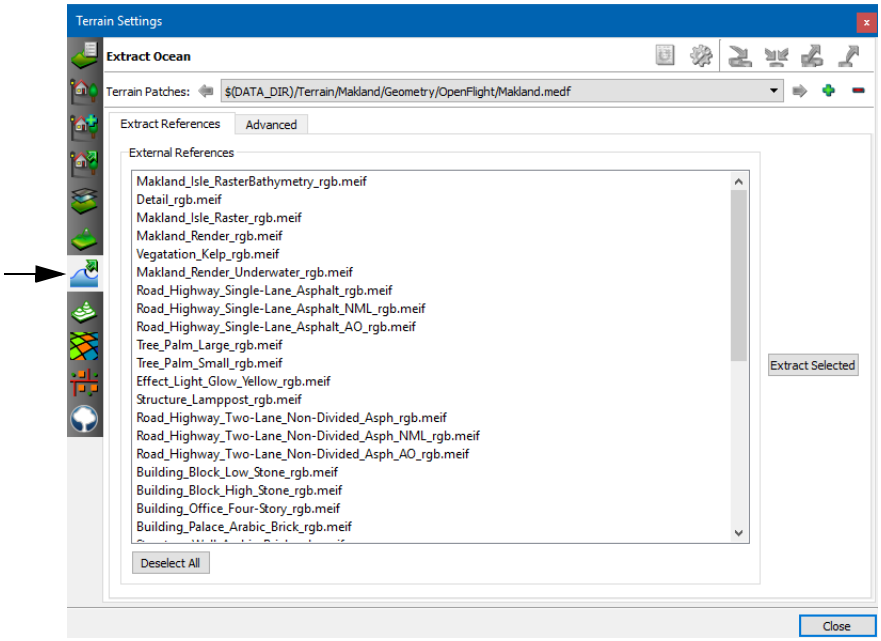


Figure 25-11. Terrain Settings dialog box, Extract Ocean page

3. In the External References list on the Extract References tab, select the water textures that you want to extract.

4. Click Extract Selected. The textures are removed and an ocean layer is added to the Terrain Contents page.
5. Select the Terrain Contents page (Figure 25-12).

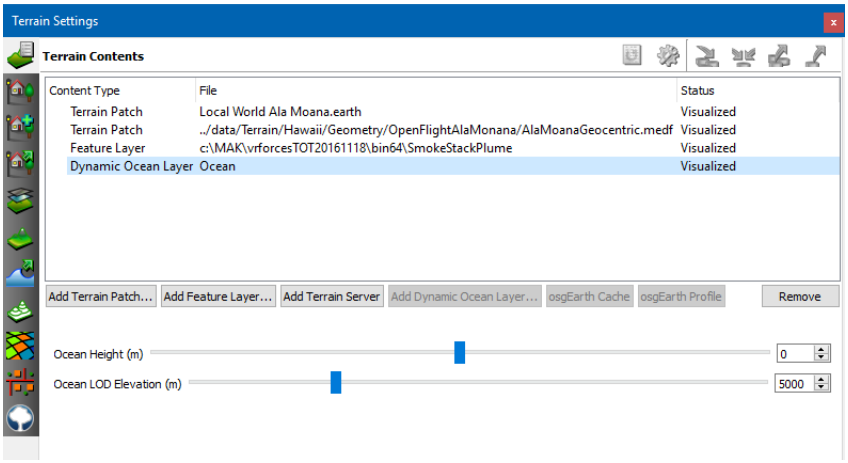


Figure 25-12. Terrain Contents with dynamic ocean layer

6. Adjust the Ocean Height so that the dynamic ocean is visible. This might require raising the ocean above sea level.
7. Optionally, adjust the Ocean LOD Elevation. For information about the Ocean LOD Elevation, please see "Setting the Ocean LOD Elevation," on page 25-18.

25.6.2. Adding a Dynamic Ocean Layer Directly

Ocean layers get created at a height of 0 meters, that is, mean sea level.

To add a dynamic ocean layer:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Terrain Contents page.
3. Click Add Dynamic Ocean Layer. (You can only have one ocean layer.) A layer is added to the Terrain Contents list and two parameters are added to the bottom of the dialog box (Figure 25-12).
4. Adjust the Ocean Height so that the dynamic ocean is visible. This might require raising the ocean above sea level.
5. Optionally, adjust the Ocean LOD Elevation. For information about the Ocean LOD Elevation, please see "Setting the Ocean LOD Elevation," on page 25-18.

25.6.3. Setting the Ocean LOD Elevation

Enabling dynamic ocean affects performance. Since the effects are hard to see once the observer is high above the ocean, VR-Vantage stops displaying them when the observer is more than 5000 meters above sea level. You can change this cutoff point by adjusting the Ocean LOD Elevation.

To set the ocean LOD elevation:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Terrain Contents page (Figure 25-12).
3. Adjust the Ocean LOD Elevation slider or change the value in the box.

25.6.4. Configuring the Ocean Height Map

The ocean height map is used to blend the dynamic ocean with the coastline. You can configure the height map resolution, its texture size, and the threshold for regenerating the height map. All of these settings interact to affect visual quality and performance.

Height map resolution is specified as meters per pixel. The height map texture size is specified in pixels. Therefore these two settings determine the length of one side of the (square) area that the height map covers. For example, if the value for height map resolution is 10 meters per pixel and the texture size is 4096 pixels, this results in a height map that is 40960 meters on a side.

As height map resolution increases and texture size decreases, visual quality may improve, but performance can degrade. Conversely, decreasing resolution and increasing the texture size can improve performance at a cost of visual quality. You will need to experiment to determine the best values for your simulation.

If you are working with a small terrain or scenes in which the observer is close to sea level, you may want to use a higher resolution. If you are less interested in how the coastline looks, you can use a lower resolution. Figure 25-12 shows a coastline at different levels of resolution.

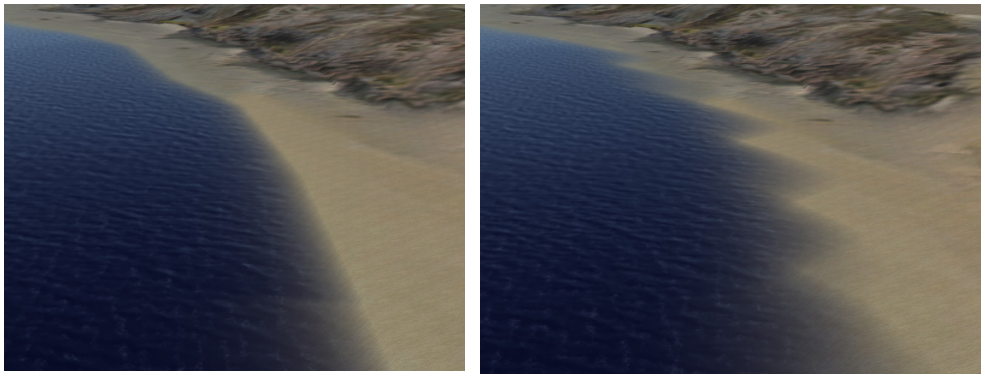


Figure 25-13. Ocean height map resolution

Setting the surface transparency also affects visualization of the coastline. Experiment with these values to achieve the best results. For details about water transparency, please see "Configuring Marine Conditions," on page 11-14.

Setting the Ocean Height Map Resolution

The ocean height map resolution is specified at meters per pixel.

To set the ocean height map resolution:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Ocean Settings page (Figure 25-14).

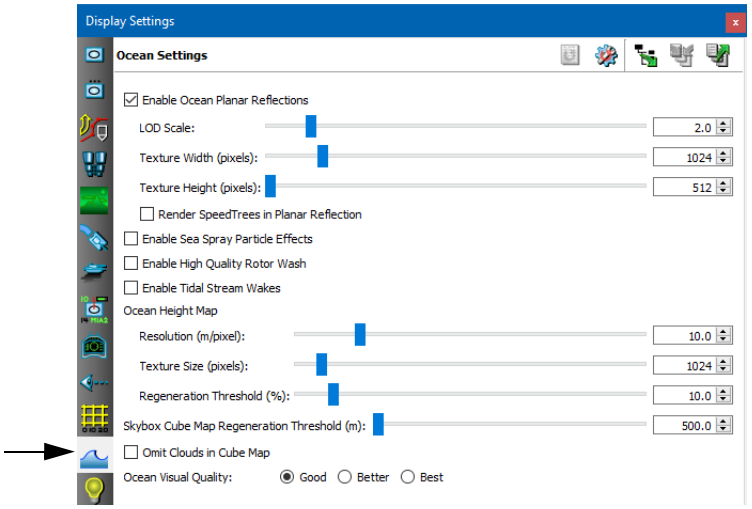


Figure 25-14. Ocean Settings

3. Adjust the Ocean Height Map Resolution slider, or enter a value in the box.

Setting the Ocean Height Map Texture Size

The ocean height map texture size is specified in pixels.

To set the ocean height map texture size:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Ocean Settings page (Figure 25-14).
3. Adjust the Ocean Height Map Texture Size slider, or enter a value in the box.

Setting the Ocean Height Map Regeneration Threshold

The ocean height map is recalculated as the observer moves. Generating the height map is computationally expensive. Therefore, it is done only when the observer moves by more than a specified threshold. You can adjust this threshold if you want to recalculate the height map more frequently. The default is 10% (0.1). This means that the observer must move by 10% of the length of one side of the height map before the height map is regenerated.

To set the ocean height map regeneration threshold:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Ocean Settings page (Figure 25-14).
3. Adjust the Ocean Height Map Regeneration Threshold slider, or enter a value in the box.

25.7. Adding Props to a Terrain

Props are objects like buildings, lampposts, or trees that exist independently of the underlying terrain elevation data. You can extract props from a terrain patch or add them from a feature layer. You can view a list of the props in the terrain, specify their opacity, and enable or disable their visibility. You can change a prop's type and you can change the model definition used to render it. You can also create new prop types to meet your simulation needs.

25.7.1. Extracting Props from a Terrain Patch

When you extract props from a terrain patch, you have to tell VR-Vantage where to find prop data, what type of prop to use, and what model definitions to map the props to, as follows:

- Where to search: A terrain patch, such as an OpenFlight terrain, often references multiple files that define models that are incorporated into the geometry of the terrain. VR-Vantage lets you specify the external files from which to extract props or you can specify a search expression that looks in all referenced files. If you select a referenced file, the process is quicker because VR-Vantage searches for props only in the specified file.
- What prop type to use: You can select a prop type from a list, specify a type of your choice, or let VR-Vantage auto-generate prop types. If you choose Auto Generate, VR-Vantage makes the prop type match the name of the referenced file. For example, a prop extracted from the file *Bridge_Arch_Stone.flt* would be given the prop type *Bridge_Arch_Stone.medf*.
- Model definition to use: You can select a model definition from a list, or choose Use Extracted Geometry. If you accept the default value, Use Extracted Geometry, VR-Vantage creates props using the geometry in the terrain patch. For example, if you chose *Trees.flt* as the referenced file, VR-Vantage creates props using the tree geometry in that file.

VR-Vantage extracts props only from the selected terrain patch. If your terrain has multiple terrain patches, you must repeat the procedure for each terrain patch from which you want to extract props.

For an example of how to extract props from a terrain patch, please see "Extracting Props from a Terrain," on page 30-8.

To extract props from a terrain:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Extract Props page (Figure 25-15).

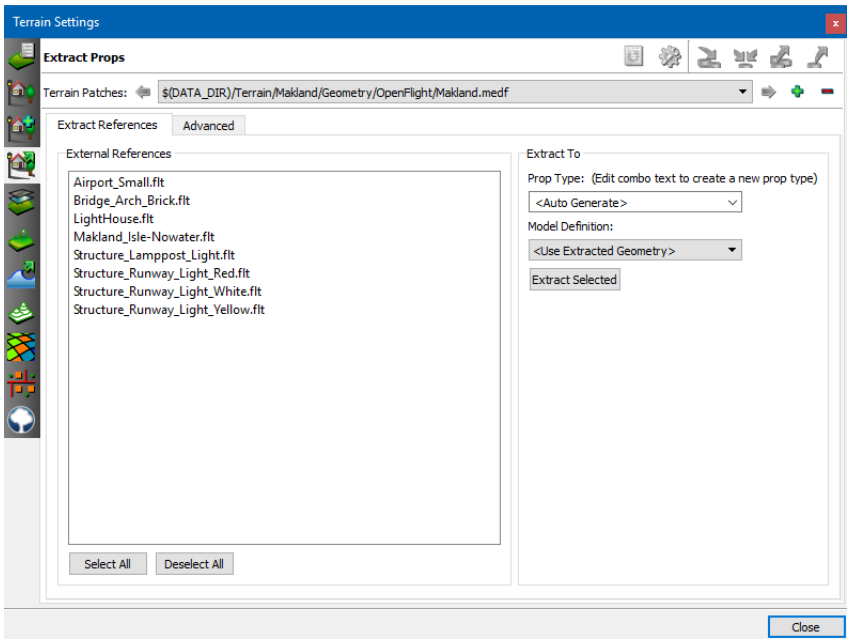


Figure 25-15. Extract Props page

3. In the Terrain Patches box, select the terrain patch from which you want to extract props.
4. To extract props by specifying external files, select the Extract References tab. To extract props using a search expression, select the Advanced tab.
5. If you chose the Extract References tab, in the External References list, select the files from which you want to extract props.

   If you chose the Advanced tab, type the search expression in the Search Expression box. Select the Regular Expression check box if you are using a regular expression.



VR-Vantage uses perl regular expression syntax.

6. Optionally, in the Prop Type box, select the type that you want to assign to the extracted props or type a prop type.
7. Optionally, in the Model Definition box, select the model definition that you want to apply to the extracted props.
8. If you are using the Advanced tab, click Extract. If you are using the Extract References tab, click Extract Selected. VR-Vantage extracts the props, if any. You can view the list of props in the Props list on the Edit Existing Props page.

25.7.2. Adding Props from a Feature Layer

A feature layer source file contains information about point, line, or areal features. Feature information is stored in a table in which each row is a feature and each feature is defined by the values in an arbitrary number of columns. To add a prop from a feature layer, you must build a rule that specifies the column headings and values for the feature you want to import. Then you must associate a prop type and model definition with the prop. You also must specify whether or not to ground clamp the prop.

To add a prop from a feature layer:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Add Props page (Figure 25-16).

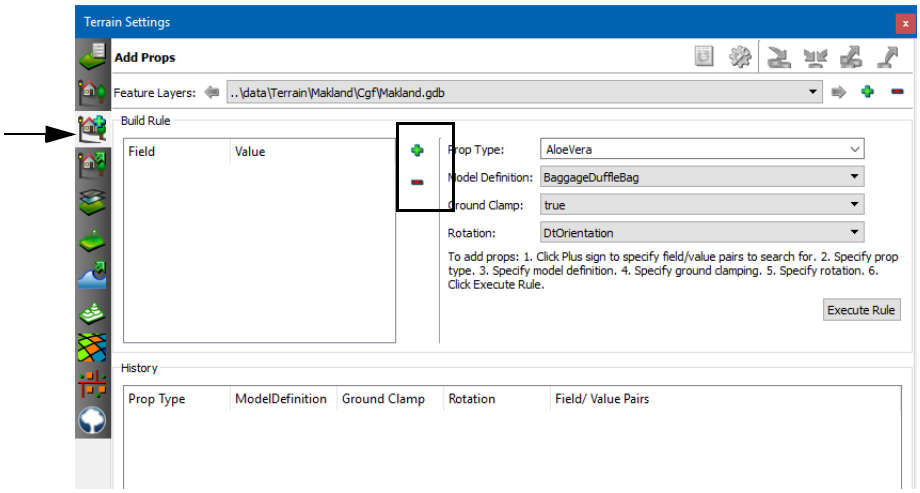


Figure 25-16. Terrain Settings, Add Props page

3. In the Feature Layers list, select the feature layer from which you want to import the props.
4. Click the Add button (+) to the right of the Field/Value window (Figure 25-16). The Build Field/Value Pair dialog box opens (Figure 25-17).

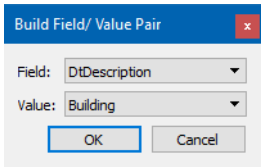


Figure 25-17. Build Field/Value Pair dialog box

5. In the Field list, select a field. Fields are the column headings in the feature definition.
6. In the Value list, select a value. Values are one or more values from the definition table that are available for the column heading.



It is your responsibility to understand the feature definitions for the features that you want to import as props.

7. Click OK. The Field/Value pair is added to the window (Figure 25-18).

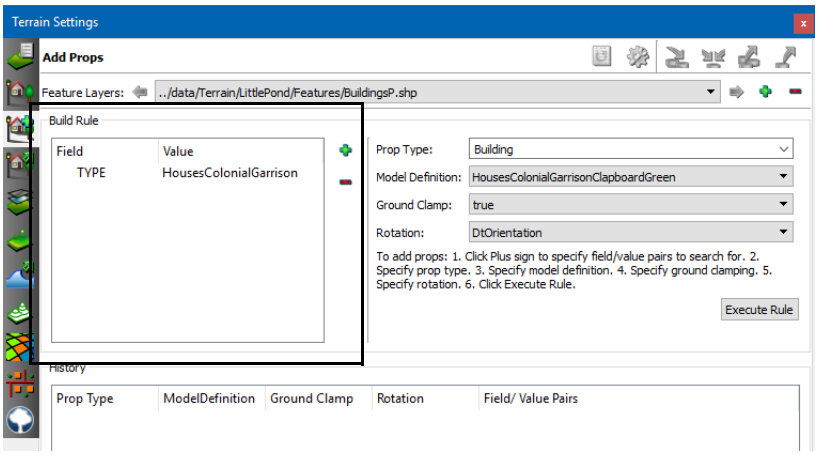


Figure 25-18. New Field/Value pair

8. Add additional Field/Value pairs as needed to define the prop that you want to add.
9. In the Prop Type list, select the prop type to assign to the prop you want to add.
10. In the Model Definition list, select the model definition you want to use for this prop.
11. In the Ground Clamp box, specify whether or not to ground clamp this prop.
12. Optionally, set the rotation.
13. Click Execute Rule. The prop is extracted from the feature layer and is listed in the History window.

25.7.3. Viewing a List of Props

To view a list of the props in the terrain:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Edit Existing Props page (Figure 25-19).

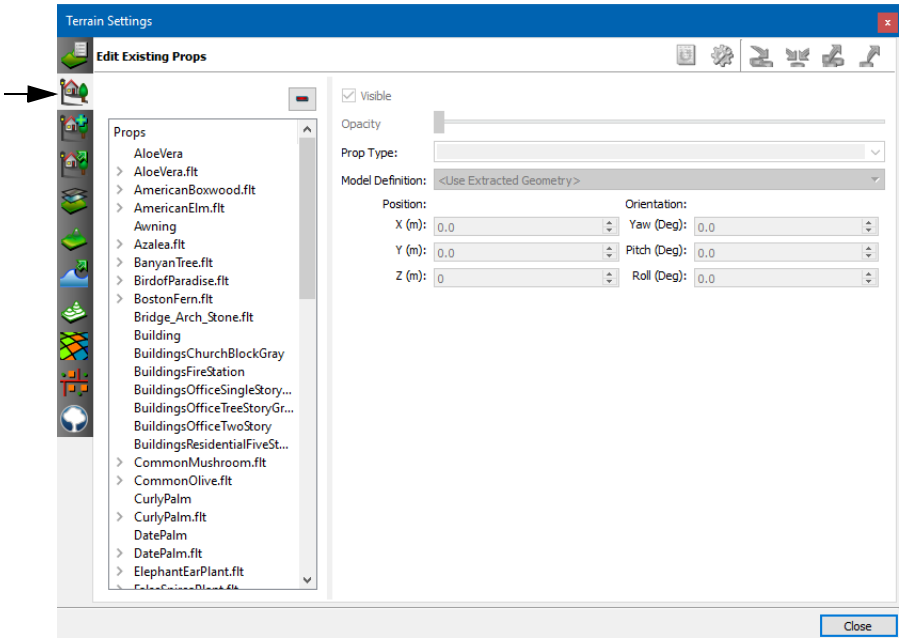


Figure 25-19. Edit Existing Props page

3. Expand the list to view the entries under a particular type of prop.

25.7.4. Selecting Props

➤ To select a prop, click it in the window or in the Objects List Panel, or select it in the list of props on the Terrain Settings dialog box, Edit Existing Props page.

Press **`Ctrl`**+click to select multiple props.



By default, the Objects List Panel just lists Entities. To see props in the list, change the filter.

25.7.5. Setting the Opacity of Props

You can set the opacity of props. This can be useful if you want to see what is inside a building or behind a prop. Figure 25-20 illustrates a building at two levels of opacity.



100% 40%

Figure 25-20. Prop opacity

To set a prop's opacity:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Edit Existing Props page (Figure 25-19).
3. In the list of props select the props whose opacity you want to change, or select a prop in the window. The Opacity slider is enabled.
4. Slide the Opacity slider to the right to increase opacity. Slide it to the left to decrease opacity.

Making a Prop Fully Transparent or Opaque

You can quickly set a prop to be fully transparent or opaque.

➤ To make a prop fully transparent, right-click the prop and on the popup menu, choose **Make Transparent**.

➤ To make a transparent prop fully opaque, right-click the prop and on the popup menu, choose **Make Opaque**.

25.7.6. Changing a Prop's Type

Every prop has a type. Props are organized by type in the list on the Edit Existing Props page.

To change a prop's type:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Edit Existing Props page (Figure 25-19).
3. In the list of props, select the props whose type you want to change.
4. Select a prop type from the Prop Type list, or type a new type name. The selected props are moved to the new prop type in the Props window.

25.7.7. Changing a Prop's Position or Orientation

You can change a prop's position and orientation. You can change the orientation of multiple props at the same time without affecting their position. However, if you select multiple props and change position, they are all placed in the same position.

To change a prop's position or orientation:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Edit Existing Props page (Figure 25-19).
3. In the list of props, select the prop whose position or orientation you want to change.
4. To change position, type or select new values in the X, Y, or Z boxes.
5. To change the orientation, type or select new values in the Yaw, Pitch, or Roll boxes.

25.7.8. Changing a Prop's Model Definition

You can change the model definition used to render a prop. For information about model definitions, please see Chapter 31, *Schemas*. For an example of changing a prop's model definition, please see "Test the Model Mapping," on page 38-10.

To change a prop's model definition:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the Edit Existing Props page (Figure 25-19).
3. In the list of props, select the prop whose model definition you want to change.
4. In the Model Definition list, select a model definition. The selected props are rendered using the new model definition.



26. Streaming Data Using Earth Files

This chapter provides an introduction to the syntax of Earth files and explains how to configure them to stream various types of terrain and feature data.

Connecting to Terrain Servers 26-2
Adding Terrain Server Connections 26-5
Connecting to Terrain Servers through a Proxy 26-5
A Simple Earth File 26-6
Including External Files into Earth Files 26-8
Supported Drivers 26-9
Loading Multiple DTED Files 26-9
Processing Streamed Data 26-10
Adding Feature Layers to an Earth File 26-11
Configuring Point Features 26-13
Configuring Linear Features 26-14
Configuring Areal Features 26-15
Using Cut-in Sites for High Resolution Insets 26-16
Using the Boundary Generation Tool 26-17
Extruded Buildings 26-18
Loading CDB Databases 28-2
Earth File Options 26-19
Streaming Buoys and Beacons 26-20
Specifying Model Definitions for Streamed Buoys 26-21
Modeling Buoys Using Texture Atlases 26-24
Configuring Streaming Beacons 26-25
Configuring Streamed Navigation Lights 26-26
Configuring Seasonal Buoys and Beacons 26-27
Streaming Daymarks 26-28
Streaming Topmarks 26-28
Generating Signal Sequences 26-29

26.1. Earth Files

An earth file is an XML format configuration file used to display terrain databases using osgEarth. The file format is documented at http://docs.osgearth.org/en/latest/. The information in this chapter supplements the information on the osgEarth web site. In particular, it explains how to configure streaming features, for which there is no GUI in VR-Vantage.



VR-Vantage only supports geocentric terrains in earth files.

26.2. Connecting to Terrain Servers

VR-Vantage supports the osgEarth libraries for creating terrains from remote or local data sources (Figure 26-1). Terrain servers are configured in earth (*.earth*) files. They support streaming of terrain data and are, therefore, ideal for displaying terrains that have large amounts of data.



Figure 26-1. Geocentric terrain

You can manipulate streamed terrains with the keyboard and mouse and can save them as an MTF terrain for quick loading.

VR-Vantage includes several terrain server configurations for VR-TheWorld Online, a server hosted by VT MAK. You must have an internet connection to connect to these servers. It also includes server configurations that stream terrain files included with VR-Vantage. These servers, such as, *Local World.earth* and *SanLuisObispo-Topo.earth*, are examples of how you can use earth files to load local data.

! Integration of osgEarth into VR-Vantage allows you to connect to data sources. However, it does not provide a license to use any data sources. It is your responsibility to comply with the licensing provisions of any site to which you connect.

"Creating a Terrain Server," on page 30-2, is a tutorial that shows how to create a terrain server connection that serves local data.

---

i When you view a geocentric terrain and move the observer more than 6 million meters or so from the earth's surface, the view is forced to focus toward the center of the earth. The practical effect of this mode is that the left and right movement keys orbit the observer around the earth, rather than moving the entire earth to the left or right.

To connect to a terrain server:

1. Choose **File** → **New Scene**. The current terrain closes.
2. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
3. Select the Terrain Contents page (Figure 26-2).

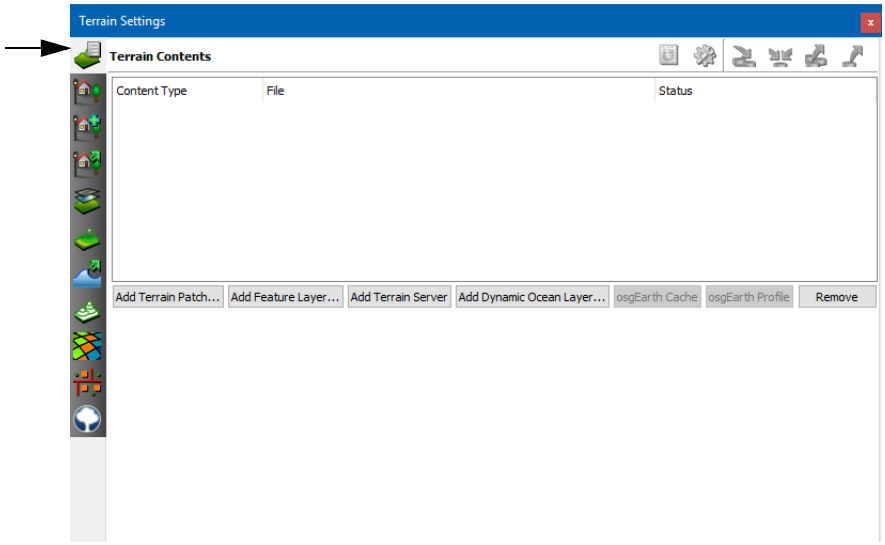


Figure 26-2. Terrain Contents

4. Click Add Terrain Server. The Add Terrain Server dialog box opens (Figure 26-3).

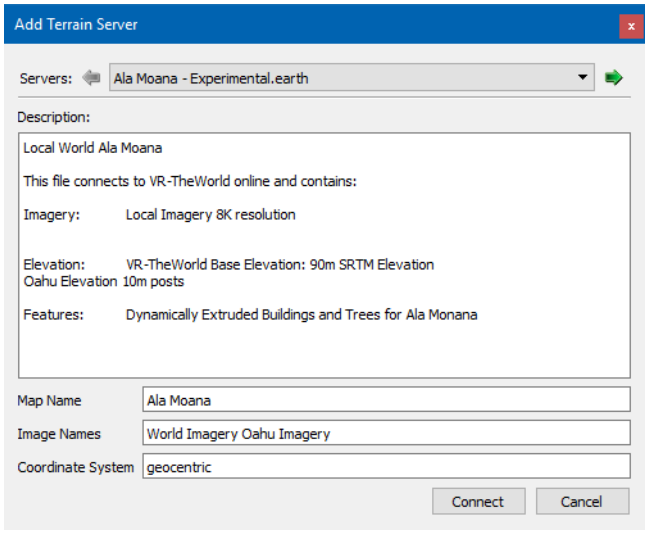


Figure 26-3. Add Terrain Server dialog box

5. Select a server from the Servers list. The dialog box displays information about the selected terrain server. It is not editable.
6. Click Connect. If the terrain data is available (remote sources might not always be accessible), it is displayed.

You can save a terrain server as part of a terrain or scene. For details, please see "Saving a Terrain," on page 25-6.

26.2.1. Adding Terrain Server Connections

You can add additional terrain server connections to the list provided with VR-Vantage. Terrain server connections are configured in XML files according to the requirements of the osgEarth libraries. The server configurations included with VR-Vantage are examples of what you can do. We recommend that you add new servers by copying an existing configuration that is similar to what you want and then editing it to point to your data.



Earth files must be placed in *./userData/servers*.

"Creating a Terrain Server," on page 30-2, is a tutorial that shows how to create a terrain server that serves local data.

26.2.2. Connecting to Terrain Servers through a Proxy

If you connect to the network through a proxy, you may need to set some environmental variables to be able to connect to external terrain servers. VR-Vantage has limited proxy capability. It supports basic authentication and may support domain authentication if your proxy allows it to be accessed through basic authentication. To set up authentication, set the following environmental variables:

- OSG_CURL_PROXY. Specifies the proxy machine.
- OSG_CURL_PROXYPORT. Specifies the proxy port.
- OSGEARTH_CURL_PROXYAUTH=username:password. Specifies the user name and password for the proxy.

26.3. A Simple Earth File

The following is an example of a simple earth file. It has one image layer and one elevation layer:

```
<map name="earth" type="geocentric" version="2">
  <!--Load a simple base image of the world-->
  <image name="World Imagery" driver="gdal">
    <url>../../data/Terrain/World/Imagery/World4km.tif</url>
  </image>
  <!--Load a folder full of terrain data as an elevation source-->
  <heightfield name="World Elevation" max_data_level="16" max_level="16"
 driver = "gdal">
    <!--To load the files in a directory, just point the URL to a
 directory instead of a file-->
    <url>../../data/Terrain/World/Elevation/earthnew/w001001.adf</url>
    <tile_size>32</tile_size>
  </heightfield>

  <model name="Country-Names" driver="feature_geom">
    <features name="Country-Names" driver="ogr">
      <url>../../data/Terrain/World/Features/world.shp</url>
    </features>
    <styles>
      <style type="text/css">
        Country-Names {
        text-provider: annotation;
        text-content:  [cntry_name];
        text-priority: [pop_cntry];
        text-halo:     #3f3f7f;
        text-font:     arial.ttf;
        text-size:     16;
        text-remove-duplicate-labels: true;
        altitude-clamping: terrain-drape;
        }
      </style>
    </styles>
  </model>
  <options>
      <lighting>true</lighting>
      <elevation_interpolation>triangulate</elevation_interpolation>
      <terrain first_lod="1">
          <min_tile_range_factor>4</min_tile_range_factor>
          <skirt_ratio>0.1</skirt_ratio>
      </terrain>
  </options>
</map>
```

Table 26-1 describes the XML elements in the file and their attributes.

Table 26-1: Earth file elements and attributes

Element	Attribute	Description
<!-- -->		Comment block.
map		Root level element. All other elements must be in the map block.
	name	The name of the map.
	type	The projection. VR-Vantage only supports geocentric terrains, which can be specified as `geocentric` or `globe`.
	version	The VR-TheWorld Server version.
image		Specifies an image layer.
	name	The name of the layer.
	driver	The driver to use to fetch tiles, for example, TMS, WMS, or GDAL.
	url	The location of the files. This could be a web address or a local path.
heightfield		Specifies an elevation layer.
	name	The name of the layer.
	driver	The driver to use to fetch tiles, for example, TMS, WMS, or GDAL.
	url	The location of the files. This could be a web address or a local path.
model		Base element for specifying a feature layer. For more information, please see "Adding Feature Layers to an Earth File," on page 26-11.
	name	Name of the model element. This can be any text.
	driver	Driver to use for creating models.
options		Specifies options that affect the entire map. For more information, please see "Earth File Options," on page 26-19.

26.3.1. Including External Files into Earth Files

Earth files can include external files. This can make management of complex terrains easier. The included files must be XML files. There are two ways to include external files, the `xi:include` element and use of earth file templates.

The following example shows how to use `xi:include`:

```
<map name="readymap.org" type="geocentric">
  <xi:include href="readymap_imagery.xml"/>
  <xi:include href="readymap_elevation.xml"/>
</map>
```

When you reference a document using `xi:include`, it can only contain a single XML node, for example:

```
<image>
  ...
</image>
```

Since using an `xi:include` element for each node that you want to import might require many such elements, you can include multiple nodes in a single file by using earth file templates, an alternative include format. Use the syntax `{% include filename %}`. This syntax is used in some of the earth files shipped with VR-Vantage. The following lines are from *MAK Earth (online).earth*:

```
<!-- Boston area models -->
  {% include vrtheworld-streaming.buildings-boston.xml %}

<!-- Hawaii area models -->
  {% include localdata.hawaii-maritime-navigation.xml %}
  {% include vrtheworld-streaming.buildings-oahu.xml %}
  {% include vrtheworld-streaming.airport-hnl.xml %}
```

The included XML files are located in the same directory.

! In an earth file, if you comment out an include line that uses the `{% include filename %}` syntax, the included file must not have any comments or it will not be processed properly. This can result in terrain errors that are not easily diagnosed. Files included with the `xi:include` syntax do not have this restriction.

26.4. Supported Drivers

Drivers are software components that allow you to load one or more types of terrain data. MAK has not tested all of the drivers supported by osgEarth. VR-Vantage supports the following drivers:

- tms
- gdal
- composite
- feature
- ogr
- feature_geom
- feature_custom
- wfs
- aglite.

Other drivers supported by osgEarth may work. If you have questions about driver support, please contact support@mak.com.

26.5. Loading Multiple DTED Files

You can load multiple DTED files from an earth file. To do so, in the `heightfield` element, specify a directory instead of a specific file, for example (from *SanLuisObispo-Topo.earth*):

```
<heightfield name="SanLuisObispo Elevation" driver = "gdal">
  <url>../../data/Terrain/SanLuisObispo/Elevation</url>
  <extensions>dt0;dt1;dt2</extensions>
  ...
</heightfield>
```

The `extensions` element lets you specify which files to load based on their file extensions. If you do not include an extensions list, the GDAL driver tries to load every file in the directory.

26.6. Processing Streamed Data

VR-Vantage has several software modules, which we generically call "processors," that search through streamed-in data looking for things like SpeedTree trees, objects that need effects textures attached, buoys, beacons, and so on. In order to do this efficiently, processors register feature names that they want to process. The registration data is in *./importConfig/registrant_feature_list.csv*.

For example, in *registrant_feature_list.csv*, it states that the processor for SpeedTree trees, speedTreeManager, is mapped to `Vegetation`. In *MAK Earth (online).earth*, SpeedTree trees are configured as follows:

```
<model name="trees" driver="feature_custom">
  <features name="Vegetation" driver="wfs">
```

The feature name is Vegetation. This means that it will be processed by the speedTreeManager.

Similarly, the featureDataBroker, which is responsible for creating streamed buoys, beacons, and the lights and marks that might be attached to them, is mapped to `s57-data` and `dynamicFeatures`. (Processors can be mapped to more than one feature type.) Examples of using this processor are:

```
<model name="LateralBuoysUS5HA52M" driver="feature_custom"
  enabled="true">
  <features name="s57-data" driver="ogr">
```

and

```
<model name="TaxiwayLights" driver="feature_custom">
  <features name="dynamicFeatures" driver="ogr">
```

Lastly, the indirectGeometryManager, which is responsible for adding point feature replaced models to the indirect rendering system, is mapped to buildings and other static streamed geometry. This allows those models to be rendered using the high performance indirect rendering system. An example of using this processor for a model section is:

```
<model name="LargeBuildings" driver="feature_custom">
  <features name="buildings" driver="ogr">
```

registrant_feature_list.csv file is fully commented and provides additional information about this scheme.

26.7. Adding Feature Layers to an Earth File

You can add the following types of features to a terrain:

- Point features, such as trees, buoys, and buildings.
- Linear features, such as roads and waterways.
- Areal features, such as forests (areal trees), bodies of water, and extruded buildings.

You can add the XML for feature layers to an earth file.

Feature layers are added to an earth file using the `model` element. The following example shows the hierarchical relationships of a `model` element and its sub-elements. Table 26-2 describes the `model` element, its attributes, and some of its sub-elements. (A particular `model` element may not have all of the sub-elements described.)

```
<model>
  <features>
    <url></url>
  </features>
  <layout>
    <tile_size_factor></tile_size_factor>
    <level></level>
  </layout>
  <styles>
  <library>
    <url></url>
  </library>
    <style></style>
    <selector></selector>
  </styles>
  <lighting></lighting>
</model>
```

Table 26-2: `model` element attributes

Element	Attribute	Description
model		Base element for specifying a feature layer.
	name	Name of the model element. This can be any text.
	driver	Driver to use for creating models.
The `features` and `layout` elements specify how to get and load the feature data. The `features` block is mandatory. `layout` is optional.		
features		Specifies source of data.
	name	Name of the element.
	driver	Driver to use to get the data.
url		Location of the feature data.
layout		Specifies how to load the data.

Table 26-2: `model` element attributes

Element	Attribute	Description
tile_size_factor		The data specified in the features element gets cut into tiles. This attribute specifies how aggressively to tile it.
level	name	The level specifies the level (LOD) beyond which not to display the features.
	max_range	Distance beyond which the feature is not displayed.
selector	class	Specifies the type of feature.
styles		The `styles` element and sub-elements specify how to render the scene. This element is mandatory.
library		A texture library.
	name	Name of the texture library.
	url	The location of the library.
style		A specific style. The name, which precedes the curly brace ({) can be any text you want. For information about the attributes that you can specify in a style, please see http://docs.osgearth.org/en/-latest/references/symbology.html.
lighting		Specifies whether or not to enable lighting. Usually true.



When you load a file using the OGR driver (specified in the `feature` element), the layer name the name of the file (minus the extension). Other files (like S-57) can have multiple layers, and the names are decided by a format-specific algorithm. For example, an S-57 file has an enumerated list of layer names.

26.7.1. Configuring Point Features

Point features, such as trees, use the `model` element. The following code, from *MAK Earth (online).earth*, adds point trees:

```
<model name="Alamoana point trees" driver="feature_custom">
  <!-- streaming point trees -->

  <features name="Vegetation" driver="ogr">


  <url>../../data/Terrain/Hawaii/Features/AlaMoanaVegetationP.shp</url>
    <typename>5</typename>
  </features>
  <layout>
    <tile_size_factor>3</tile_size_factor>
  </layout>

  <styles>
    <style type="text/css">
        Vegetation {
        marker:              "Trees" + [veg] + "ST" ;
        marker-scale:        1;
        marker-placement: vertex;
        altitude-clamping:   terrain;
        altitude-resolution: 0.00008583068847656250;
        altitude-offset:     0;
        }

    </style>
  </styles>
  <clustering>True</clustering>
    <lighting>true</lighting>
</model>
```

26.7.2. Configuring Linear Features

Linear features, such as roads, rivers, or railways, use the `model` element. The following code, from *MAK Earth (online).earth*, configures roads:

```
<model name="Oahu_Roads" driver="feature_geom" overlay="true"
  enabled="false">

  <features name="Roads" driver="wfs">
    <url>http://vr-theworld.com/vr-theworld/features/wfs</url>
    <typename>55</typename>
  </features>

  <!-- Sets the name property on each geometry to the value of the "name"
  attribute -->
  <feature_name>[name]</feature_name>

    <!-- Appearance details -->
    <style type="text/css">
      Oahu_Roads {
      stroke: #ff0000;
      stroke-width: 10.0;
      altitude-offset: 10;
      }
    </style>
</model>
```

26.7.3. Configuring Areal Features

Areal features, such as forests, use the `model` element. The following code, from *MAK Earth (online).earth*, configures a forest with three types of trees:

```
<!-- Forest -->
<model name="trees" driver="feature_custom">
  <features name="Vegetation" driver="wfs">
    <url>http://vr-theworld.com/vr-theworld/features/wfs</url>
    <typename>7</typename>
  </features>
  <layout>
    <tile_size_factor>3</tile_size_factor>
    <crop_features>true</crop_features>
    <level max_range="8000" class="trees-1"> </level>
    <level max_range="2000" class="trees-2"> </level>
    <level max_range="1000" class="trees-3"> </level>
  </layout>
  <styles>
    <style type="text/css">
      trees-1 {
      marker:              "TreesNorwaySpruceST";
      marker-placement:    random;
      marker-density:      800;
      marker-scale:        1;
      altitude-clamping:   terrain;
      altitude-resolution: 0.0001;
      marker-random-seed:  1;
      }
      trees-2 {
      marker:              "TreesColoradoBlueSpruceST";
      marker-placement:    random;
      marker-density:      800;
      marker-scale:        1;
      altitude-clamping:   terrain;
      altitude-resolution: 0.0001;
      marker-random-seed:  2;
      }
      trees-3 {
      marker:              "TreesAmericanElmST";
      marker-placement:    random;
      marker-density:      800;
      marker-scale:        1;
      altitude-clamping:   terrain;
      altitude-resolution: 0.0001;
      marker-random-seed:  3;
      }</style>
    </styles>
    <clustering>true</clustering>
    <lighting>true</lighting>
</model>
```

26.8. Using Cut-in Sites for High Resolution Insets

Users of terrain servers, such as VR-TheWorld Server, often want to add high resolution sites of particular interest while using the lower resolution terrain provided by the terrain server for the rest of the world. You can add a high resolution terrain patch to a terrain server, but it overlays the terrain provided by the server and this can cause unexpected visual problems. Ideally, you should cut out the relevant areas of terrain provided by the server and add your high resolution terrain to the cut out area. For example, the VR-Vantage terrain *MAK Earth (online).mtf* has two terrain patches, *MAK Earth (online).earth*, which provides the base terrain from VR-TheWorld Server, and *./data/Terrain/Hawaii/Geometry/OpenFlightAlaMoana/AlaMoanaModified-GeocentricProxy.osg*, which is a locally stored high resolution inset.

The `mask` element lets you cover the base terrain data with your hand-modeled data. A mask element in an earth file looks similar to the following:

```
<mask name="HonoluluRunway26R-8L-mask" driver="feature" >
  <features driver="ogr">
    <geometry>
      POLYGON((-157.91489143 21.32914026 3.016, -157.91525638
  21.32914026 3.016, -157.91666238 21.32912541 3.016, -157.91697822
  21.32934847 3.016, -157.91709263 21.32968744 3.016, -157.91709263
  21.32968744 7.016, -157.91709263 21.32968744 11.016, -157.91709263
  21.32968744 15.016, -157.91710262 21.32968706 15.016, -157.91697822
  21.32934847 16.016, -157.91709263 21.32968744 16.016, -157.91710262
  21.32968706 15.016))
    </geometry>
  </features>
</mask>
```

VR-Vantage includes a tool, the osgEarth Boundary Generation Tool, that can calculate the geometry of an OpenFlight file. To cut out a section of terrain from the terrain server, add a `mask` element to the earth file. Use the output from the boundary tool for the data in the geometry element.

Table 26-3 describes the `mask` element and its sub-elements.

Table 26-3: `mask` element

Element	Attribute	Description
mask		Element for cutting in geometry.
	name	Name of the mask.
	driver	The driver to use.
features		List of features in the mask.
	driver	The driver to use. OGR reads raw vector feature data from shapefiles and other formats.
geometry		Specifies the 3D boundaries of the site. You must know this information.
	geometry_url	The location of the file with the feature data.

26.8.1. Using the Boundary Generation Tool

The Boundary Generation Tool takes an OpenFlight file and calculates the bounding geometry of the file. It returns the coordinates in a text file. The general process of cutting in a site is as follows:

1. Run the Boundary Generation Tool against an OpenFlight file.
2. Insert the output of the boundary generation tool in a `mask` element in your earth file. (For details about `mask` elements, please see "Using Cut-in Sites for High Resolution Insets," on page 26-16.)
3. Load the terrain.
4. Add the FLT file as a terrain patch.
5. Save the terrain.

Running the Boundary Generation Tool

The output from the Boundary Generation Tool tool, in *boundary.txt* or the output file you specify, looks similar to the following:

```
POLYGON((-157.91489143 21.32914026 3.016, -157.91525638 21.32914026
  3.016, -157.91666238 21.32912541 3.016, -157.91697822 21.32934847
  3.016, -157.91709263 21.32968744 3.016, -157.91709263 21.32968744
  7.016, -157.91709263 21.32968744 11.016, -157.91709263 21.32968744
  15.016, -157.91710262 21.32968706 15.016, -157.91697822 21.32934847
  16.016, -157.91709263 21.32968744 16.016, -157.91710262 21.32968706
  15.016))
```

The first two vertices are the latitude and longitude. The third vertex is the height, in meters. The remainder of the vertices map the outline of the geometry.

To run the boundary generation tool:

1. Open a console window.
2. Change directory to *./bin64*.
3. Run the tool:

   ```
   osgearth_boundarygen [--out file_name --no-geocentric
     --convex_hull --verbose --view --precision precision
     --tolerance tolerance] model_file
   ```

   Table 26-4 describes the options.

Table 26-4: osgEarth Boundary Generation Tool command-line options

Option	Description
`--convex_hull`	Calculate a convex hull instead of a full boundary.
`model_file`	The input model file.
`--no-geocentric`	Skips geocentric reprojection. Use for flat databases.
`--out` *`file_name`*	Specifies the output file for boundary geometry. Default: *boundary.txt*.

Table 26-4: osgEarth Boundary Generation Tool command-line options

Option	Description
`--precision` *precision*	Specifies the number of significant digits in the output coordinates. Default: 12.
`--tolerance` *tolerance*	Specifies the tolerance for combining similar vertices along a boundary. Default: 0.005.
`--verbose`	Prints progress messages to the console.
`--view`	Show the result in a 3D window.

26.9. Extruded Buildings

If you are simulating in an urban environment, you want your terrain to have realistic 3D buildings. However, modeling hundreds or thousands of buildings by hand would be an arduous and expensive task. osgEarth can extrude buildings from areal features and apply textures to them to create a realistic environment with relatively little work. Publicly available GIS data that has footprint and height data can provide the input.

The following code (from *MAK Earth (online).earth*) adds extruded buildings to the terrain:

```
<!-- Extruded buildings Oahu -->
<model name="buildings" driver="feature_geom">

  <!-- Streamed buildings -->
  <features name="buildings" driver="wfs">
    <url>http://vr-theworld.com/vr-theworld/features/wfs</url>
    <typename>65</typename>
  </features>

  <layout>
    <tile_size_factor>3</tile_size_factor>
  </layout>

  <styles>
    <library name="us_resources">
      <url>http://vr-theworld.com/vr-
theworld/filemanager/download/public/Library/catalog.xml</url>
    </library>

    <style type="text/css">
      buildings {
        extrusion-height:      [hgt2d];
        extrusion-flatten:     true;
        extrusion-wall-style:  building-wall;
        extrusion-roof-style:  building-rooftop;
        extrusion-random-seed: 1;
        altitude-clamping:     terrain;
        altitude-resolution: 0.0001;
      }
      building-wall {
        skin-library: us_resources;
        skin-tags:    building;
```

```
            skin-random-seed: 1;
            fill:         #ffffff;
          }
          building-rooftop {
            skin-library: us_resources;
            skin-tags:    rooftop;
            skin-tiled:   true;
            skin-random-seed: 1;
            fill:         #ffffff;
          }
        </style>
      </styles>
      <lighting>true</lighting>
      <cluster_culling>false</cluster_culling>

    </model>
```

For descriptions of the elements in a `model` block, please see Table 26-2.

26.10. Earth File Options

An earth file can have `option` elements that affect the entire map. Table 26-5 describes the option elements.

Table 26-5: `option` element

Element	Attribute	Description
lighting		Enables or disables lighting for the entire map. This is usually true.
elevation_interpolation		Specifies how OSG interprets elevation between grid posts. If you are using VR-Forces or visualizing a VR-Forces simulation, set this to triangulate to ensure correlation.
cache		Specifies caching for terrain data.
	type	The type of caching to use.
	path	The location of the data cache.

26.11. Streaming Buoys and Beacons

VR-Vantage can display models of buoys and beacons specified in shapefiles and S-57 data, which is streamed in using earth files.

Streamed features (including buoys) are organized into layers. In the earth file, the names in the **styles** block specify the layer the streamed features will belong to. By default, VR-Vantage defines seven layers for buoys. They correspond to the different buoy layers in an S-57 file. These layers are named LateralBuoys, SafeWaterBuoys, ISDBuoys, RegulatoryBuoys, InstallationBuoys, Cardinal-Buoys, and SpecialPurposeBuoys.

The configuration file *./appData/importConfig/s57_layer_list.csv* specifies which style names in an earth file correspond to recognized streamed feature types. The file currently contains two lines. The first starts with the keyword "Buoys" and is followed by a comma-separated list of all the style names that will be recognized (and streamed in) as buoys. The second line starts with the keyword "Beacons" and is followed by the list of recognized beacon styles. They can be anything you like, but need to correspond with whatever style names you use in the earth file.

The following is an example of a **style** block for a buoy (Special Purpose Buoys):

```
<model name="buoys" driver="feature_custom">
  <features name="Special Purpose Buoys" driver="ogr">
   ...
  </features>
  <layout>
   ...
  </layout>
  <styles>
    <style type="text/css">
        <!-- | | | This is the layer name -->
        <!-- v v v                        -->
      SpecialPurposeBuoys {
      ......
      }
    </style>
  </styles>
```

26.11.1. Specifying Model Definitions for Streamed Buoys

Within each style block, you specify the model definition to use for a streamed buoy or beacon. Since buoys and beacons have different styles and within each style have different colors and markings, VR-Vantage can map the S-57 data to buoy and beacon models in the following ways:

- Specify a model definition for each type of buoy. Buoy color and numbering is the same for all buoys.
- Generate a model definition name using a pattern and S57 attributes.
- Specify a model definition for a unique buoy based on the buoy's name.
- Use a texture atlas to model buoys. (A texture atlas is an image that has sub-images that can be referenced by a coordinate system. Those sub-images can be applied to 3D models.)

All of these methods are supported by VR-Vantage and are described in this section. However, VR-Vantage uses texture atlases for all streaming buoys and beacons.

Specifying a Model Definition for a Type of Buoy

To specify the model definition for a type of buoy, you specify it as the marker name:

```
<styles>
   <style type="text/css">
    SpecialPurposeBuoys {
    marker: "BuoysSpecialPurpose" ;
    ...
```

In this case, the model definition “BuoysSpecialPurpose” is used for all buoys streamed in for that SpecialPurposeBuoys layer.

This approach is the simplest way to specify model definitions for buoys.

Generating a Model Definition Name for a Buoy

Instead of specifying a model definition, you can have VR-Vantage build a model definition using the buoy’s attributes. This approach is more complex than specifying one model definition for a type of buoy, but it is more flexible. You can accommodate many variations of a given type of buoy without having to specify many model definitions.

Buoy attributes can be strings or enumerated values (such as shape (BOYSHP), color (COLOUR), and color pattern (COLPAT)).

The following code results in model definitions like “Buoys352”.

```
<styles>
  <style type="text/css">
    SpecialPurpose {
    marker: "Buoys" + [BOYSHP] + [COLOUR] + [COLPAT];
```

Creating and managing model definitions with coded numbers in this way is difficult. Therefore, VR-Vantage provides a mechanism to map the numeric attribute values to strings.

The enumerated value-to-string mappings are in *./appData/importConfig/s57_attr_map.csv*. You can change them to any strings that you want to use. Given this information, you can specify your model definition as a pattern and provide the enumerated attribute values. VR-Vantage uses them to decode and build the requested model definition. The pattern and attribute values are specified as a colon (:)-separated list of the following form:

```
"pattern:Attribute1=value:Attribute2=value" ...
```

The pattern itself indicates where attribute values should be replaced using '[Attribute]' substrings. For example, consider an input string such as:

```
"Buoys[COLOUR][BOYSHP]:COLOUR=5:BOYSHP=3"
```

The default mappings for COLOUR and BOYSHP in *s57_attr_map.csv* are as follows:

```
BOYSHP,,Nun,Can,Spherical,Pillar,Spar,Barrel,Super,Ice
COLOUR,,White,Black,Red,Green,Blue,Yellow,Grey,Brown,Amber,Violet,Orange,
  Magenta,Pink
COLPAT,,HorizontalStriped,VerticalStriped,DiagonalStriped,Squared,
  Striped,BorderStriped
```

Therefore, the model definition would be BuoysBlueSpherical, since the enumerated COLOUR value '5' (the fifth value in the list) is mapped to "Blue", and the BOYSHP value '3' is mapped to "Spherical". Some of these attributes, such as COLOUR, may be lists of enumerated values. The strings for enumerated lists are concatenated. The following input string:

```
"Buoys[BOYSHP][COLOUR][COLPAT]:COLOUR=3,1:BOYSHP=2:COLPAT=2
```

results in the model definition "BuoysCanRedWhiteVerticalStriped".

To generate the input string like the one above from an earth file, you would use:

```
marker: "Buoys[BOYSHP][COLOUR][COLPAT]"
            + ":COLOUR=" + [COLOUR]
            + ":BOYSHP=" + [BOYSHP]
            + ":COLPAT=" + [COLPAT]
```

The order that the actual attribute/value pairs are encoded (after the pattern) does not matter.

Designators

Many buoys have some form of one to three numbers, letters, or both that appear on the buoy itself. S-57 does not provide an attribute that indicates what this symbol is, but it is usually found at the end of the OBJNAM attribute, such as “Boston South Channel Buoy 13” or “President Roads Junction Lighted Buoy PR”. If the OBJNAM attribute is provided, the model definition builder tries to extract this symbol. It must be 3 characters or less, and be all numerics, all capital letters, or a mix of the them. You can add this to a model definition pattern using the [DESIGNATOR] substring, for example:

```
marker: "Buoys[BOYSHP][COLOUR][COLPAT][DESIGNATOR]"
+ ":COLOUR=" + [COLOUR]
+ ":BOYSHP=" + [BOYSHP]
+ ":COLPAT=" + [COLPAT]
+ ":OBJNAM=" + [OBJNAM]
```

This code, when streaming in “Boston South Channel Buoy 13”, would result in a model definition of “BuoysCanGreen13”, since the BOYSHP attribute for that buoy maps to “Can” and COLOUR attribute to “Green”.

Unique Buoys

There may be cases where you want to use a specific model definition for a particular buoy. The buoy might be a special landmark that needs to be recognizable. The file *appData/importConfig/s57_modeldef_by_name_map.csv* contains a list of buoy names (the OBJNAM attribute) and model definition pairs. If the OBJNAM attribute value is provided in the marker string and if it matches one of the buoy names in *s57_modeldef_by_name_map.csv*, then the specified model definition is used instead of a generated definition.

26.11.2. Modeling Buoys Using Texture Atlases

This approach is more complex than specifying a model definition per type of buoy or building one from attributes. It requires understanding how to use texture atlases. However, this method allows you to have a large number of buoys in a scene with a wide variety of colors, patterns, and designators using a very limited number of models. This dramatically reduces the scene complexity (thereby increasing frame rate) when you use instancing. This is the approach that VR-Vantage uses for all buoys and beacons it ships with.

Each buoy shape (can, pillar, nun, and so on) is a separate model. Each model has a texture atlas that has all the different colors and patterns needed for that buoy shape. In the `marker` field (in the `model` statement) a simple model definition pattern is used:

```
Buoy[BOYSHP]
```

The other attributes are passed in the same fashion as described in "Generating a Model Definition Name for a Buoy," on page 26-21 when mapping attributes to model definition substrings:

```
marker: "Buoys[BOYSHP]"
  + ":BOYSHP=" + [BOYSHP]
  + ":COLOUR=" + [COLOUR]
  + ":COLPAT=" + [COLPAT]
  + ":OBJNAM=" + [OBJNAM] ;
```

In this example, all the attributes are mapped to strings, but the COLOUR and COLPAT strings are used to look up the appropriate uv texture coordinate offsets for the buoy. The texture-uv mapping file has the same name as the buoy model plus the extension *dtxc* (for example, *PillarBuoy.dtxc* and *Can.dtxc*).

The *.dtxc* files can support multiple atlases that may be used on the same model. VR-Vantage uses multiple atlases for buoys with topmarks, and beacons with daymarks and topmarks. (For information about topmarks and daymarks, please see "Streaming Daymarks," on page 26-28 and "Streaming Topmarks," on page 26-28.) The model definition for the buoy specifies a key (found in the *.dtxc* file) to be used for the main texture (`mainTextureKey`) and for the topmark's texture (`topmarkTexturekey`).

The buoy is also built with special geometry and a texture atlas for displaying the alpha-numeric designator. The geometry includes a switch to select between sites for placing one, two, or three alphanumeric characters. The characters themselves are then mapped from a texture atlas using the designator string found and a uv-coordinate texture mapping file that is shared by all buoys (and daymarks), named *LettersNumbers.ctxc*.

26.11.3. Configuring Streaming Beacons

You can specify beacon feature styles the same way you specify buoys. You can add any number of beacon styles names. The S57 attribute enumeration-to-string mappings for beacons are in *s57_layer_list.csv*.

The following code is an example of a beacon model block:

```
<model name="beacons" driver="feature_custom" enabled="true">
  <features name="Lateral Beacons" driver="ogr">
    <!-- From S-57 data -->
    <ogr_driver>S57</ogr_driver>
    <url>../../data/Terrain/Massachusetts/Features/US5MA10M.000</url>
    <layer>BCNLAT</layer>
  </features>
  <layout>
    <tile_size_factor>3</tile_size_factor>
  </layout>
  <styles>
    <style type="text/css">
      LateralBeacons {
      marker: "Beacons[BCNSHP]"
           + ":BCNSHP=" + [BCNSHP]
           + ":COLOUR=" + [COLOUR]
           + ":COLPAT=" + [COLPAT]
           + ":PERSTA=" + [PERSTA]
           + ":PEREND=" + [PEREND]
           + ":OBJNAM=" + [OBJNAM] ;
           marker-scale:        1;
           marker-placement: vertex;
           altitude-clamping:   terrain;
           altitude-resolution: 0.00008583068847656250;
           altitude-offset:     0;
      }
    </style>
  </styles>
</model>
```

Model definitions for beacons work the same way as for buoys. They are configured in *s57_attr_map.csv*. It can contain enumeration mappings for any enumerated S-57 attribute that you want to use to create model definitions. Default values are in *s57_defaults.csv*. This contains the default name and default model definition for each style group specified in *s57_layer_list.csv*. The delivered *s57_defaults.csv* contains:

```
# If the final model definition generated does not exist, one of these
  model definitions will be used
Buoys,DefaultModelDefinition,BuoysGreenPillar1
Beacons,DefaultModelDefinition,BuoysRedPillar2

# If the feature has no name, use one of these
Buoys,DefaultName,Unnamed Buoy
Beacons,DefaultName,Unnamed Beacon
```

26.11.4. Configuring Streamed Navigation Lights

Navigation lights are managed somewhat differently from other streamed features. These lights do not result in actual objects added to the scene. The information is used to create and attach light controllers to existing objects. The lights themselves need to already exist as part of the buoy or beacon model they appear on.

Each model that has a light should have a comment-tagged switch for choosing a light group of the specified color. Each light in turn has a switch for turning the light itself on or off. We use the tag `@dis light_color` to tag a color switch, with values 0-3 corresponding to white, green, red, and yellow. The switch to turn lights off and on is `@dis navigation_lights`.

VR-Vantage streams in light data using `model` blocks similar to those used for buoys and beacons. You can apply style names. *light_layer_list.csv* controls which styles will be read. *Massachusetts S-57 (online).earth* uses a single style name for all lights, **NavigationLights**. The following example shows a `model` block for a light:

```
<model name="LightsUS5MA10M" driver="feature_custom"  enabled="true">
  <features name="s57-data" driver="ogr">
    <!-- From S-57 data -->
    <ogr_driver>S57</ogr_driver>

    <url>../../data/Terrain/Massachusetts/Features/US5MA10M.000</url>
    <layer>LIGHTS</layer>
<!--  From a Shape file -->
<url>../../data/Terrain/Massachusetts/Features/US5MA10M/LIGHTS.shp</url>
  </features>
  ...
  <styles>
    <style type="text/css">
      NavigationLights {
        marker: "COLOUR=" + [COLOUR] + ":SIGSEQ=" + [SIGSEQ] + ":EXCLIT="
  + [EXCLIT];
        marker-scale:        1;
        marker-placement: vertex;
        altitude-clamping:   terrain;
        altitude-resolution: 0.00008583068847656250;
        altitude-offset:     0;
      }
    </style>
  </styles>
</model>
```

Since we are not building model definitions, we do not need to map enumerated values to strings. However, we do map S-57 light colors to the colors that we support. This mapping is in *light_attr_map.csv*.

VR-Vantage associates light controllers with other streamed features based on the S-57 or shape file locations of the lights and features, as follows:

- If the COLOUR attribute is provided, the attached light controller sets the light color using the light color switch.
- If the SIGSEQ attribute is provided, the light is turned on and off according to the specified sequence.
- If the EXCLIT attribute is provided, the light controller enables or disables the light based on daylight and fog conditions.

The light color switch and light switch values can be changed by specifying different ones through the API, specifically using:

```
de.scene().terrain().navigationLightManager().
     setLightColorSwitchNumber()
```

and

```
de.scene().terrain().navigationLightManager().setLightSwitchNumber()
```

26.11.5. Configuring Seasonal Buoys and Beacons

Seasonal buoys and beacons are supported. Include the PERSTA/PEREND attributes in the earth file as part of the marker field, and buoys and beacons will only appear during the correct season. Example:

```
marker: "Buoys[COLOUR][COLPAT][BOYSHP][DESIGNATOR]"
  + ":BOYSHP=" + [BOYSHP]
  + ":COLOUR=" + [COLOUR]
  + ":COLPAT=" + [COLPAT]
  + ":PERSTA=" + [PERSTA]
  + ":PEREND=" + [PEREND]
  + ":OBJNAM=" + [OBJNAM] ;
```

26.11.6. Streaming Daymarks

VR-Vantage supports streamed daymarks for beacons. A limited number of daymark models and textures are provided as examples. *Massachusetts S-57 (online).earth* has a `model` statement to stream daymarks from the Boston Harbor S-57 data. As with buoys and beacons, the `model` entry specifies a style name (Daymarks), which corresponds to a layer specified in *daymark_layer_list.csv*. OSGEarth models that use the feature_custom driver and have a style name that is in the layer list file, are processed as daymarks. S-57 attribute values can be mapped using the *daymark_attr_map.csv* file.

The beacon models provided with VR-Vantage contain multiple daymark placard shapes (Diamond, Square, and Triangle), which are selected based on the TOPSHP attribute. The TOPSHP attribute is mapped to '0', '1', or '2', which are the switch values for the diamond, square, or triangle shaped marks. Other attribute values, including TOPSHP, COLOUR, COLPAT, and CATSPM are used to look up UV coordinates in the model's texture atlas using a uv-texture coordinate mapping file similar to the one described for buoys. The mappings are in *./data/Objects/Beacons/daymark.dtxc*. These daymark shapes, plus textures provided, represent all the daymarks attached to beacons in Boston Harbor. The model definition for daymarks includes a parameter that specifies a string key to use in the *.dtxc* file ("daymarkTextureKey").

26.11.7. Streaming Topmarks

Topmarks can be streamed in and attached to buoys and beacons similarly to daymarks. The shape is determined by the TOPSHP attribute. The attribute is mapped to a switch value using the *topmark_attr_map.csv* file. Textures for the topmarks can be selected from a texture atlas similarly to buoys and daymarks. The model definition for a buoy or beacon that needs to map attributes to uv-texture coordinates using that model's *.dtxc* file will have a string key (`topmarkTextureKey`) that identifies the mapping entries in the file that pertain to the topmark texture atlas.

26.11.8. Generating Signal Sequences

The S-57 attribute SIGSEQ (signal sequence) specifies the flashing sequence that a navigational aid light on a buoy or beacon displays. These sequences (along with color) are used to distinguish among different buoys and beacons at night. However, this is not a required S-57 attribute. If the SIGSEQ attribute is not present, VR-Vantage tries to generate a signal sequence using the LITCHR (light characteristic), SIGPER (signal period) and SIGGRP (signal group) attributes for the most commonly seen sequences.

As part of the class responsible for generating these sequences, the VR-Vantage API includes a virtual function for each sequence type that is not implemented, so that a developer can implement it. The assumption is that not many of the other sequence types will be used, but it is data-specific. Once we know what sequences are needed for the S-57 data in use, it should be relatively straightforward to implement those additional sequence generation methods.

The class used to generate signal sequences is *DtLightSignalSequenceGenerator*, in the vrvCore library. *DtLightSignalSequenceGenerator.h* provides documentation, including which methods are implemented and which are not. To extend this class, a creator is provided, *DtLightSignalSequenceGeneratorCreator*, an extended instance of which can be registered through a plug-in with the display engine.



27. Using MetaFlight Terrains

To display MetaFlight files in VR-Vantage applications, you must first process them. This chapter explains how to do so.

Loading MetaFlight Terrains 27-2
Building Efficient MetaFlight Terrains for VR-Vantage 27-2
Processing MetaFlight Files for Use in VR-Vantage 27-3
Splitting Datasets into Individual MetaFlight Files 27-3
Converting Virtual Texture Datasets to Tiled Textures 27-4
The mftTool 27-4
Using the mftTool to Process MetaFlight Files 27-6
Convert Geometry Grid Datasets with Tiled Textures 27-7
Convert Source Data into MEDF Format 27-7
Create an MTF File for the MetaFlight Terrain 27-8

27.1. Loading MetaFlight Terrains

When you load a MetaFlight file, VR-Vantage is actually loading many individual OpenFlight files. If file caching is enabled, VR-Vantage caches the OpenFlight files to disk in a quick-loading, encrypted format so that the files can load more quickly the next time you load the terrain. The cached files are handled the same way that any other cached files are handled. For details, please see "Configuring File Caching," on page 29-2.



Geometry files referenced by a MetaFlight file must be in OpenFlight format, and not in Vega Prime's binary format (*.vsb*). MAK's terrain libraries do not support *.vsb* files.

27.2. Building Efficient MetaFlight Terrains for VR-Vantage

VR-Vantage can only load MetaFlight terrains built with certain tools. All MetaFlight files must be preprocessed by VR-Vantage tools before they can be loaded. The following are recommendations for constructing MetaFlight terrains that can be processed and loaded into VR-Vantage:

- The MetaFlight terrain can contain one or more Geometry Grid Datasets (GGDS).
- The MetaFlight terrain can contain one or more Virtual Texture Datasets.
- The MetaFlight terrain can contain one or more Cultural Feature Datasets.
- The MetaFlight terrain can be built in the following coordinate systems: Geodetic, Geocentric, UTM, or Flat Earth.
- For the best conversion of Virtual Textures into a tiled texture GGDS, match every Virtual Texture layer with a geometry layer.
- For optimal performance the GGDS should be set up in a Pyramid structure (Level 1: 1x1, Level 2: 2x2, Level 3: 4x4, and so on).
- For optimal performance levels 1-4 should be made of FILE LOD GEOM tiles only. Do not put any children LOD GEOM tiles in these levels.
- For optimal performance the terrain should be balanced so that any single FILE LOD GEOM is not too large and can be paged in real-time (based on machine speed and video card). In other words, keep any single file small.
- For optimal performance any single FILE LOD GEOM should not contain too many LOD GEOM nodes as children.
- Geometry Grid Tile size (in meters) should match Virtual Texture Tile coverage (in meters.)
- For optimal performance limit the number of external references. This will reduce cull times. (For example, instead of having 10 individual buildings, externally reference one city block with all 10 buildings.)

27.3. Processing MetaFlight Files for Use in VR-Vantage

Before you can load a MetaFlight file in a VR-Vantage application, you must process it as described in this chapter. If you try to load an unprocessed MetaFlight file, VR-Vantage displays an error message. If you run the MAK processing tools on a MetaFlight database that does not conform, an error message is sent to the console window.

To process a MetaFlight file, you must:

1. Convert source data into MEDF and MEIF format.
2. Split each dataset into its own MetaFlight file.
3. Convert geometry from using virtual texture datasets into using tiled textures. (If the terrain does not use virtual textures, you can skip this step.)
4. Use the VR-Vantage terrain composition process to combine the MetaFlight terrain files and save them in MTF format.



Geometry files referenced by a MetaFlight file must be in OpenFlight format, and not in Vega Prime's binary format (*.vsb*). MAK's terrain libraries do not support VSB files.

27.4. Splitting Datasets into Individual MetaFlight Files

MAK tools work on one Geometry Grid Dataset (GGDS) at a time. Some MetaFlight files contain multiple GGDSs, so each one has to be moved into its own file. It is your responsibility to understand the syntax of the MetaFlight file and to know whether or not you need to split up datasets.

To split datasets:

1. Create a copy of the original file for each dataset. Give each a name that identifies the dataset it contains.
2. Open each copy of the original file in a text editor.
3. In each file, delete all of the <GeometryGridDataset> XML element blocks except the one for the dataset you want to have in this file.

When you are finished, each MetaFlight file will have only one GGDS and possibly one or more Virtual Texture Datasets (VTDS).

27.5. Converting Virtual Texture Datasets to Tiled Textures

The display engine does not handle virtual textures at runtime. Therefore you must convert the virtual texture datasets into tiled textures.

27.5.1. The mftTool

The mftTool is a command-line tool for converting virtual texture datasets into tiled textures so that they can be loaded by VR-Vantage. It creates the highest resolution textures possible from the virtual texture dataset and binds them to individual tiles.

The syntax for mftTool is as follows:

```
mftTool -f filename [-c index -w index -l index -m string
   -r path -x size -s directory -t filename -A -g -e -u
   -p threads --processEmats --extendLevelZero distance -v
   -h --]
```

Table 27-1 describes the command-line options.

Table 27-1: mftTool command-line options

Option	Description
(-- | --ignore_rest)	Ignore all arguments after this one.
(-A | --regenerateAll)	Forces the regeneration of all tiled textures. This can be much slower and is needed less frequently if the virtual texture dataset does not change.
(-c | --col) *index*	The column to convert, using a zero-based index.
(-e | --encryptImages)	The tiled textures are generated in the MEIF file format. If not specified the file format will be DDS.
--extendLevelZero *distance*	Specifies the page-in range (in meters) of grid level zero in the MEDF and MetaFlight files that the mftTool creates. Use this option if the terrain disappears when you zoom out, particularly in Plan View mode. Increasing the range causes the terrain to page in at a greater distance.
(-f | --file) *filename*	The name of the MetaFlight file that contains the GGDS.

Table 27-1: mftTool command-line options

Option	Description
(-g | --generateFiles)	Generate tiled textures from the virtual texture and generate tiles in MEDF format from the FLT tiles that use the new textures. An MFT file is published. The output name will be equal to the MetaFlight file specified with the -f option with the string _medf appended to the end before the extension. For example, converting a file named *yourTerrain.mft* creates a new MetaFlight file named *yourTerrain_medf.mft* in the same directory as *yourTerrain.mft*. If the generated tiled texture already exists, the file is not regenerated. The -u and -g arguments are mutually exclusive.
(-h | --help)	Displays usage information.
(-l | --level) *index*	The level to convert, using a zero-based index.
(-m | --missing) *string*	The missing texture placeholder.
(-p | --threads) *threads*	Specifies the number of threads to run. Specifying zero results in the number of threads equaling the number of logical processors the host machine has.
--processEmats	Find materially classified (emat) versions of the virtual textures and generate new emat textures.
(-r | --searchDir) *directory*	Adds a directory to the search path when resolving external references. During processing any external references not found are removed from the generated files.
(-s | --subDir) *directory*	The path to the substitution texture specified by the VTDS. The arguments -s and -t are mutually exclusive.
(-t | --vtFile) *file*	The name of the MetaFlight file that contains the VTDS. The arguments -s and -t are mutually exclusive.
(-u | --publishMftOnly)	Publish a new MetaFlight file that can be loaded by VR-Vantage applications. The output name will be equal to the MetaFlight file specified with the -f option with the string _medf appended to the end before the extension. Converting a file named *yourTerrain.mft* creates a new MetaFlight file named *yourTerrain_medf.mft* in the same directory as *yourTerrain.mft*. The -u and -g arguments are mutually exclusive.
(-v | --version)	Display version information and exit.

Table 27-1: mftTool command-line options

Option	Description
(-w | --row) *index*	The row to convert, using a zero-based index.
(-x | --textureSize) *size*	Specifies the maximum texture size to generate in pixels per dimension. Larger textures use more memory and are slower to page in, but allow for greater virtual texture resolution.

If neither the generate (-g) or the publish only (-u) argument is specified, mftTool prints out information regarding the MetaFlight terrain.

27.5.2. Using the mftTool to Process MetaFlight Files

To process a terrain with a single MetaFlight terrain file that has one GGDS and one VTDS, run the mftTool as follows:

```
mftTool -f C:\yourTerrain\yourTerrain.mft
   -s C:\yourTerrain\ds200-vt-otw-virtual-texture -g -A -e
```

The following example shows how to process a terrain with two MetaFlight terrain files, one with the GGDS and one with the VTDS. In this case, the tiles also contain external references to other OpenFlight models.

```
mftTool
   -f C:\yourTerrain\ds100-flight-terrain\
   ds100-flight.mft
   -t C:\yourTerrain\ds200-vt-otw-virtual-texture\
   ds200-vt.mft-r C:\yourTerrain\textures
   -r C:\yourTerrain\objects
   -m C:\yourTerrain\MissingTexture.png -x 4096 -g
```

This creates a new MetaFlight file that references the new geometry grid dataset with virtual textures bound to it.

27.5.3. Convert Geometry Grid Datasets with Tiled Textures

The mftTool only needs to publish the MFT when the GGDS uses tiled textures. Cultural feature datasets are often built this way. These can be converted using the medfTool like all other files. The command syntax for the medfTool is described in "Compressing Model Files," on page 35-28.

To only publish a new MetaFlight for the cultural dataset, do the following:

```
mftTool -f C:\yourTerrain\flt\ds7_culture_db5_build-
   ing_area_new-hier_cult\ds7_culture_db5_building.mft -u
   -p 0
```

To convert a MetaFlight with tiled textures into a compressed format, use the medfTool, as follows:

```
medfTool -s C:\yourTerrain\flt\textures
   -s C:\yourTerrain\flt\objects
   --directory C:\yourTerrain\flt\
   ds7_culture_db5_building_area_new-hier_cult -x flt
   -m rgb -m rgba -m jpg -m bmp -p 0 -z 1
```

! The medfTool should never be run on a directory that contains GGDS tiles that use Virtual Textures.

27.6. Convert Source Data into MEDF Format

Many file formats, including OpenFlight files, are often slow to load, so it is best to convert them to the fast loading MEDF (MAK Encrypted Data Format) format. Do this for terrains that do not have virtual textures (it is done automatically when creating geometry grid datasets from virtual texture datasets) and on cultural feature datasets. This should be done on all data files being loaded, but is required for all external references used by MetaFlight terrains.

! You must convert the files to MEDF and MEIF before you run the mftTool.

To convert external references in separate directories:

```
medfTool -s C:\yourTerrain\flt\textures --directory
   C:\yourTerrain\flt\objects -x flt -m rgb -m rgba -m jpg
   -m bmp -p 0 -z 1
```

To convert external reference textures that are in the same directory:

```
medfTool --directory C:\yourTerrain\flt\textures -m rgb
   -m rgba -m jpg -m bmp -p 0 -z 1
```

27.7. Create an MTF File for the MetaFlight Terrain

Once you have processed all of the MetaFlight files, create an MTF file for the terrain.



Any missing models or white objects usually indicate missing external references. Fixing any missing search paths passed to the tools should resolve most problems.

To create an MTF file for the MetaFlight terrain:

1. Start any VR-Vantage application.
2. Add each MetaFlight terrain as a terrain patch.
3. Save the terrain as an MFT terrain.

28. Configuring CDB Terrain Databases

This chapter explains how to configure earth files to load CDB terrain databases.

> The VC 15 version of VR-Vantage does not support CDB.

Loading CDB Databases 28-2
Configuring CDB earth Files 28-2
- Configuring the Elevation Dataset (001) 28-3
- Configuring the Imagery Dataset (004) 28-5
- Configuring the GeoSpecific (GS) Dataset (100) 28-6
- Configuring the GeoTypical (GT) Dataset (101) 28-13

CDB Configuration Files 28-17

28.1. Loading CDB Databases

VR-Vantage can display multiple CDB datasets (elevation, imagery, geotypical points). MAK products load the datasets using an earth file. The earth file entries point to the CDB root path of the database and specify the CDB driver and the CDB dataset codes.

Because MAK uses osgEarth to load CDB databases, you have the same flexibility with CDB databases that you do with other database types. For example, you can:

- Use streaming data (from VR-TheWorld Server), or other source imagery to build a skirt around a CDB inset.
- Place an OpenFlight inset into a CDB database.
- Add lights, buildings, forests, grass, and other clutter on top of an existing CDB database.

VR-Vantage includes an example CDB terrain – *Test-CDB.mtf*. This terrain demonstrates the ability to load data in CDB format. For details about the CDB API, please see https://portal.presagis.com/support/solutions/articles/19000014917-cdb-api-

28.2. Configuring CDB earth Files

This section explains how to configure your CDB earth file to load the different datasets you need. You may need to configure the following datasets:

- Elevation
- Imagery
- Geospecific
- Geotypical
- Extended imagery.

28.2.1. Configuring the Elevation Dataset (001)

To load an elevation dataset, use the `elevation` element, as in the following example:

```
<elevation name="cdb_elevation"
              driver="cdbraster"
              cdb_dataset="1"
              cdb_component_selector1="1"
              cdb_component_selector2="1"
              cdb_cache_file="25"
              cacheid="camppend1-cdb-elevation" >
  cdb_root>../../../../../../../Presagis/CDB/
    CDB_Camp_Pendleton_3_0</cdb_root>
</elevation>
```

It tells osgEarth to use the CDB raster driver.

Table 28-1 describes the attributes for the `elevation` element.

Table 28-1: Attributes for elevation and imagery datasets

Attribute Name	Description
Name	Specifies a name that identifies the dataset you are loading in osgEarth.
Driver	Specifies the driver to use for this dataset (must be `cdbraster`).
cdb_dataset	(Optional) Specifies the CDB dataset to read. For elevation always use "1". Default: 1.
cdb_component_selector1	(Optional) Specifies the first CDB component selector to use. Component selectors are part of the sub-dataset naming convention. For a list of selectors, please see Table 28-2. Default: 1.
cdb_component_selector2	(Optional) Specifies the second CDB component selector to use. Default: 1.
cdb_cache_file	(Optional) Specifies how many CDB tiles to keep in memory while building an OSG tile. Default: 20.
cdb_max_lod	(Optional) Specifies the maximum LOD of this dataset. Set this value if you have a CDB database that is larger than 500 geocells. For example, if you have the CDB World Wide database, set the value to 0. This will restrain the CDB Driver to pre-parse all of the LOD folder. If the value is not set (default) it reads the CDB to find the maximum LOD of the elevation dataset.
Cacheid	Specifies where to save the cache files, if caching is enabled.
cdb_root	Specifies the top folder of the CDB database.

Table 28-2 lists the `elevation` sub-dataset codes.

Table 28-2: `elevation` sub-dataset codes

Dataset Code (Dxxx)	Component Selector 1 (Sxxx)	Component Selector 2 (Txxx)	Description
1	1	1	Primary Elevation.
1	1	2	Primary Elevation Control (deprecated in CDB 3.2). (Not supported.)
1	1	3	Primary Alternate Elevation. (Not supported.)
1	2-99	1	Subordinate Elevation (deprecated in CDB 3.2). (Not supported.)
1	2-99	2	Subordinate Elevation Control (deprecated in CDB 3.2). (Not supported.)
1	100	1	Subordinate Bathymetry.
1	100	2	Subordinate Alternate Bathymetry. (Not supported.)
1	101	1	Subordinate Tide Elevation. (Not supported.)

Configuring the Elevation Sub-dataset (Bathymetry)

Adding the bathymetry allows you to add dynamic ocean in Vantage. The following example shows how to add bathymetry:

```
<elevation name="cdb_bathymetry"
             driver="cdbraster"
             cdb_dataset="1"
             cdb_component_selector1="100"
             cdb_component_selector2="1"
             cdb_cache_file="10"
             cacheid="camppend1-cdb-elevation" >
  <cdb_root>../../../../../../../Presagis/CDB/
    CDB_Camp_Pendleton_3_0</cdb_root>
  <offset>true</offset>
</elevation>
```

28.2.2. Configuring the Imagery Dataset (004)

The `image` element tells osgEarth to use the CDB driver to load the imagery dataset, as shown in the following example:

```
<image name="cdb_imagery"
         driver="cdbraster"
         cdb_dataset="4"
         cdb_component_selector1="1"
         cdb_component_selector2="1"
         cdb_cache_file="25"
         cacheid="camppend1-cdb-imagery" >
  <cdb_root>../../../../../../../Presagis/CDB/
    CDB_Camp_Pendleton_3_0</cdb_root>
</image>
```

Table 28-1 describes the attributes for the `image` element. They are the same as for the elevation attribute, except that `cdb_dataset` must be 4.

Table 28-3 lists the codes for the imagery sub-dataset.

Table 28-3: Imagery sub-dataset codes

Dataset Code (Dxxx)	Component Selector 1 (Sxxx)	Compo-nent Selec-tor 2 (Txxx)	Description
4	1	1	Yearly Visual Spectrum Terrain Imagery (VSTI) representation.
4	2	1	Alternate Spring representation (depre-cated in CDB 3.2).
4	2	2	Alternate Summer representation (deprecated in CDB 3.2).
4	2	3	Alternate Fall representation (deprecated in CDB 3.2).
4	2	4	Alternate Winter representation (depre-cated in CDB 3.2).
4	3	1-12	Alternate Monthly representation.
4	4	1-4	Quarterly VSTI representation.
4	5	1	Subordinate VSTLM. (Corresponds to the terrain light maps draped (orthographically) over the terrain skin derived from the Primary Terrain Eleva-tion Dataset. It may be single-channel monochrome or 3-channel color image.) Please see "Configuring the Imagery Lightmap Sub-dataset (004-005)," on page 28-6 for more information.

Configuring the Imagery Lightmap Sub-dataset (004-005)

The following example shows how to add a lightmap. This code is different from the primary cdb imagery dataset.

```
<lightmap>
  <image name="cdb_imagery_lightmap"
           driver="cdbraster"
           cdb_dataset="4"
           cdb_component_selector1="5"
           cdb_component_selector2="1"
           cacheid="camppend3-cdb-lightmap"
           cache_enabled="false">
  <cache_policy usage="none"/>
  <cdb_root>../../../../../../../Presagis/CDB/
    CDB_Camp_Pendleton_3_0</cdb_root>
   </image>
</lightmap>
```

28.2.3. Configuring the GeoSpecific (GS) Dataset (100)

Geospecific data replicates the real world data. The following example shows how to configure a geospecific dataset:

```
<model driver="cdb"
         name="GS Man Made"
         cdb_dataset="100"
         cdb_component_selector1="1"
         cdb_component_selector2="1"
         cacheid="camppend3-cdb-gsmodels">
  <cdb_root>../../../../../../../Presagis/CDB/
    CDB_Camp_Pendleton_3_0</cdb_root>
  <sigsize_file>../userData/servers/
    Sample_SignificantSize_SwitchDistance.xml</sigsize_file>
  <pagingdistance_file>../userData/servers/
    Sample_Model_PagingDistance.xml</pagingdistance_file>
  <lightdefaultvalues_file>../appData/cdb/
    Lights_DefaultValues_3_0.xml</lightdefaultvalues_file>
</model>
```

Table 28-4 describes the attributes for the geospecific dataset.

Table 28-4: Attributes for geospecific dataset

Attribute Name	Description
Name	Specifies a name that identifies the dataset you are loading in osgEarth.
Driver	Specifies the driver to use for this dataset (must be `cdb`).
cdb_dataset	Specifies the CDB dataset to read. For geospecific always use “100”.
cdb_component_selector1	(Optional) Specifies the first CDB component selector to use. Component selectors are part of the sub-dataset naming convention. For a list of selectors, please see Table 28-5. Default: 1.

Table 28-4: Attributes for geospecific dataset

Attribute Name	Description
cdb_component_selector2	(Optional) Specifies the second CDB component selector to use. Default: 1.
Cacheid	Specifies where to save the cache files if caching is enabled. If Accelerated File Loading is enabled, the CDB geospecific models are converted to MEDF and the textures are converted to MEIF on the first load. They will be used the next time the CDB database is loaded.
cdb_root	Specifies the top folder of the CDB database.
elevation_cache	(Optional) Creates an elevation binary cache file for clamp models. Default: true.
texture_mipmap_offset	(Optional) Specifies an offset to load lower mipmap textures from all geotypical textures. (Geospecific models can have geotypical textures.) Allowed values: 0-9. 0 means no offset. 1-9 means load a lower numbered LOD by the amount of the offset. For example if a model is using a texture LOD 9, setting the offset to 2 causes it to load LOD 7.
debug	(Optional) Enables (true) or disables (false) debug information on CDB tiles.
debug_modelcolor	(Optional) If set to true (debug also need to set to true), color the models with the same color as the tiles.
range_factor	(Optional) Multiplier of the tile paging values found in *Model_PagingDistance.xml*.
sigsize_file	(Optional) Replace the default *./appData/cdb/SignificantSize_SwitchDistance.xml*.
pagingdistance_file	(Optional) Replace the default *./appData/cdb/Model_PagingDistance.xml*.
lightdefaultvalues_file	(Optional) Replace the default light configuration file (*Lights_DefaultValues_3_2.xml*) with another one. Use this parameter only for Component Selector 1 (4 and 5, that is, Airport Light and Environment Light).

Table 28-5 lists geospecific sub-dataset codes.

Table 28-5: Geospecific sub-dataset codes

Dataset Code (Dxxx)	Component Selector 1 (Sxxx)	Component Selector 2 (Txxx)	Description
100	1	1	Man-made point features.
100	1	3	Man-made lineal features (not supported).
100	1	5	Man-made areal features (not supported).
100	2	1	Natural point features.
100	2	2	Natural lineal features (not supported).
100	2	5	Natural areal features (not supported).
100	3	1	Trees point features.
100	4	1	Airport light point features.
100	4	3	Airport lineal features (not supported).
100	4	5	Airport areal features (not supported).
100	5	1	Environmental light point features.

Configuring the Geospecific Airport Lightpoint Sub-dataset (100-004)

The following example code shows how to configure a geospecific airport light-point dataset:

```
<model driver="cdb"
         name="GS Airport Light"
         cdb_dataset="100"
         cdb_component_selector1="4"
         cdb_component_selector2="1"
         cacheid="camppend3-cdb-gsairportlights">

  <cdb_root>../../../../../../../Presagis/CDB/
    CDB_Camp_Pendleton_3_0</cdb_root>
</model>
```

You can test the proper functioning of the lightpoints by turning them on and off using debugging tools:

To turn on/off airport light points using the debug window:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 24-14).
3. Select the Show Developer Debug Overlay check box. The Debug window opens (Figure 28-1).



Figure 28-1. Debug window

4. Expand Light Point Settings.
5. Toggle the slider of the runway you would like to change. For example, in the Camp Pendleton terrain, if you want to close the runway KNFG: 21, set the slider to 0.

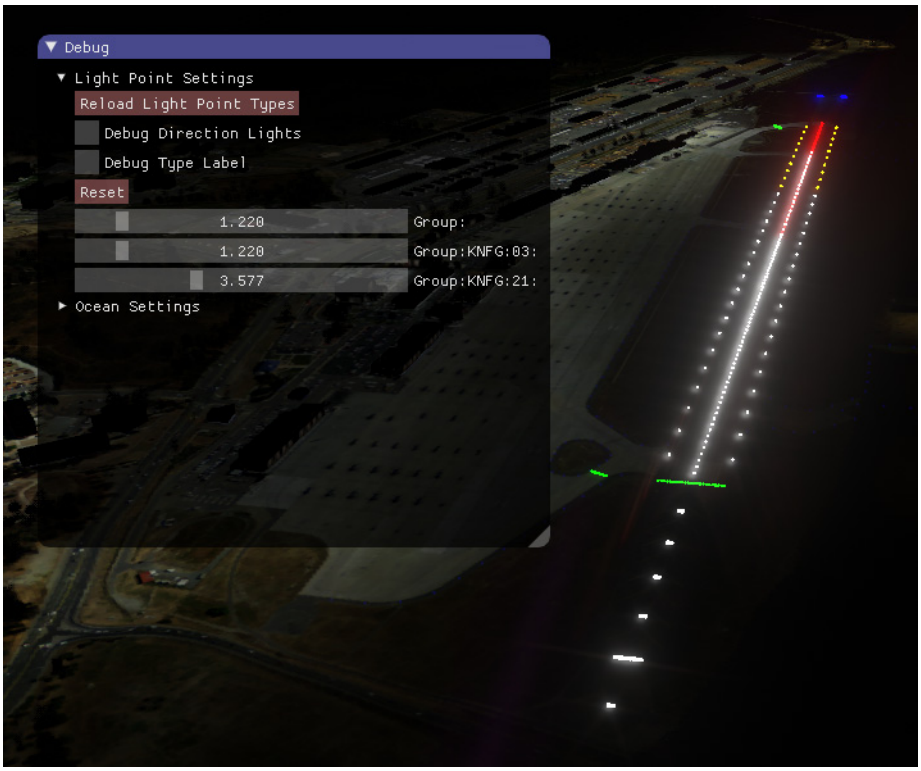


Figure 28-2. Camp Pendleton airport with debug window

Configuring the Geospecific Environmental Lightpoint Sub-dataset (100-005)

The following example shows how to configure geospecific environmental light-points:

```
<model driver="cdb"
           name="GS Environment Light"
           cdb_dataset="100"
           cdb_component_selector1="5"
           cdb_component_selector2="1"
           cacheid="camppend3-cdb-gsenvlights">
  <cdb_root>../../../../../../../Presagis/CDB/
    CDB_Camp_Pendleton_3_0</cdb_root>
</model>
```

Configuring Light Point Properties

You can change the behaviors of a type of light point. Light types are configured in *./appData/importConfig/Lights_DefaultValues_3_2.csv*. To change a light point type, you can edit the file in a text editor or spreadsheet. Each light type has the following attributes:

- Code: Light type code (do not edit).
- Description: Light type description (do not edit).
- Intensity: From 0.0 to 1.0.
- Red-Green-Blue: RGB value of the color (from 0.0 - 1.0 for each channel).
- Directionality: Directional, Omnidirectional. Bidirectional.
- Horizontal Lobe: When a light type is not native to the CDB specification, the Horizontal field represents the light point's half-intensity horizontal lobe width for the client-device (range from 0.0 to 360.0). When the light type is native to the CDB specification, the Horizontal field is used as a floating-point modifier that multiplies the horizontal lobe width calculated by the client-device. It applies only to Directional and Bidirectional light types. Default: 1.0.
- Vertical Lobe: When a light type is not native to the CDB specification, the Vertical field represents the light point's half-intensity vertical lobe width for the client-device (range from 0.0 to 360.0). When the light type is native to the CDB specification, the Vertical field is used as a floating-point modifier that multiplies the vertical lobe width calculated by the client-device. It applies only to Directional and Bidirectional light types. Default: 1.0.
- Residual Intensity: When a light type is not native to the CDB specification, Residual_Intensity represents the residual intensity of the light. Residual intensity is the intensity of the light (range normalized from 0.0 to 1.0) outside of the lobe defined by Lobe_Width:Horizontal and Lobe_Width:Vertical fields. When the light entry is native to the CDB specification, Residual_Intensity is used as a floating-point modifier that multiplies the residual intensity calculated by the client-device. It applies only to Directional and Bidirectional light types. Default: 1.0.
- Frequency: A floating-point value greater than or equal to 0.0 that sets the blink or rotating frequency of the light in Hertz (cycles per second). A value of 0.0 disables all blinking and rotating properties. Default: 0.0.
- Duty Cycle: A floating-point value ranging from 0.0 to 1.0 that sets the duty cycle of the light. Duty cycle is defined as the percentage of time the light is turned on over a complete cycle. A value of 0.0 permanently turns the light off. A value of 1.0 turns it on. The value is ignored if Frequency = 0.0. Default: 0.5.
- Min Pixel Size: Minimum number of pixels that represent the light point when you are far from it. (This attribute is specific to VR-Vantage applications.)
- Max Pixel Size: Maximum number of pixels that represent the light point when you are close from it. (This attribute is specific to VR-Vantage applications.)
- Max Visible Distance: Distance in meters when the light point will disappear. (This attribute is specific to VR-Vantage applications.)

To change the behaviors of a light point type, you need to know what light type is being used.

To identify the type of light point:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 24-14).
3. Select the Show Developer Debug Overlay check box. The Debug window opens (Figure 28-1).
4. Expand Light Point Settings.
5. Select the Debug Type Label check box.
6. Zoom in on the light point that you want to edit. It is labeled with a number (Figure 28-3).



Figure 28-3. Lightpoint type labels

28.2.4. Configuring the GeoTypical (GT) Dataset (101)

Geotypical data is data that is typical for a particular type of object but does not necessarily match the object in the real world.

The following example shows how to add a geotypical dataset to an earth file:

```
<model driver="cdb"
         name="GT Trees"
         cdb_dataset="101"
         cdb_component_selector1="2"
         cdb_component_selector2="1"
         cacheid="camppend3-cdb-gtmodels2">
  <cdb_root>../../../../../../../Presagis/CDB/
    CDB_Camp_Pendleton_3_0</cdb_root>
  <sigsize_file>../userData/servers/
    Sample_SignificantSize_SwitchDistance.xml</sigsize_file>
  <pagingdistance_file>../userData/servers/
    Sample_Model_PagingDistance.xml</pagingdistance_file>
  <speedtree_catalog>Camp_Pendleton_3_0_SpeedTrees.xml
    </speedtree_catalog
  <features name="Vegetation"/>
  <styles>
    <style type="text/css">
    Vegetation {
      marker-scale:        1;
      marker-placement: vertex;
      altitude-clamping:   terrain;
      altitude-resolution: 0.00008583068847656250;
      altitude-offset:     0;
    }</style>
  </styles>
</model>
```

Table 28-6 describes the attributes for a geotypical dataset.

Table 28-6: Attributes for geotypical dataset

Attribute Name	Description
Name	Specifies a name that identifies the dataset you are loading in osgEarth.
Driver	Specifies the driver to use for this dataset (must be `cdb`).
cdb_dataset	(Optional) Specifies the CDB dataset to read. For geotypical data always use “101”.
cdb_component_selector1	(Optional) Specifies the first CDB component selector to use. Component selectors are part of the sub-dataset naming convention. For a list of selectors, please see Table 28-7. Default: 1.
cdb_component_selector2	(Optional) Specifies the second CDB component selector to use. Default: 1.

Table 28-6: Attributes for geotypical dataset

Attribute Name	Description
Cacheid	Specifies where to save the cache files if caching is enabled. If Accelerated File Loading is enabled, the CDB geotypical models are converted to medf and the textures are converted to meif on the first load. They will be used the next time the CDB database is loaded.
cdb_root	Specifies the top folder of the CDB database.
elevation_cache	(Optional) Creates an elevation binary cache file for clamp models. Default: true.
texture_mipmap_offset	(Optional) Specifies an offset to load lower mipmap textures from all geotypical textures.Allowed values: 0-9. 0 means no offset. 1-9 means load a lower numbered LOD by the amount of the offset. For example if a model is using a texture LOD 9, setting the offset to 2 causes it to load LOD 7.
cache_policy	(Optional) Determine whether or not to cache models.
debug	(Optional) Enables debug information on CDB tiles
debug_modelcolor	(Optional) If set to true, color the models with the same color as the tiles. The debug attribute must also be set to true.
range_factor	(Optional) Multiplier of the tile paging values found in *Model_PagingDistance.xml*.
sigsize_file	(Optional) Replace *./appData/cdb/SignificantSize_SwitchDistance.xml* with another version.
pagingdistance_file	(Optional) Replace the *./appData/cdb/default Model_PagingDistance.xml* with another version.
rendering_method	Valid values are OSG_RENDERING or INDIRECT_RENDERING. If the attribute is not included, it defaults to INDIRECT_RENDERING. Using the indirect rendering value sets this model layer to use the optimized indirect rendering system, which allows the models to be rendered much faster than the OSG rendering method.
speedtree_catalog	(Optional) Specifies a file that lists the CDB geotypical SpeedTree replacement. (For more information, please see "CDB Configuration Files," on page 28-17.) The <features> and <styles> elements must be present as in the example for SpeedTrees to work with CDB.

Table 28-7 lists the codes for the geotypical sub-dataset.

Table 28-7: Codes for geotypical sub-datasets

Dataset Code (Dxxx)	Component Selector 1 (Sxxx)	Component Selector 2 (Txxx)	Description
101	1	1	Man-made point features.
101	1	3	Man-made lineal features (not supported).
101	1	5	Man-made areal features (not supported).
101	2	1	Tree point features.
101	2	2	Tree lineal features (not supported).
101	2	5	Tree areal features (not supported).
101	3	1	Moving Model location features (not supported).

28.2.5. Configuring the Extended Imagery Dataset (900)

The following example shows how to add the extended imagery dataset to an earth file.



This dataset needs the CAE API pro license and the configuration file *.\appData\cdb\900_ExtImagery.xml*. These extensionS are not part of the CDB specification, but can permit vendors to extend the CDB to their needs.

```
<image name="cdb_cartography"
         driver="cdbraster"
         cdb_dataset="900"
         cdb_component_selector1="1"
         cdb_component_selector2="1"
         cdb_cache_file="25"
         cacheid="yemen-cdb-cartography"
  <cdb_root>../../../../../../../Presagis/CDB/CDB_Yemen_3_2</cdb_root>
</image>
```

Table 28-8 describes the attributes for configuring extended imagery.

Table 28-8: Extended imagery attributes

Attribute Name	Description
Name	Specifies a name that identifies the dataset you are loading in osgEarth.
Driver	Specifies the driver to use for this dataset (must be `cdbraster`).
cdb_dataset	(Optional) Specifies the CDB dataset to read. For imagery data always use "900". Default: 1.
cdb_component_selector1	(Optional) Specifies the first CDB component selector to use. Component selectors are part of the sub-dataset naming convention. For a list of selectors, please see Table 28-9. Default: 1.
cdb_component_selector2	(Optional) Specifies the second CDB component selector to use. Default: 1.
cdb_cache_file	Specifies how many CDB tiles to keep in memory while building an OSG tile. Default: 20.
Cacheid	Specifies where to save the cache files if caching is enabled.
cdb_root	Specifies the top folder of the CDB database.
cdb_max_lod	(Optional) Specifies the maximum LOD of this dataset. Set this value if you have a CDB database that is larger than 500 geocells. For example, if you have the CDB World Wide database you can set that value to 0. This will cause the CDB driver to pre-parse all of the LOD folder. If the value is not set (default) it reads the CDB to find the maximum LOD of the imagery dataset.

Table 28-9 lists the sub-dataset code for extended imagery.

Table 28-9: Extended imagery sub-dataset codes

Dataset Code (Dxxx)	Component Selector 1 (Sxxx)	Component Selector 2 (Txxx)	Description
900	1	1	Cartography imagery.

28.3. CDB Configuration Files

VR-Vantage uses the following files to configure CDB terrain databases.

- *Datasets.xml*. This file lists the CDB 3.2 Datasets code and folder name. Do not edit this file until a new CDB specification is released or MAK creates an extended CDB dataset. This list is read by the MAK CDB API when the CDB DLLs are loaded.
- *Feature_Data_Dictionary.xml*. This file contains all the FACC, FSC code (Facc sub code) definitions from the CDB 3.2 specification. Do not edit this file.
- *SignificantSize_SwitchDistance.xml*. CDB uses the concept of Significant Size to switch between LODs. Since this concept is not supported currently by makOSG, the makOSG Openflight plug-in was modified to support FLT CDB models. To support the CDB model Significant Size, the plug-in was modified to convert SigSize into normal LOD switch in/out distance. The Switch distance is supported by makOSG. The plug-in contains default values that can be overwritten by this file. As a guideline, each CDB LOD in the file (also associated to a significant size) should have a different switch in distance greater than the next LOD (this will prevent abnormal behavior at runtime).

  Inside this file, switch distance can be defined by CDB datasets.

  You can override this file for each dataset by adding the following code to your earth file:

  ```
  <sigsize_file>.\yourSigSizeFile.xml</sigsize_file>
  ```

- *Model_PagingDistance.xml*. Configures the distance for the view point when each LOD tile is getting loaded. This can be configured by CDB datasets.

  This file can be overwritten in the earth file for each dataset by using:

  ```
  <pagingdistance_file>./AnotherModelPagingDistanceFile.xml</pagingdistance
    _file>
  ```

- *Light_DefaultValues_3_2.xml*. This file contains all the CDB 3.2 light point default values defined in *CDB Specification - Volume 2.pdf*, Appendix E. By default, VR-Vantage uses *./appData/importConfig/Light_DefaultValues_3_2.csv* to configure light properties for CDB databases. You can override it by adding the `<lightdefaultvalues_file>` element to your earth file and specifying *Light_DefaultValues_3_2.xml* (or *Light_DefaultValues_3_0.xml*) as the source of light properties.

  If it is specified, the Model_CDB plug-in reads this file at load time.

  These values complete the information found in your CDB database under *./Metadata/Lights.xml*. Do not edit this file unless an error is found.

  You can specify a light properties file for each light dataset by using:

  ```
  <lightdefaultvalues_file>./AnotherLightDefaultValuesFile.xml</lightdefaul
    tvalues_file>
  ```

For example, if you want to use the CDB 3.0 default values you could add this to the airport lights section:

```
<lightdefaultvalues_file>./appData/cdb/Light_DefaultValues_3_0.xml</light
  defaultvalues_file>
```

- *Light_DefaultValues_3_0.xml*. This file contains all the CDB 3.0 light point default values defined in the *CDB Specification - Volume 2.pdf*, Appendix E. Do not edit this file unless you find an error in it except for the following attributes:
  - `<Radius>12.0</Radius>`. Radius in meter of the osg light point.
  - `<MinPixelSize>2.0</MinPixelSize>`. Minimum pixel size of the osg light point node.
  - `<MaxPixelSize>4.0</MaxPixelSize>`. Maximum pixel size of the osg light point node.
  - `<MaxVisibleDistance>12000</MaxVisibleDistance>`. Maximum visible distance in meters.
- *Lights_xxx.xml*. This file shows how to do specific light tuning. Do not change the name of this file. The content must be updated with your device specific tuning and you must put the file in your CDB database's */Metadata* directory.
- *900_ExtImagery.xml*. This file is needed to extend the API to support the 900_ExtImagery dataset in the Yemen CDB database. (You must have a CAE API pro license to support this.)
- *XXX_SpeedTrees.xml*. This file is used to map speedTrees to the CDB geotypical model. For example, you can create a file called *Camp_Pendleton_3_0_SpeedTrees.xml* that will be referenced in the Geotypical tree section your earth file. (For details, please see "Configuring CDB earth Files," on page 28-2). The file will usually be in the same folder as the earth file.



29. Terrain Performance and Configuration

This chapter covers miscellaneous issues that affect terrain performance and functioning.

Configuring File Caching 29-2
- Specifying the Location to Cache Terrain Server Data 29-2
- Caching earth File Terrain Data 29-3
- Caching earth File Terrain Data Offline 29-4
- Enabling Texture Compression 29-5
- Clearing the File Cache 29-5

Displaying DDS Textures Correctly 29-6
- Flipping DDS Textures Globally 29-7

Preprocessing Paged Terrains 29-8

Configuring Paged and Streaming Terrains 29-8

Debugging Earth Files 29-9

Using Shader-based Effect Maps 29-10
- Composite Effects Maps 29-12
- Debugging Shaders 29-13
- Reloading Shaders 29-14

Displaying Terrain Intersection Lines 29-15

29.1. Configuring File Caching

When file caching is enabled, VR-Vantage stores information about the terrain and models in a format that enables it to be loaded quickly. This improves loading time the next time the cached objects are loaded. If caching is disabled, terrains and models must be loaded from their native formats, which may take longer.



When VR-Vantage processes a request to load a file, such as an external reference in a terrain, it checks the cache and the directory of the requested file for an MEDF version of the file. If it does not exist, it loads the file in the original format.

To enable or disable file caching:

1. Choose **Settings** → **Application**. The Application Settings dialog box opens.
2. Select the File Caching Settings page (Figure 29-1).

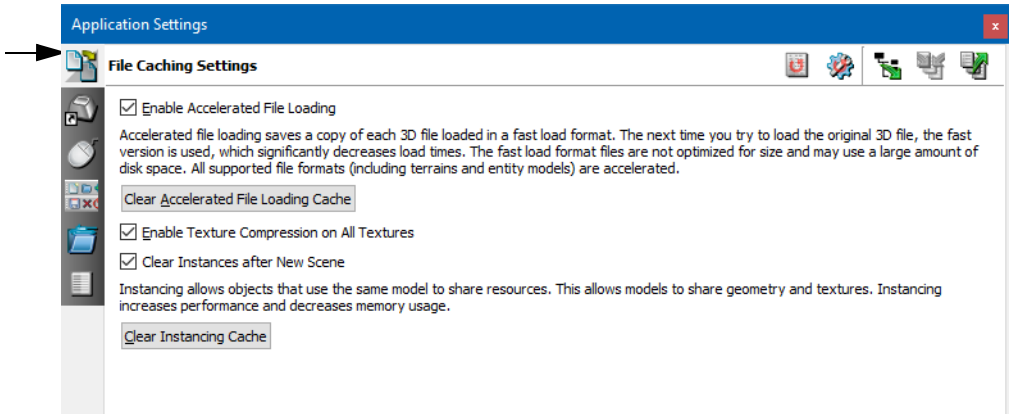


Figure 29-1. File Caching Settings page

3. Select or clear the Enable Accelerated File Loading check box.

29.1.1. Specifying the Location to Cache Terrain Server Data

When file caching is enabled, VR-Vantage caches data from terrain servers in the default locations. You can override this by specifying an alternative caching location using the OSGEARTH_CACHE_PATH environment variable. (The OSGEARTH_CACHE_PATH setting overrides the `cache` element in an earth file.)

29.1.2. Caching earth File Terrain Data

You can cache the terrain data in earth files to speed up loading terrain and to allow you to load terrains without connecting to a terrain server, either locally or over the internet. The osgearth_cache tool lets you cache imagery and elevation data offline. (For details, please see "Caching earth File Terrain Data Offline," on page 29-4.) You can also generate cached data through the GUI. The advantage of caching data from the GUI is that you can cache imagery, elevation, and feature data and can use the mouse to specify the area to cache rather than needing to calculate the latitude and longitude, as required by osgearth_-cache.



It may take a very long time to generate the cache for large areas.

To generate cached data:

1. Load a terrain that uses an earth file.
2. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
3. Select the Terrain Contents page.
4. Click the osgEarth Cache button. The osgEarth Cache dialog box opens (Figure 29-2).

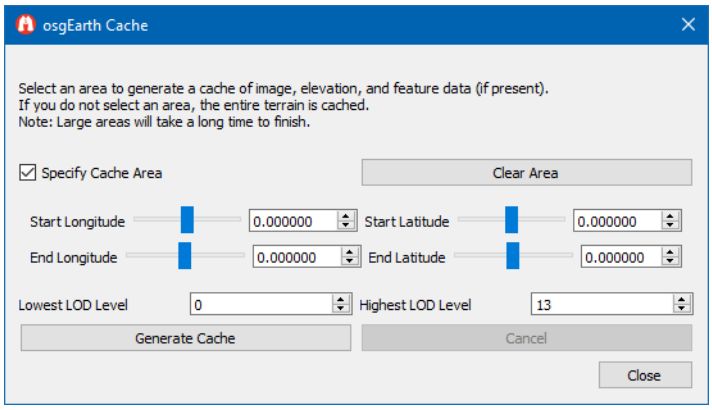


Figure 29-2. osgEarth Cache dialog box

5. If you want to generate cached data for the entire terrain, make sure that the Specify Cache Area check box is cleared.
6. If you want to generate cached data for a portion of the terrain:
   a. Select the Specify Cache Area check box.
   b. Specify the latitude and longitude of a rectangle that defines the area you want to cache. You can do this by adjusting the sliders or entering values for the four values or you can draw a rectangle on the terrain.

To draw a rectangle on the terrain, click the left mouse button and release it to specify the upper left corner of the rectangle. Drag the mouse to define the cache area. Left-click again to specify the lower right corner.

7. Optionally specify the lowest LOD level and the highest LOD level.
8. Click Generate Cache.

29.1.3. Caching earth File Terrain Data Offline

VR-Vantage includes a tool, *./bin64/osgearth_cache*, that you can use to cache terrain data offline. When you need to use the terrain, the data will have already been cached. osgearth_cache caches imagery and elevation. It does not cache feature data. The syntax is as follows:

```
osgearth_cache --list file.earth | --seed file.earth
   [--estimate --min-level level --max-level level
   --bounds xmin ymin xmax ymax* --index shapefile --mp
   --mt --concurrency --verbose]| --purge file.earth
```

where:

- `--list` *`file.earth`* lists information about the cache in an earth file. If the earth file uses template syntax to include other files, you must append *.template* to the name of the file, for example, *myfile.earth.template*. If the file name has spaces, enclose it in parentheses.
- `--seed` *`file.earth`* seeds the cache for an earth file. If the earth file uses template syntax to include other files, you must append *.template* to the name of the file, for example, *myfile.earth.template*. If the file name has spaces, enclose it in parentheses.
- `--estimate` prints out an estimation of the number of tiles, disk space, and time it will take to perform the seed operation.
- `--min-level` *`level`* specifies the lowest LOD level to seed. Default: 0.
- `--max-level` *`level`* specifies the highest LOD level to seed. Default: highest available.

! We strongly recommend that you specify `--max-level` and seed just the areas that you want.

- `--bounds` *`xmin ymin xmax ymax`* specifies the geospatial bounding box to see, in map coordinates. You can include multiple instances of this option. Default: entire map.
- `--index` *`shapefile`* tells osgearth_cache to use the feature extents in the specified shapefile to set the bounding boxes for seeding.
- `--mp` specifies use of multiprocessing to process the tiles. This is useful for GDAL sources because it avoids the global GDAL lock.
- `--mt` specifies use of multithreading to process the tiles.
- `--concurrency` specifies the number of threads or processes to use if `--mp` or `--mt` are used.
- `--verbose` displays the progress of the seed operation.

- `--purge` *`file.earth`* purges a layer cache in an earth file. This command is interactive.

The following example caches a file named *VR-TheWorldOnline.earth*:

```
osgearth_cache.exe --seed ../userData/servers/VR-TheWorld-
  Online.earth --max-level 2
```

The following example caches a portion of a file named *HawaiiLocalServer-mak.earth*:

```
osgearth_cache.exe --seed ..\userData\servers\Local World
  Ala Moana.earth --max-level 18 --bounds
  -162.0013879999999915 14.9985579999999992
  -149.9985580000000027 25.0013879999999986
```

29.1.4. Enabling Texture Compression

When texture compression is enabled, textures are converted internally to a compressed format (DXT). Compressing textures decreases memory use and increases rendering performance.



The textures provided with VR-Vantage are already compressed. This setting is useful if you are adding your own data.

To enable or disable texture compression:

1. Choose **Settings** → **Application**. The Application Settings dialog box opens.
2. Select the File Caching Settings page (Figure 29-1).
3. Select or clear the Enable Texture Compression On All Textures check box.

29.1.5. Clearing the File Cache

You can clear the file cache if you need to free disk space.

The cache is normally configured to be in *./appData/cache*. You can delete files from this directory if you want to. The most likely reason to do this is if you load a terrain and realize that it should have had its DDS textures flipped, but has now been saved with them oriented the wrong way. You could delete the cache, then reload the terrain and toggle DDS Textures to the opposite setting.

To clear the file cache:

1. Choose **Settings** → **Application**. The Application Settings dialog box opens.
2. Select the File Caching Settings page (Figure 29-1).
3. Click Clear Accelerated File Loading Cache.

29.2. Displaying DDS Textures Correctly

DDS is a DirectX bitmap format that is often used for textures. DirectX expects the origin of the image to be at the top left corner. OpenGL expects the image origin to be at the bottom left corner. By default, VR-Vantage assumes that images have the OpenGL orientation. However, sometimes images are not oriented properly and therefore you may have to tell VR-Vantage to flip them.

You can change the DDS textures setting as follows:

- On a per-model basis. (For details, please see "Flipping DDS Textures for a Model," on page 35-23.)
- When you add a terrain patch. (For details, please see "Adding Elevation Data (Terrain Patches) to a Terrain," on page 25-7.)
- As a global setting. (For details, please see "Flipping DDS Textures Globally," on page 29-7.)

29.2.1. Flipping DDS Textures Globally

You can specify that the DDS textures setting apply to all terrains. This setting is primarily beneficial for paged terrains because you cannot know ahead of time if textures need to be flipped, as you could for a model or terrain patch. If a paged terrain has DDS textures that are upside down relative to VR-Vantage's expectations, the textures can be flipped when the terrain gets paged in.

To enable or disable DDS texture flipping for all terrains:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Loader Settings page (Figure 29-3).

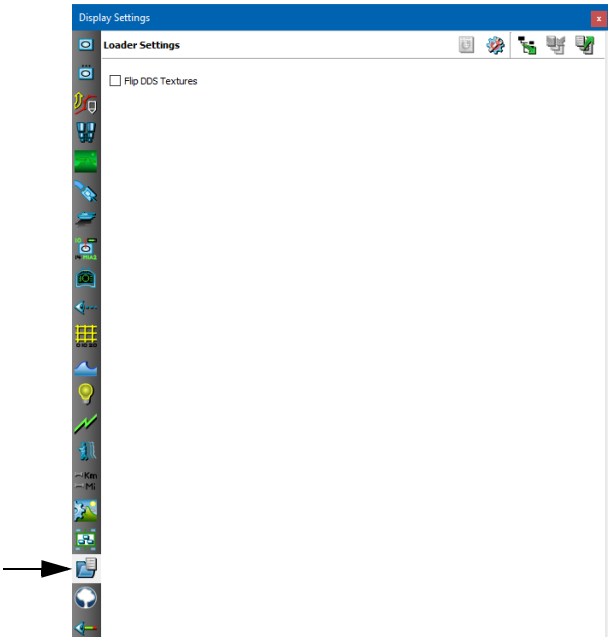


Figure 29-3. Loader Settings page

3. Select the Flip DDS Textures check box.
4. Click Close.

29.3. Preprocessing Paged Terrains

VR-Vantage supports preprocessing external references used by paged terrains. VR-Vantage does not require terrain formats such as TerraPage or pageable IVE to be preprocessed. However these terrains may use external references in formats such as OpenFlight, which load slowly. Although VR-Vantage can accelerate these files at run-time using the accelerated cache feature, the first time you load one of these terrains you can experience significant performance issues. You can use the medfTool to preprocess these external references and save them in MEDF format. Thereafter, VR-Vantage will load the accelerated version at all times.

For example, if a TerraPage terrain references */externals/tree.flt*, preprocessing creates a file called */externals/tree.medf*. When VR-Vantage loads the terrain, it loads the MEDF version instead of the OpenFlight version.

We recommend that you preprocess all data to minimize load times and maximize run-time performance. For details about the medfTool, please see "Compressing Model Files," on page 35-28.

29.4. Configuring Paged and Streaming Terrains

You can improve performance for some terrains by configuring the following environment variables:

- OSG_MAX_PAGEDLOD. OSG_MAX_PAGEDLOD specifies the target maximum number of paged LODs to maintain. Default: 200. Set it to a lower value if you need to reduce memory consumption when using paged terrains.
- GDAL_CACHEMAX. GDAL_CACHEMAX specifies the maximum cache size for streaming terrain. The default value is 200 MB, which is often adequate. However, if you are using a highly detailed terrain, increasing the cache size to a larger value, such as 2000, can significantly improve performance.

29.5. Debugging Earth Files

Earth files can contain many layers of elevation, imagery, and feature data. When you are debugging performance problems for a terrain that uses an earth file it can be helpful to turn off individual layers and see how that affects performance. You can do that with the osgEarth Profile dialog box.

In addition to seeing how enabling and disabling a layer affects performance, you change test the effect of changing the opacity of a layer and disabling lighting. You can only disable lighting for model layers (features area specified using the `model` element).



This feature is for use in debugging only. You cannot use it to configure different views of the terrain. Changes are not saved. The earth file is not edited.

To turn earth file layers on and off:

1. Load an earth file.
2. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
3. Select the Terrain Contents page (Figure 29-4).

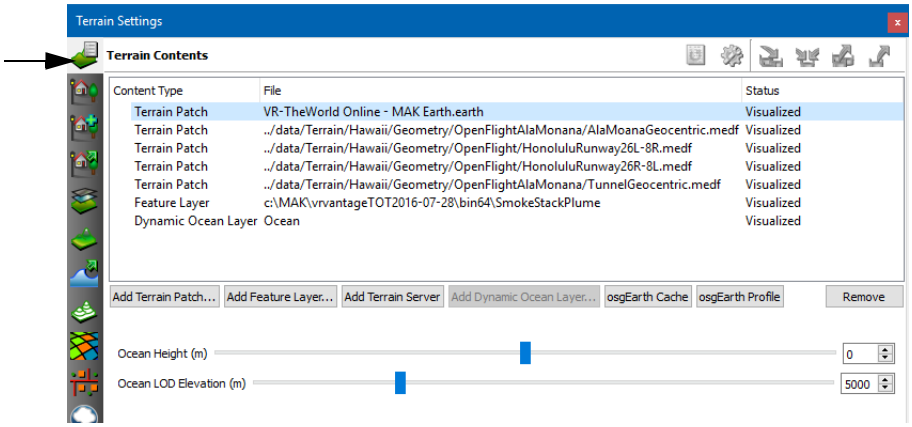


Figure 29-4. Terrain Contents page

4. In the terrain contents window, select the earth file that you want to debug.
5. Click osgEarth Profile. The osgEarth Profile dialog box opens (Figure 29-5).

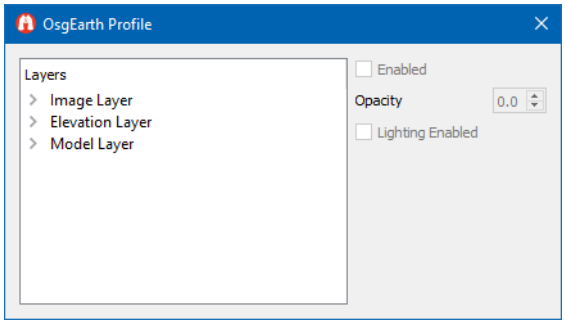


Figure 29-5. osgEarth Profile dialog box

6. Expand the list of the layers that you want to check.
7. Select a layer in the list. Clear the Enabled check box and see how performance changes.
8. To see the effect of lighting on a particular layer, clear the Lighting Enabled check box.
9. Optionally, change the opacity of a layer.
10. Repeat this process for each layer that you want to check.
11. Click Close.

29.6. Using Shader-based Effect Maps

VR-Vantage supports several types of shader-based effect texture maps. Shaders are computer programs that run on the graphics processing unit (GPU). VR-Vantage makes extensive use of them for lighting effects.

Effect maps are raster images that apply highly realistic textures to the terrain and models. By applying different types of effect maps to terrain and models, you can improve the visual qualities of your simulation without the overhead of high polygon counts.

VR-Vantage supports the following types of shader-based effect maps:

- Normal, or bump, maps. Normal maps give terrains the appearance of relief, such as a rocky landscape.
- Specular maps. Specular maps affect the highlight color of objects.
- Ambient occlusion maps. Ambient occlusion maps model areas that do not receive direct light, such as cracks and crevices and shaded areas of terrain and models. These areas are lit only by ambient light.
- Reflection maps. Reflection maps affect the reflectivity of surfaces, such as windows. Reflection maps reflect objects in the sky, not the terrain.
- Emissive maps. Emissive maps control the emissivity of whatever they are applied to based on the current ambient light values. The alpha channel is an ID that is set from 0-255. All the pixels that should light together (like a window, and perhaps a surrounding sill) should have the same ID. When the ambient light level drops below a given ID level, those pixels are emissive. Channel 2 (green) is used to indicate the intensity of the emissivity. These textures need to use a lossless compression scheme, such as PNG. *LittlePond.mtf* has sample emissivity textures on many of the houses.
- Emissive Nightmap maps. 3 Channel.
- Lightmap maps. 3 Channels. Lightmaps store precomputed lighting for surfaces in a scene.
- Gloss maps. 1 Channel. Gloss maps control how wide or narrow the specular highlight appears.
- Reflection maps. 4 Channel. Reflects the environment onto an object.
- Nightmaps maps. 3 Channel.
- Flow maps. 3 Channel. Flow maps define direction-based distortion, such as water flow.

You can include effect maps as part of an OpenFlight model or terrain or as raster maps (on the Raster Maps page of the Terrain Settings dialog box). VR-Vantage uses a naming convention for effect map files that lets it process the files properly without any additional configuration. To specify the type of map file, use the following naming conventions:

- Normal map: *filename_NML.png*
- Ambient Occlusion map: *filename_AO.png*
- Specular map: *filename_SPC.png*
- Reflection map: *filename_RFL.png*
- Emissive map: *filename_EMM.png*.
- Emissive Nightmap: *filename_NGT.png*.
- Lightmap: *filename_LMM.png*.
- Gloss maps: *filename_GLS.png*.
- Reflection map: *filename_RFL.png*.
- Nightmap: *filename_NTE.png*.
- Flow map: *filename_FLO.png*.



The map code must be uppercase on Linux.

To see the effect of effect maps, you must enable lighting. For details, please see Chapter 13, *Lighting Effects*.

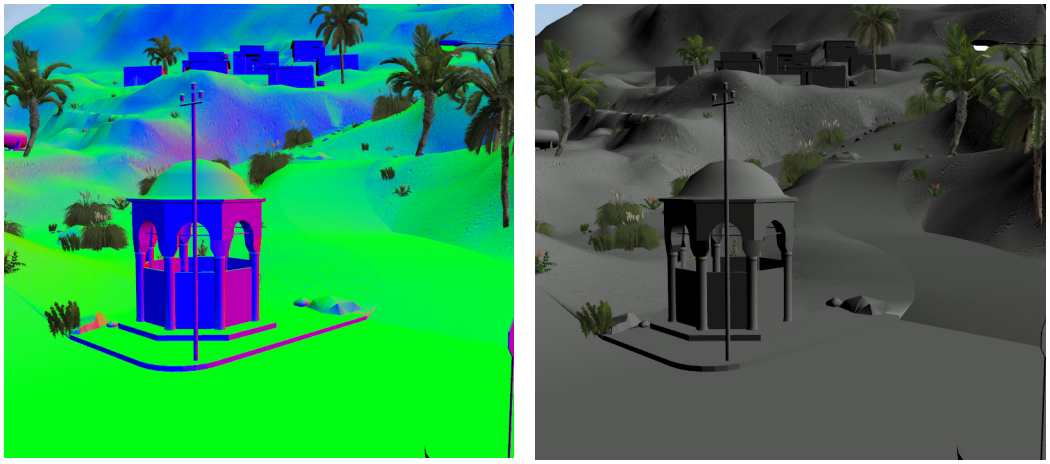
29.6.1. Composite Effects Maps

Ffects maps can be composited into a single file using channels. The name of the file represents each effects map in its respective channel, Red, Green or Blue (for example, House_GLS_SPR_AO_REF.png - Glossy, Specular, Ambient Occlusion and Reflection composite map). Only one of the textures can be a composite texture, but all files have to use the Composite Effects Maps naming convention.

- NML: 3 Channel Normal map
- SPR: 1 Channel Specular map
- EMS: 1 Channel Emissive map
- LMM: 3 Channel Lightmap (Supported for CDB)
- GLS: 1 Channel Gloss map
- AO: 1 Channel Ambient Occlusion Map
- REF: 1 Channel Reflection map
- NGT: 3 Channel Emissive Nightmap map
- NTE: 3 Channel Nightmap map
- FLO: 1 Channel Flow map.

29.6.2. Debugging Shaders

VR-Vantage provides some shader debugging capabilities within its lighting shaders. The shaders are instrumented such that setting different flags (through uniforms) changes the behavior of the shader. Typically it changes fragment colors based on some parameter or parameters. Figure 29-6 shows a view of VR-Village with two different shader debug settings.



Bumpmap Normals | Diffuse Light

Figure 29-6. Shader component visualization

To debug shaders:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 29-7).

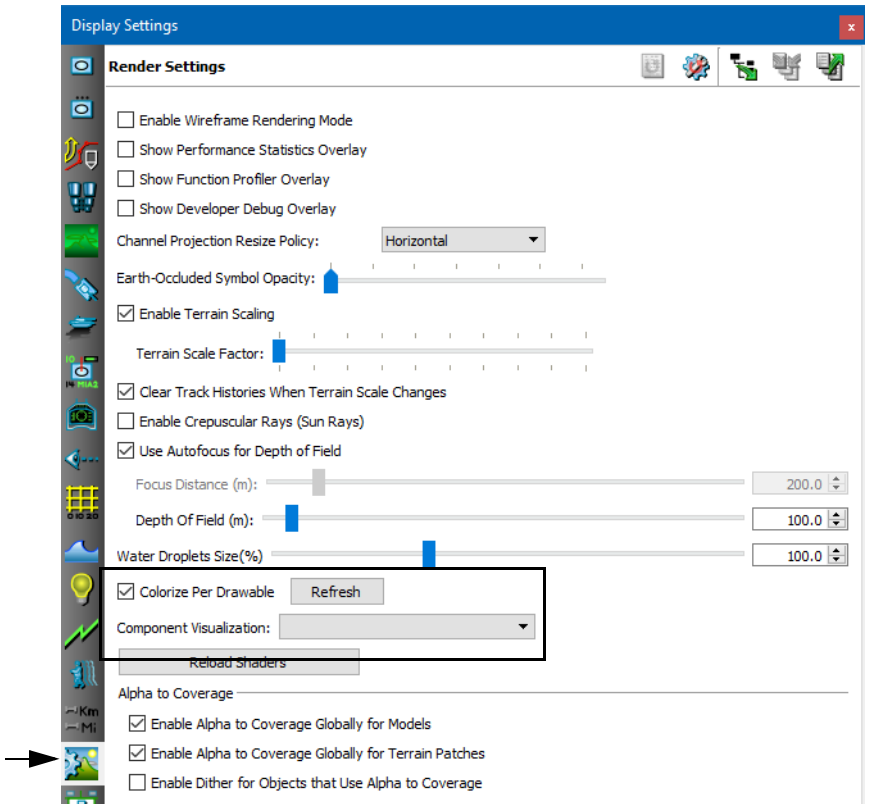


Figure 29-7. Render Settings page

3. Select an option from the Component Visualization list. (If you have SensorFX installed, the list includes options for visualizing materially classified textures.)

29.6.3. Reloading Shaders

Developers who are fine tuning shaders for terrain and object models do so by editing text files that configure the shaders. They can load their new shaders without shutting down VR-Vantage. Compile failures are reported in the console. We recommend that you display it while debugging.

To dynamically reload shaders:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Render Settings page (Figure 29-7).
3. Click Reload Shaders.

29.7. Displaying Terrain Intersection Lines

You can display terrain intersector lines to help you debug problems with terrain performance.

To display terrain intersection Lines

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Intersector Settings page (Figure 29-8).

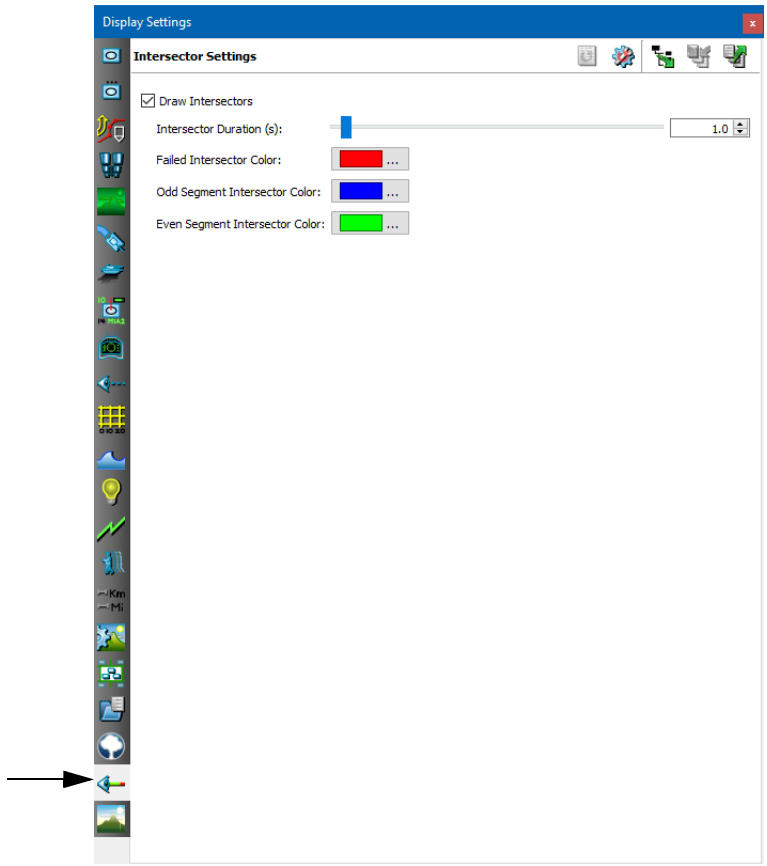


Figure 29-8. Intersector Settings page

3. Select the Draw Intersectors check box.
4. Optionally, set the duration of the intersector lines.
5. Optionally change the colors used for intersector lines.



30. Terrain Tutorials

This chapter explains how to create a local terrain server. It also shows how to extract props from a terrain.

Creating a Terrain Server 30-2
- Add a Map Element 30-2
- Add Elevation Data to the Earth File 30-2
- Add Imagery to the Earth File 30-3
- Add Feature Data to the Earth File 30-3
- Add Options to the Earth File 30-6
- Save the Earth File 30-7
- Load the Earth File and Save is as an MTF File 30-7

Extracting Props from a Terrain 30-8
- Create a New Terrain and Add a Terrain Patch 30-9
- Extract the Props 30-9
- Change the Model Definition for a Prop 30-11

30.1. Creating a Terrain Server

Terrain servers are configured in earth files, which are XML files using osgEarth syntax. All the information that VR-Vantage needs to load elevation, imagery, and feature data is provided. The data can be stored locally, on a network, or the internet. Data is loaded locally or streamed over the network and paged in as needed.

VR-Vantage comes with a terrain *Little Pond (online).earth* that merges the local Little Pond data with streaming terrain from VR-TheWorld Server. For this tutorial we will ignore the rest of the world and focus on Little Pond.

One approach to creating an earth file is to start with one of the earth files provided with VR-Vantage, keep the basic structure, and add, edit, or replace elements as needed. For this tutorial, we will start from an empty file and build all the major elements.

30.1.1. Add a Map Element

The root element in an earth file is the `map` element. It encloses all other elements. Open a text editor and add the following code:

```
<map name="Little Pond" type="geocentric" version="2">


</map>
```

30.1.2. Add Elevation Data to the Earth File

Elevation data is configured using an `elevation` element. We want to specify the same elevation data here that we did in the previous Little Pond tutorial. Add the following text after the opening of the `map` element:

```
<elevation name="Little Pond Elevation" driver = "gdal">
  <url>../../data/Terrain/LittlePond/Elevation/w072n042.dt1</url>
  <tile_size>32</tile_size>
</elevation>
```

The location of data in an earth file is provided in a `url` element. We need to use the gdal driver to load elevation data.

30.1.3. Add Imagery to the Earth File

Image data is configured using `image` elements. Add the following text after the `elevation` block:

```
<image driver="composite">
  <image name="Little Pond Imagery" driver="gdal">
    <url>../../data/Terrain/LittlePond/Imagery/W72N42.tif</url>
  </image>

  <image name="Little Pond Imagery" driver="gdal">
    <url>../../data/Terrain/LittlePond/Imagery/LittlePond.tif</url>
  </image>
</image>
```

The Little Pond terrain has two image files, a large low-resolution image and a smaller higher resolution image. If we were to simply add two `image` elements to the earth file, only the last one listed would be loaded. We would only see one of the images.

To use more than one image, we place an `image` element for each image file inside an `image` element that uses a composite driver. List the images from bottommost to topmost. If you list them in the opposite order from the code, *LittlePond.tif* would be underneath *W72N42.tif* and you would not be able to see it.

We use the gdal driver to load the images.

30.1.4. Add Feature Data to the Earth File

In earth files, you add feature data, such as buildings and vegetation, using a `model` element. You can add as many `model` elements as you need. A `model` element specifies the location of a shapefile. Then it adds `style` elements for each feature type to add from the shapefile. In the previous Little Pond tutorial, we added feature layers using *BuildingsP.shp* and *VegetationP.shp*. Then we added one kind of building (green colonial garrison house) and one kind of plant (big leaf maple tree). In this tutorial we will add one kind of house and all of the vegetation in the shapefile.

Add a Building to the Earth File

Add the following code to your earth file after the `image` block:

```
<model name="Little Pond Buildings" driver="feature_geom">
  <features name="Buildings" driver="ogr">
    <url>../../data/Terrain/LittlePond/Features/BuildingsP.shp</url>
  </features>
  <layout>
    <tile_size_factor>3</tile_size_factor>
  </layout>
  <styles>
    <style type="text/css">
    HousesColonialGarrison {
    model:
  "../../data/Structures/HousesColonialGarrisonClapboardGreen.medf";
    model-heading:       0-[rotation];
    model-scale:        1;
    model-placement: vertex;
    altitude-clamping:   terrain;
    altitude-resolution: 0.00008583068847656250;
    altitude-offset:     0;
    }
    </style>

    <selector class="HousesColonialGarrison">
      <query>
        <expr>TYPE = 'HousesColonialGarrison'</expr>
      </query>
    </selector>
  </styles>
  <clustering>false</clustering>
  <lighting>true</lighting>
</model>
```

As in the other elements, the `url` element specifies the location of the shapefile. To load buildings, you need to use the feature_geom driver for the `model` element and the ogr driver for the `features` element.

The `styles` element lets you specify a `style` for each building you want to display. This example shows just one building. Take a look at *Little Pond (online).earth* to see the other buildings that can be configured from this shapefile.

Each style has a name, in this case HousesColonialGarrison. The `model` attribute specifies the model used to display the object. The `selector` element matches the style to an entry in the TYPE column in the database file that accompanies the shapefile (*BuildingsP.dbf*).

To better understand this relationship, open *BuildingsP.dbf* in a spreadsheet application (Figure 30-1). Remember that in the first Little Pond tutorial, you built a rule for extracting props from the feature layer. You chose TYPE in the field list. That is because the TYPE column lists the different types of buildings in the shapefile. You do the same thing with the `selector` element.

	A	B	C	D	E
1	LENGTH	WIDTH	HEIGHT	ROTATION	TYPE
203	8.21000000000	11.02000000000	9.00000000000	345	HousesColonialGarrison
204	8.21000000000	11.02000000000	9.00000000000	35	HousesColonialGarrison
205	8.21000000000	11.02000000000	9.00000000000	320	HousesColonialGarrison
206	8.21000000000	11.02000000000	9.00000000000	230	HousesColonialGarrison
207	8.21000000000	11.02000000000	9.00000000000	230	HousesColonialGarrison
208	8.21000000000	11.02000000000	9.00000000000	250	HousesColonialGarrison
209	8.21000000000	11.02000000000	9.00000000000	155	HousesColonialGarrison
210	8.21000000000	11.02000000000	9.00000000000	325	HousesColonialGarrison
211	8.21000000000	11.02000000000	9.00000000000	250	HousesColonialGarrison
212	8.21000000000	11.02000000000	9.00000000000	140	HousesColonialGarrison
213	8.21000000000	11.02000000000	9.00000000000	335	HousesColonialGarrison
214	8.21000000000	11.02000000000	9.00000000000	65	HousesColonialGarrison
215	8.21000000000	11.02000000000	9.00000000000	210	HousesColonialGarrison
216	8.21000000000	11.02000000000	9.00000000000	125	HousesColonialGarrison
217	8.21000000000	11.02000000000	9.00000000000	335	HousesColonialGarrison
218	8.21000000000	11.02000000000	9.00000000000	30	HousesColonialGarrison
219	8.21000000000	11.02000000000	9.00000000000	335	HousesColonialGarrison

Figure 30-1. *BuildingsP.shp*

Add Vegetation to the Earth File

To add vegetation to the terrain, add the following code to the earth file after the `model` block for the buildings:

```
<model name="Little Pond Vegetation" driver="feature_custom">
  <features name="Vegetation" driver="ogr">
    <url>../../data/Terrain/LittlePond/Features/VegetationP.shp</url>
  </features>
  <layout>
    <tile_size_factor>3</tile_size_factor>
  </layout>
  <styles>
    <style type="text/css">
      Vegetation {
      marker:              [veg] + "ST" ;
      marker-scale:        1;
      marker-placement: vertex;
      altitude-clamping:   terrain;
      altitude-resolution: 0.00008583068847656250;
      altitude-offset:     0;
      }
    style>
  </styles>
  <clustering>true</clustering>
  <lighting>true</lighting>
</model>
```

The vegetation in VR-Vantage uses SpeedTree models. To load SpeedTrees from an earth file, you need to use the feature_custom driver in the `model` element. You use the ogr driver in the `features` element. Similar to the buildings, the `model` block for vegetation specifies a shapefile and styles. However, unlike with the buildings, we do not specify individual models to load. The marker attribute indicates that VR-Vantage should take every entry in the VEG column of *VegetationP.shp*, append ST to it, and look for a model of that name. For example, the column has several entries named TreesBigLeafMaple. VR-Vantage adds ST to form TreesBigLeafMapleST, finds the model definition TreesBigLeafMapleST, and loads the model specified in the model definition.

30.1.5. Add Options to the Earth File

Finally, add some common options to the file. In particular, this section specifies a location for the cache. Add this block just before the closing of the `map` element:

```
<options>
  <lighting>true</lighting>
  <elevation_interpolation>triangulate</elevation_interpolation>
  <terrain>
    <min_tile_range_factor>3</min_tile_range_factor>
    <skirt_ratio>0.1</skirt_ratio>
  </terrain>
  <cache type="filesystem">
    <path>../../appData/cache</path>
  </cache>
</options>
```

30.1.6. Save the Earth File

The convention for saving earth files is to save them in the *./userData/servers* directory. Save the file as *myLittlePond.earth*.

30.1.7. Load the Earth File and Save is as an MTF File

Your earth file for Little Pond is complete. Now you can load it in VR-Vantage and compare it to the terrain you created through the GUI. After you load it, you should save it as an MTF file.

To load the earth file:

1. Start VR-Vantage.
2. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
3. Select the Terrain Contents page.
4. Click Add Terrain Server. The Add Terrain Server dialog box opens.
5. In the Servers list, select *myLittlePond.earth*.
6. Click Connect. The terrain loads. It is visible as a small patch on the earth's sphere (Figure 30-2).

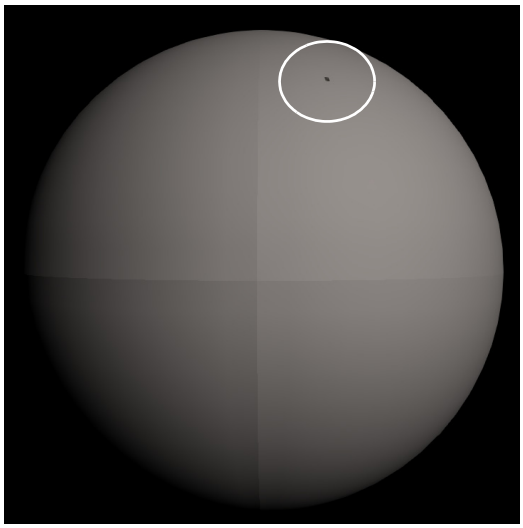


Figure 30-2. Little Pond terrain

7. Zoom in on the terrain (Figure 30-3).

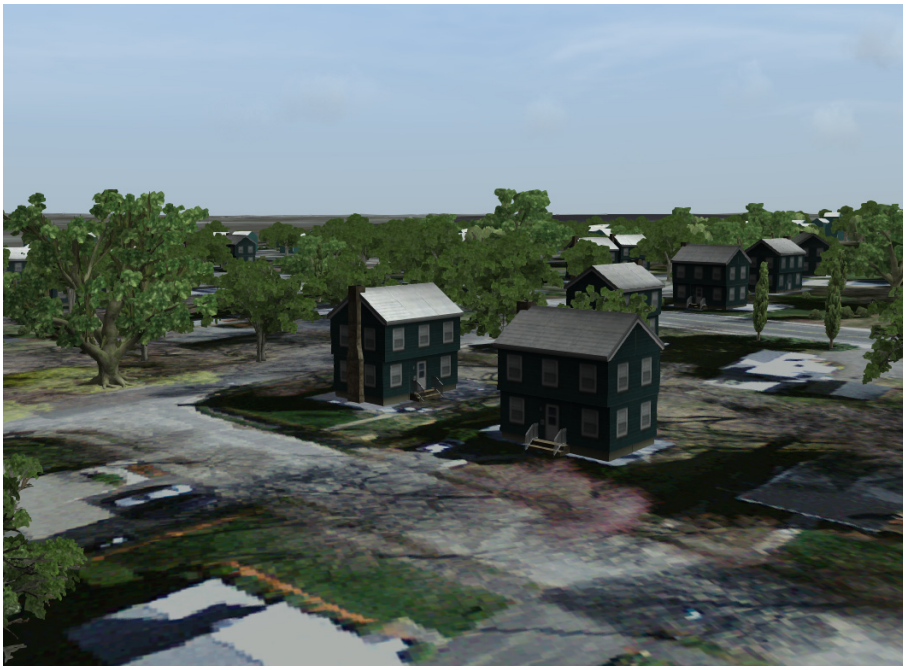


Figure 30-3. Little Pond terrain

8. Choose **File** → **Save Terrain**. The Save Terrain dialog box opens.
9. Give the terrain a name. Do not use Little Pond unless you want to overwrite the Little Pond terrain provided with VR-Vantage.

30.2. Extracting Props from a Terrain

In this tutorial, we show how to create a prop by extracting information from a terrain.

The general steps for extracting props from a terrain are:

1. Create a new terrain with a terrain patch ("Create a New Terrain and Add a Terrain Patch," on page 30-9).
2. Select the external references that you want to extract as props and extract them ("Extract the Props," on page 30-9).
3. Optionally, change the model definitions ("Change the Model Definition for a Prop," on page 30-11).

30.2.1. Create a New Terrain and Add a Terrain Patch

To create the new terrain and add a terrain database:

1. Choose **File** → **New Terrain**. If any terrain data is loaded, it is removed.
2. Choose **File** → **Add Terrain Patch**. The Add Terrain Patch dialog box opens and an Open dialog box opens on top of it.
3. Navigate to the *./data/Terrain/Makland/Geometry/OpenFlight* directory.
4. Select *Makland.flt*.
5. In the Add Terrain Patch dialog box, click OK. The terrain is loaded.

30.2.2. Extract the Props

To extract the props:

1. Move the observer so that you have a view of the plateau in Makland similar to Figure 30-4.



Figure 30-4. Makland

2. Click on the trees or buildings in the scene. Notice that you cannot select them. They are part of the terrain; they do not exist as props.
3. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
4. Select the Extract Props page (Figure 30-5). On the Extract References tab, the External References window lists all of the objects in the terrain that are incorporated into the terrain by reference. These are the objects that you can extract as props. You can extract them individually. For this tutorial, we will extract all of them.

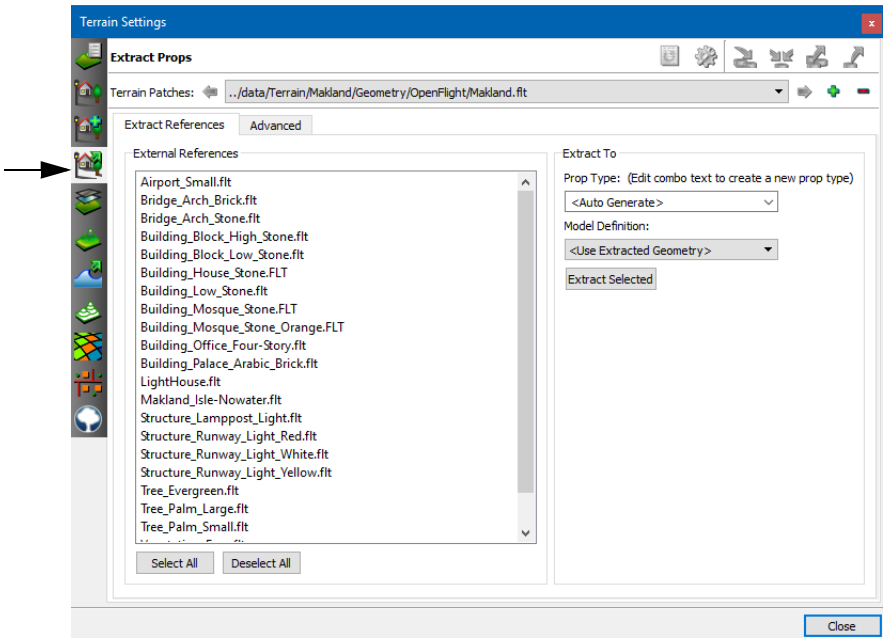


Figure 30-5. Extract props page

5. Click Select All. All of the references are selected.
6. In the Extract To group box, click Extract Selected. The objects are extracted as props. The list of external references is cleared.
7. Click the trees and buildings on the plateau. You can now select them because they are props, not part of the terrain (Figure 30-6). They are displayed using the geometry in the OpenFlight files from which they were referenced. You can change the models that the props use if you want to. For example, you might want to change the trees to SpeedTrees.



Figure 30-6. Selected props

30.2.3. Change the Model Definition for a Prop

To change the model definition for a prop:

1. Select one of the trees.
2. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
3. Select the Edit Existing Props page. The selected prop is highlighted in the list of props (Figure 30-7).

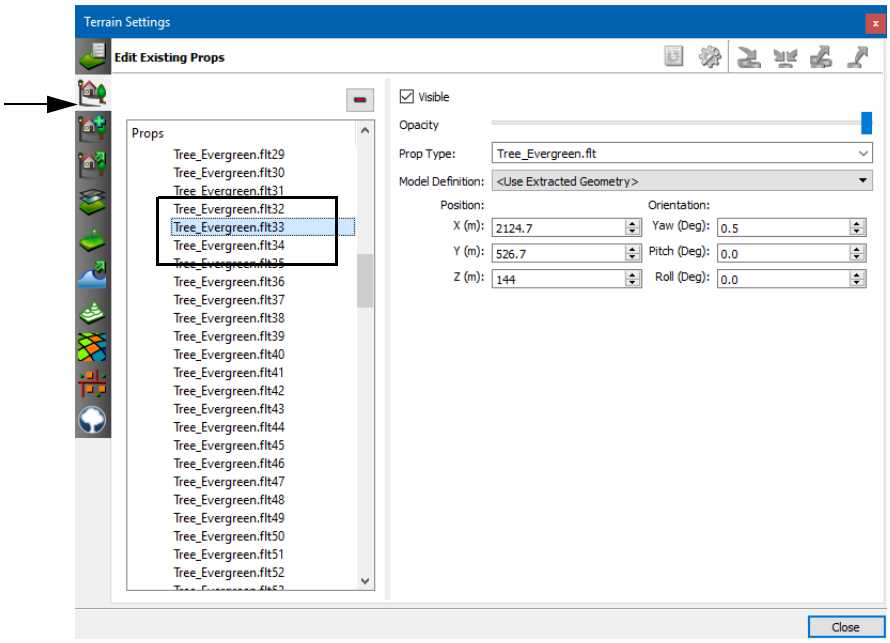


Figure 30-7. Edit Existing Props page

4. In the Model Definition box, select a model definition from the list, such as TreesGiantRedwoodST. The tree changes to the new model definition (Figure 30-8).



Figure 30-8. New model definition for tree

5. To restore the tree to its original model, on the Edit Existing Props page, change the model definition to Use Extracted Geometry.



VIII. Modeling Simulation Objects

When VR-Vantage discovers a simulation object (entity or unit) on the network, it must decide how to represent it in the GUI. This is accomplished by mapping 3D models and 2D icons to object type enumerations. The chapters in this section explain how models and iconography are configured and mapped to object types.

VIII.1. Introduction to VR-Vantage Object Modeling

The objects that VR-Vantage can display that are not part of the terrain, such as simulation objects, tactical graphics, detonations, and cockpit displays, are generically called scene elements. Scene elements are defined in element definitions. When VR-Vantage joins an exercise, it receives information about objects over the network. The objects are identified using object type enumerations. VR-Vantage maps these object types to element definitions. (For more information about object type enumerations and mapping them to element definitions, please see "Introduction to Object Type Mapping," on page 34-2.)

An element definition is a set of visual definitions that tell VR-Vantage how to render (display) the object. Visual definitions specify the 3D model, 2D icon, or graphic, and the visual effects that the object supports, such as trailing effects, height-above-terrain lines, track histories, and so on. A scene element can have up to three different element definitions – one for each model set (3D, 3D Colorized, 2D Icon). VR-Vantage chooses the correct element definition to use based on the model set being used by the observer. (For more information about model sets, please see "Model Sets," on page 30-4.)

Element definitions are created and edited in the Visual Model Editors dialog box. In addition to letting you manage element definitions, this dialog box lets you create and manage model definitions and schemas. Model definitions specify values for some basic object parameters. Most importantly for most VR-Vantage users, model definitions specify the 3D model to use for element definitions in the 3D Models model set. What this means is that if you want to add a new model to VR-Vantage, you will specify the model file, such as an OpenFlight file, in a model definition and then specify that model definition in the element definition for the simulation object you want to add.

Schemas define the most basic scene elements, such as an entity, SpeedTree tree, or line. Most VR-Vantage users will never have to create or edit a schema. However, if you create a new model definition, you must make sure it uses the correct schema and that it includes all required parameters.

Figure 30-1 illustrates these relationships using an M1A2 entity. VR-Vantage receives the object type 1:1:225:1:1:3:-1 over the network (callout 1). In the Simulation Object Type Mappings dialog box, this object type is mapped to the M1A2 Abrams entity definition (2). (An entity definition is a specific type of element definition.) The M1A2 Abrams entity definition (in the Visual Model Editors dialog box) specifies all of the visual definitions for this entity (3). The Main Model visual definition has a modeldefinition parameter. The value for this parameter is TrackedM1A2DESERT. The TrackedM1A2DESERT model definition has a filename parameter that specifies *M1A2.medf* as the model to use for entities that use this model definition (4). The TrackedM1A2DESERT model definition has a filename parameter because it uses the Entity schema and the Entity schema requires a filename (5).

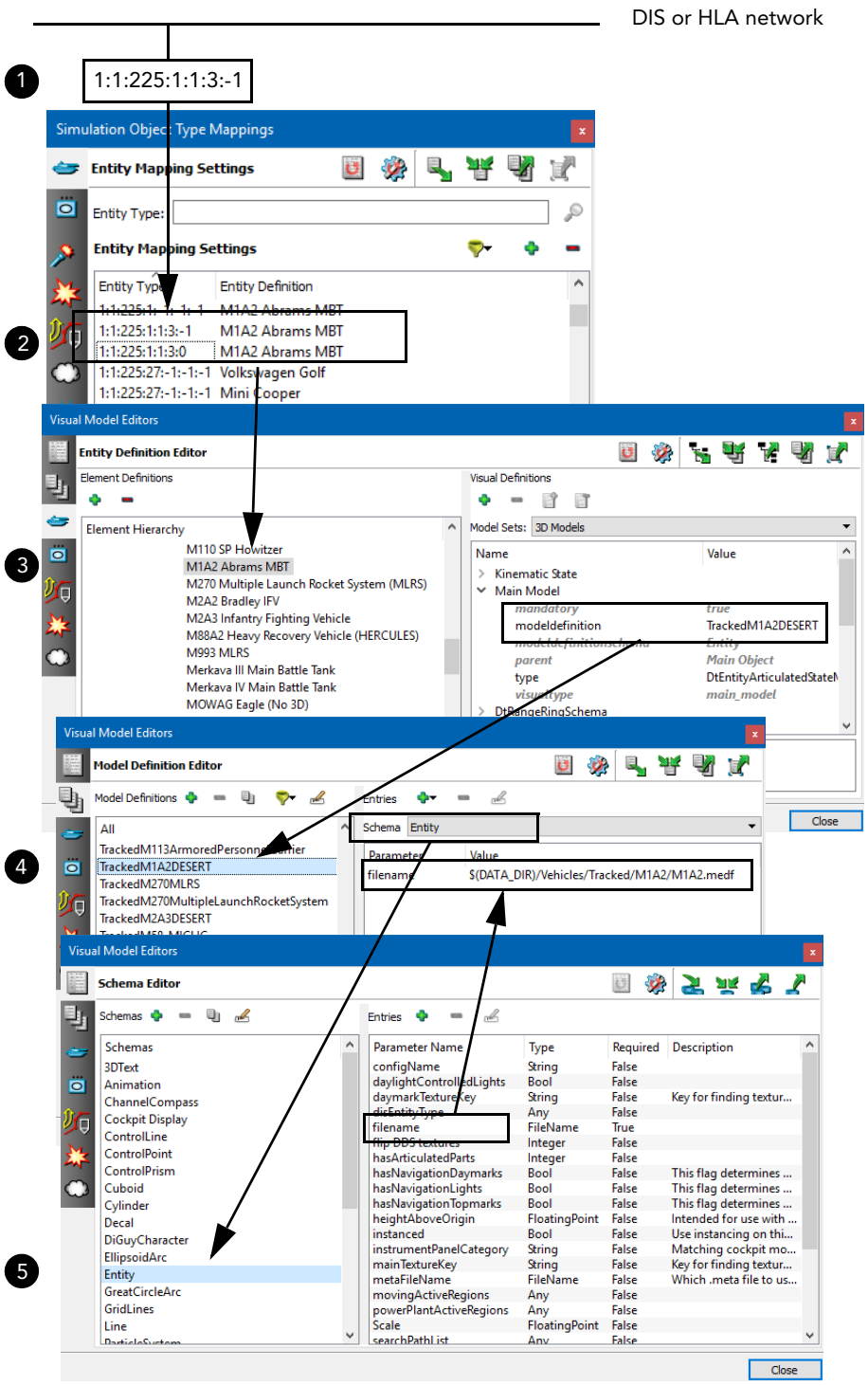


Figure 30-1. Element definitions and simulation object type mapping

VIII.1.1. Model Formats for 3D Entities, Objects, and Effects

VR-Vantage includes 3D models for vehicles, lifeforms, environmental objects, and effects. The models are in a compressed, encrypted format (extension *.medf*).



VR-Vantage includes a utility, the MEDF Tool, that lets you encrypt files in MEDF format. You cannot use it to decrypt encrypted files. For more information, please see "Compressing Model Files," on page 35-28.

You can provide your own models if they are in one of the following supported formats:

- OpenFlight.
- 3ds.
- OBJ.
- COLLADA (*.dae*).
- OSG (*.osg*, *.osgb*, and *.ive*).
- NewTek LightWave (*.lwo* and *.lws*).
- MAK GDB.

VIII.1.2. Model Sets

The model or icon that is used for an object is determined by its element definition. For convenience, element definitions are organized into model sets. For example, all of the element definitions that specify 3D models are in the 3D Models model set, and so on. As a result, when you choose the 3D Models model set for an observer mode, all of the entities use a 3D model. (The model set is an observer-specific setting. This is not really an issue in VR-Vantage PVD and VR-Vantage IG, which each support only one model set. VR-Vantage Stealth lets you change model sets.)

Each VR-Vantage application includes one or more of the following types of object models and icons:

- 3D Models. Realistic 3D models. This is the default model set for the Stealth observer mode.
- 3D Colorized Models. 3D colorized models are cartoon-like models that create an exaggerated notional view of the world. This is the default model set for XR observer mode.
- 2D Icons. 2D MIL-STD 2525B font-based icons. This is the default model set for the Plan View observer mode. You can also configure VR-Vantage to use 2525a images or any arbitrary images for 2D icons.

Table 30-1 lists the VR-Vantage model sets and the VR-Vantage applications that support them.

Table 30-1: VR-Vantage model sets

Model Set	Application
3D Models	VR-Vantage Stealth VR-Vantage IG
3D Colorized Models	VR-Vantage Stealth
2D Icons	VR-Vantage Stealth VR-Vantage PVD



The model sets available in each VR-Vantage application are fixed. You cannot add model sets.

For information about changing model sets, please see "Changing the Model Set," on page 19-11.

VIII.1.3. Mapping Object Types to Element Definitions

Object type mapping is the process of associating an object type enumeration for a simulation object or other scene element with an element definition.

DIS and HLA objects are specified by an object type enumeration. An object type enumeration is a list of seven components as specified by the DIS protocol and HLA RPR FOM. If none of the seven components contains a wildcard (-1) value, then the object type refers to a specific simulation object. If some components contain a wildcard, then the object type refers to a class of simulation objects. A larger number of wildcards indicates a broader, more general class. For complete details about object type enumerations, please see *Reference for Enumerations for Simulation Interoperability* (SISO enumerations document, formerly the DIS enumerations document). It is available at the IEEE World Wide Web site: http://shop.ieee.org/store/ and at the SISO web site: http://www.sisostds.org/.

When a VR-Vantage application discovers a simulation object, it reads the enumeration and then displays the simulation object using the element definition that is mapped to the enumeration.

For details about object type mapping, please see Chapter 34, *Mapping Object Types to Element Definitions*



31. Schemas

This chapter explains how to create and edit model schemas.

Creating and Editing Schemas 31-2
Creating Schemas 31-3
Deleting Schemas 31-4
Copying Schemas 31-4
Adding a Parameter to a Schema 31-5
Deleting a Parameter from a Schema 31-6
Editing a Schema Parameter 31-6

31.1. Creating and Editing Schemas

The elements that make up a scene have parameters that help define them, for example a model filename. The set of parameters for a particular scene element is a schema. VR-Vantage uses schemas to validate element definitions. You can create, copy, and delete schemas. For any schema, you can add and delete parameters and edit parameter values.



Changes to schemas are saved automatically. For more information, please see "Managing VR-Vantage Settings," on page 3-15.

Model definition schemas define the elements that can be displayed in a scene. The model definition schemas provided with VR-Vantage are:

- 3DText
- Animation
- Channel Compass
- Cockpit Display
- ControlLine
- ControlPoint
- ControlPrism
- Cuboid
- Cylinder
- Decal
- DiGuyCharacter
- EllipsoidArc
- Entity
- GreatCircleArc
- GridLines
- Line
- ParticleSystem
- Ribbon
- RotorWash
- SpeedTree
- TacticalSmoke
- Terrain
- TidalStreamWake
- Tree
- Wake
- Watercraft
- WaterImpact
- WidgetTheme.

31.2. Creating Schemas

To create a schema:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Schema Editor page (Figure 31-1).

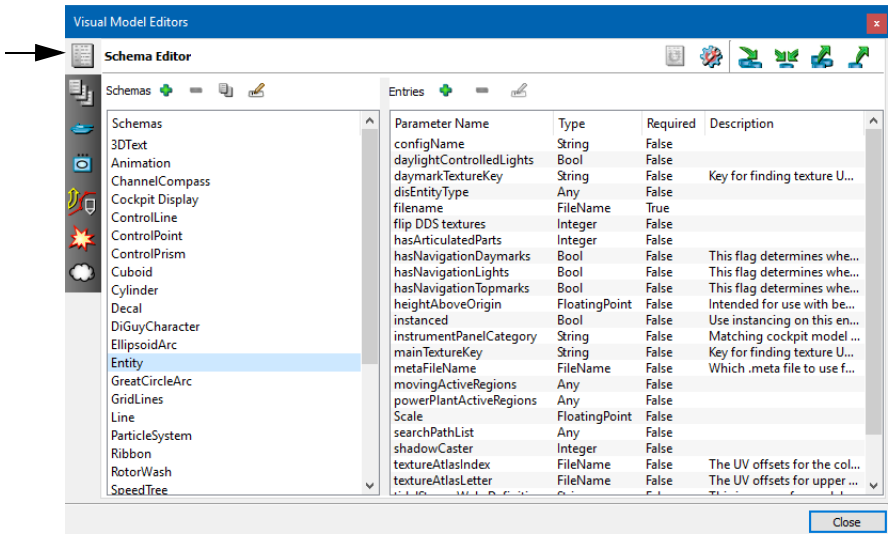


Figure 31-1. Schema Editor page

3. Click the Add button (+) next to the Schemas label. The Create Instance dialog box opens (Figure 31-2).

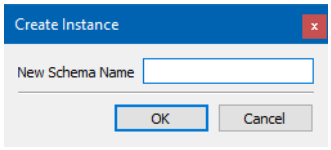


Figure 31-2. Create Instance dialog box

4. Type a name for the new schema.
5. Click OK.
6. Add parameters as described in "Adding a Parameter to a Schema," on page 31-5.

31.3. Deleting Schemas

> If a schema is used by a model definition, you cannot delete it. The Delete Schema button will be unavailable.

To delete a schema:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Schema Editor page (Figure 31-1).
3. In the Schemas list, select the schema you want to delete.
4. Click the Delete button (−) next to the Schemas label. You are prompted to confirm the deletion.
5. Click Yes. The schema is removed from the list.

31.4. Copying Schemas

If you want a new schema that is a variant of an existing one, it can be easier to copy the existing schema and edit it than it is to create a new schema and add all of the parameters.

To copy a schema:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Schema Editor page (Figure 31-1).
3. Select the schema that you want to copy.
4. Click the Duplicate button (). The Duplicate Schema dialog box opens.
5. Type a name for the copied version of the schema.
6. Click OK.
7. The new schema is added to the list.
8. Edit the parameters as described in "Editing a Schema Parameter," on page 31-6.

31.5. Adding a Parameter to a Schema

To add a parameter to a schema:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Schema Editor page.
3. In the Schemas list, select the schema to which you want to add a parameter.
4. Click the Add button (+) next to the Entries label. A parameter called "key" is added to the parameter list (Figure 31-3).

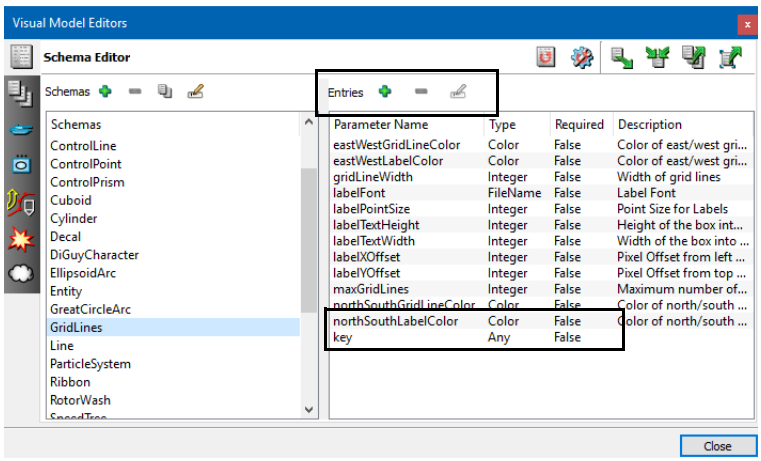


Figure 31-3. New schema parameter

5. Double-click "key". The Rename Parameter dialog box opens.
6. Type a name for the parameter.
7. If the parameter has a specific type, click the value in the Type column and edit it.
8. If the parameter is required, click the value in the Required column and change it to True.
9. Optionally, add a description. The description provides context sensitive help on the element definition pages. To add a description:
   a. Click the empty field in the Description column. The Edit Text dialog box opens.
   b. Type the descriptive text.
   c. Click OK.

31.6. Deleting a Parameter from a Schema

> **!** If you delete a parameter from a schema, any element definitions that use the schema may become invalid. Invalid element definitions are shown in red in the list of definitions.

To delete a parameter from a schema:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Schema Editor page (Figure 31-1).
3. In the Schemas list, select the schema from which you want to delete a parameter.
4. Select the parameter that you want to delete.
5. Click the Delete button (−) next to the Entries label.

31.7. Editing a Schema Parameter

To edit a parameter for a schema:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Schema Editor page (Figure 31-1).
3. In the Schemas list, select the schema for which you want to edit a parameter.
4. Click the parameter attribute that you want to edit. The attribute becomes editable.
5. Type the new attribute value.



32. Model Definitions

This chapter explains how to create and edit model definitions. "Adding a Model to VR-Vantage," on page 38-2 walks you through the process of adding a model.

Creating and Editing Model Definitions 32-2
Work Flow Issues Model Definitions 32-3
Creating a Model Definition 32-5
Deleting a Model Definition 32-6
Copying a Model Definition 32-7
Filtering the List of Model Definitions 32-8
Editing a Model Definition 32-8
- Adding Parameters to a Model Definition 32-8
- Editing a Model Definition Parameter 32-10
- Deleting Parameters from a Model Definition 32-11

Saving Model Definitions 32-11
Importing Model Definitions 32-12
- Importing Model Definitions from the Command Line 32-12

Exporting Model Definitions 32-13
- Exporting Selected Model Definitions 32-13

32.1. Creating and Editing Model Definitions

Many of the visual definitions in an element definition have a model definition parameter. A model definition specifies a set of parameters that are appropriate to the object type. (The parameters are determined by the model definition's schema. For information about model schemas, please see "Creating and Editing Schemas," on page 31-2.) For most VR-Vantage users, the most important thing to know about model definitions is that for 3D models, the model definition specifies the 3D model, such as an OpenFlight file, that is used to display an entity. If you want to add new models to VR-Vantage, you will probably have to create a new model definition for each one.

VR-Vantage comes with an extensive set of model definitions that are mapped to the included models. You can edit or delete these model definitions and you can create new ones.

Model definitions are created and edited on the Model Definition Editor page of the Visual Model Editors dialog box (Figure 32-1).

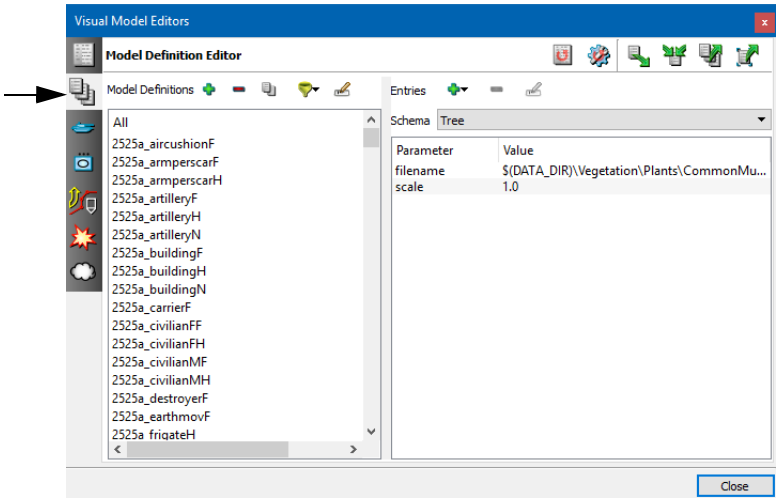


Figure 32-1. Model Definition Editor page

The model definitions provided with VR-Vantage are stored in *./appData/definitions/ModelDefinitions*. When you open the Visual Model Editors dialog box, it reads the files in this directory in alphabetical order. If files contain duplicate entries, the entry in the last file loaded takes precedence. You can use command line options to change the location of the model definitions that get loaded at startup. For details, please see "Importing Model Definitions from the Command Line," on page 32-12.

32.2. Work Flow Issues Model Definitions

A default VR-Vantage installation has multiple model definition files. (This is to accommodate the simulation model set organization in VR-Forces.) When you start VR-Vantage they all get loaded. When you open the Visual Model Editors dialog box, all of the model definitions are listed. However, the editor does not keep track of which model definitions come from which model definition file. When you create or edit model definitions, VR-Vantage does not automatically save them. Therefore, you are responsible for saving new and edited model definitions to the correct location.

The default model definition files are:

- *./appData/definitions/ModelDefinitions/AggregateLevelModel Definitions.ommx*. Model definitions specific to aggregate-level scenarios in VR-Forces.
- *./appData/definitions/ModelDefinitions/baseModel Definitions.ommx*. Model definitions used by both entity-level scenarios and aggregate-level scenarios in VR-Forces.
- *./appData/definitions/ModelDefinitions/coreModel Definitions.ommx*. Model definitions for objects that do not have *.entity* files in VR-Forces.
- *./appData/definitions/ModelDefinitions/EntityLevelModel Definitions.ommx*. Model definitions for simulation objects in entity-level scenarios in VR-Forces.

Before you add, delete, or edit a model definition, consider which of the following work flows best meets your need:

- If you want to create new model definitions, the preferred work flow is to export them to a new file. This is the easiest way to avoid overwriting other model definition files and makes them easily portable to newer releases of VR-Vantage.

  If you want to add the new model definitions to an existing model definition file:

  a. Replace the model definitions in the editor by using the Load and Replace option to load the model definition file you want to edit. (For details, please see "Importing Model Definitions," on page 32-12.)

  b. Add the new model definitions.

  c. Export all changes back to the original model definition file.

- If you have edited existing model definitions, you can export them to their original file or to a new file:
  - If you export to a new file, make sure that the file name is alphabetically later in alphabetical order than the original model definition file. This ensures that when the new model definition is loaded it will overwrite the original one.

– If you want to export to the original model definition file:

a. Replace the model definitions in the editor by using the Load and Replace option to load the model definition file you want to edit. (For details, please see "Importing Model Definitions," on page 32-12.)

b. Make your changes.

c. Export all changes back to the original model definition file.

- If you want to delete a model definition, you must delete it from the source model definition file. Follow the work flow for editing an original model definition file.

For more information about how to export model definitions, please see "Exporting Model Definitions," on page 32-13.

! Do not use the Export All option if you are editing the combined model definition files that get loaded by default unless you are very certain about how you want to use the resultant file. If you export to a file in the *./appData/definitions/ModelDefinitions* directory, it will conflict with the other files in that directory.

32.3. Creating a Model Definition

! VR-Vantage does not save changes to model definitions automatically. If you want to keep changes, you must export them to a file.

To create a model definition:

1. Decide if you want to export the new model definition to its own file or to an existing file. (For details about the model definition work flow, please see "Work Flow Issues Model Definitions," on page 32-3.)
2. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
3. Select the Model Definition Editor page (Figure 32-1).
4. If you want to export the new model definition to an existing file, use the Load and Replace procedure to load the file. (For details, please see "Importing Model Definitions," on page 32-12.)
5. Optionally, filter the model definitions in the list. (For details, please see "Filtering the List of Model Definitions," on page 32-8.)
6. Click the Add button (+) next to the Model Definitions label. The Add Model Definition dialog box opens (Figure 32-2).

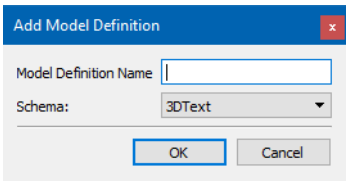


Figure 32-2. Add Model Definition dialog box

7. Type a name for the new model definition.

! Model definitions for DI-Guy characters must end with the letters DIG. Model definitions for SpeedTrees must end in ST.

8. In the Schema list, select a schema for the model. (This list matches the list of schemas on the Schema Editor page.)
9. Add parameters to the model, as described in "Adding Parameters to a Model Definition," on page 32-8.
10. Export the model definition. (For details, please see "Exporting Model Definitions," on page 32-13.)

32.4. Deleting a Model Definition

If you want to delete a model definition, you have to delete it from its original model definition file. Since the Visual Model Editors dialog box loads multiple model definition files when it starts up, you must be sure that you only load the model definition file that you want to edit.



If a model definition is being used, you cannot delete it.

To delete a model definition:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Model Definition Editor page (Figure 32-1).
3. Click the Import and Replace button (). The Import Model Definitions dialog box opens.
4. Select the model definition file you want to import.
5. Click Open.
6. Optionally, filter the model definitions in the list. (For details, please see "Filtering the List of Model Definitions," on page 32-8.)
7. Select the model that you want to delete.
8. Click the Delete button () next to the Model Definitions label. You are prompted to confirm the deletion.
9. Click Yes. The model definition is removed from the list.
10. Click the Export All button (). The Export Model Definitions dialog box opens.
11. Select the model definition file to save to.
12. Click Save.

32.5. Copying a Model Definition

If you want a new model that is a variant of an existing one, it can be easier to copy the existing model and edit it, than to create a new model and add all of the parameters.

! VR-Vantage does not save changes to model definitions automatically. If you want to keep changes, you must export them to a file.

To copy a model definition:

1. Decide if you want to export the new model definition to its own file or to an existing file. (For details about the model definition work flow, please see "Work Flow Issues Model Definitions," on page 32-3.)
2. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
3. Select the Model Definition Editor page (Figure 32-1).
4. If you want to export the new model definition to an existing file, use the Load and Replace procedure to load the file. (For details, please see "Importing Model Definitions," on page 32-12.)
5. Optionally, filter the model definitions in the list. (For details, please see "Filtering the List of Model Definitions," on page 32-8.)
6. Select the model that you want to copy.
7. Click the Duplicate button ( ). The Duplicate Model Definition dialog box opens (Figure 32-3).

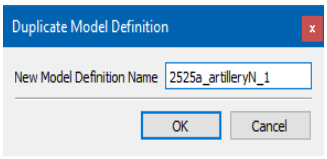


Figure 32-3. Copy model definition

8. Type a name for the copied version of the model.
9. Click OK.
10. The new model definition is added to the list.
11. Edit the parameters as described in "Editing a Model Definition," on page 32-8.
12. Export the model definition. (For details, please see "Exporting Model Definitions," on page 32-13.)

32.6. Filtering the List of Model Definitions

By default, the Model Definition Editor page lists all models that VR-Vantage knows about. You can filter the list to more easily work with a specific type of model.

➤ To filter the list of model definitions, click the Filter button (▾) and select a schema type from the list.

32.7. Editing a Model Definition

You can add parameters to a model definition, you can delete parameters, and you can change their values. You can only add parameters that are appropriate to the schema type for a model.

! VR-Vantage does not save changes to model definitions automatically. If you want to keep changes, you must export them to a file.

---

i If you change the model definition for a simulation object that is already in the scene, the updated definition is not applied to the object. To apply an edited model definition to all simulation objects, disconnect from the exercise and then reconnect.

32.7.1. Adding Parameters to a Model Definition

When you create a new model definition, you must specify the parameters required by its schema. You can only add parameters that are permitted by the schema.

To add a parameter to a model:

1. Decide if you want to export the edited model definition to its own file or to an existing file. (For details about the model definition work flow, please see "Work Flow Issues Model Definitions," on page 32-3.)
2. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
3. Select the Model Definition Editor page (Figure 32-1).
4. If you want to export the edited model definition to an existing file, use the Load and Replace procedure to load the file. (For details, please see "Importing Model Definitions," on page 32-12.)
5. Optionally, filter the model definitions in the list. (For details, please see "Filtering the List of Model Definitions," on page 32-8.)
6. Select the model for which you want to add a parameter.

7. Click the Add Parameter button (). A list of parameters is displayed (Figure 32-4).

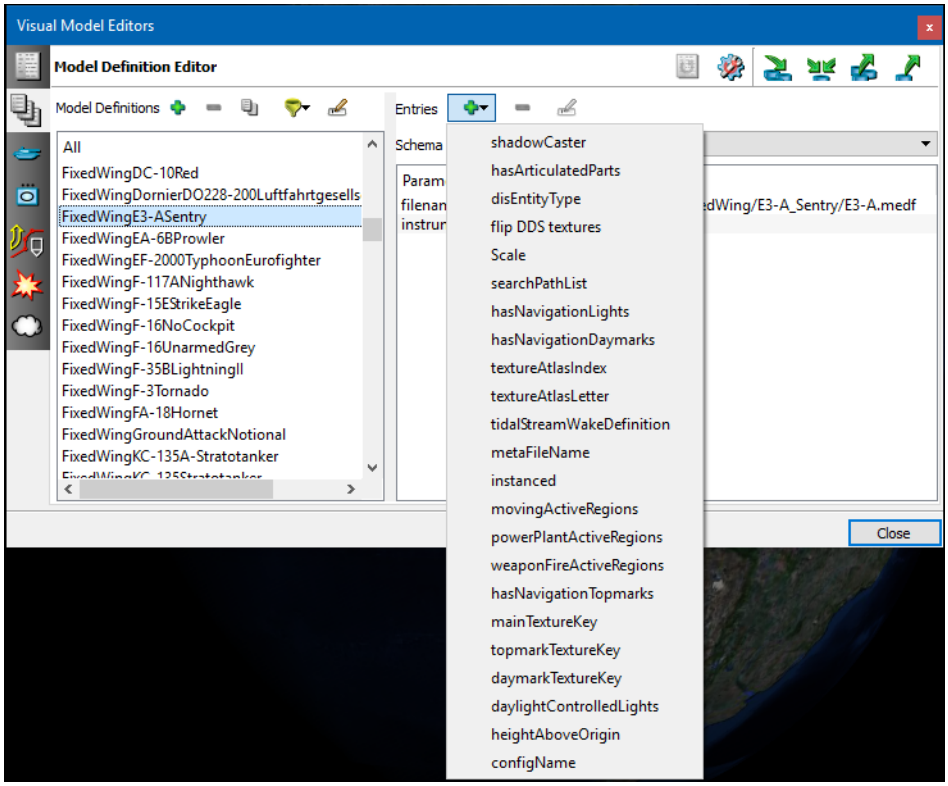


Figure 32-4. Add parameter list

8. Select the parameter you want to add. It is added to the parameter list.
9. Click the value for the parameter. The field becomes editable (Figure 32-5).

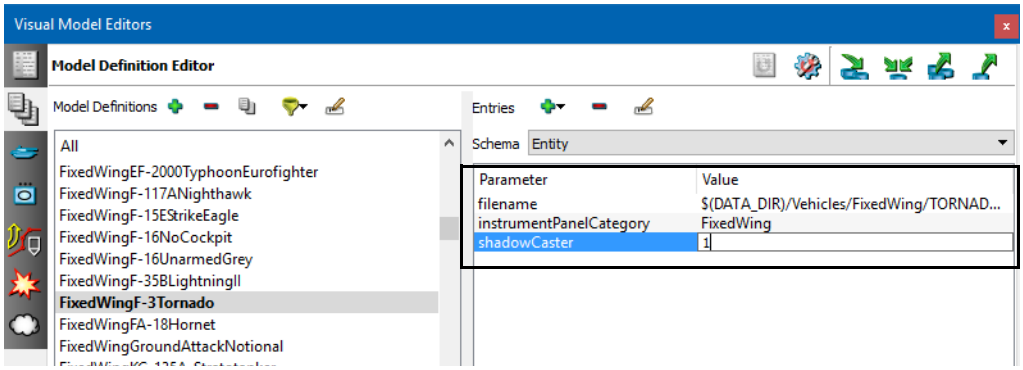


Figure 32-5. Editable parameter

10. Type or select a value for the parameter.
11. Click away from the field.

12. When you have finished editing the model definition, export it. (For details, please see "Exporting Model Definitions," on page 32-13.)

32.7.2. Editing a Model Definition Parameter

To edit a parameter:

1. Decide if you want to export the edited model definition to its own file or to an existing file. (For details about the model definition work flow, please see "Work Flow Issues Model Definitions," on page 32-3.)
2. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
3. Select the Model Definition Editor page (Figure 32-1).
4. If you want to export the edited model definition to an existing file, use the Load and Replace procedure to load the file. (For details, please see "Importing Model Definitions," on page 32-12.)
5. Optionally, filter the model definitions in the list. (For details, please see "Filtering the List of Model Definitions," on page 32-8.)
6. Select the model for which you want to edit a parameter.
7. Click the value for the parameter you want to edit. The field becomes editable (Figure 32-5).
8. Change the value for the parameter.
9. Click away from the field.
10. When you have finished editing the model definition, export it. (For details, please see "Exporting Model Definitions," on page 32-13.)

32.7.3. Deleting Parameters from a Model Definition

If you want to delete parameters from a model definition, you must export the changes back to the original file.

! If you delete a parameter from a model definition, the definition may become invalid. Invalid model definitions are shown in red.

To delete a parameter from a model:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Model Definition Editor page (Figure 32-1).
3. Use the Load and Replace procedure to load the model definition file that has the model definition you want to edit.
4. Optionally, filter the model definitions in the list. (For details, please see "Filtering the List of Model Definitions," on page 32-8.)
5. Select the model from which you want to delete a parameter.
6. Select the parameter that you want to delete.
7. Click the Delete button (−) next to the Entries label. The parameter is removed.
8. Export the model definition to the original model definition file using Export All. (For details, please see "Exporting Model Definitions," on page 32-13.)

32.8. Saving Model Definitions

If you add, edit, or delete model definitions, you must explicitly export the new or updated definitions to a file. For details, please see "Exporting Model Definitions," on page 32-13.

32.9. Importing Model Definitions

You can import model definitions and replace the loaded model definitions. You can import model definitions and merge them with the loaded model definitions. For details about how merging works, please see "Managing VR-Vantage Settings," on page 3-15.

To import model definitions and replace the existing definitions:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Model Definition Editor page (Figure 32-1).
3. Click the Import and Replace button (). The Import Model Definitions dialog box opens.
4. Select the model definition file you want to import.
5. Click Open.

To merge model definitions:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Model Definition Editor page (Figure 32-1).
3. Click the Import and Merge button (). The Merge Model Definitions dialog box opens.
4. Select the model definition file you want to merge.
5. Click Open.

32.9.1. Importing Model Definitions from the Command Line

You can merge in model definitions with the `--mergeModelDefs {`*`directory | filename`*`}` command-line option. If you specify a directory, VR-Vantage merges all model definition files in the directory. If you specify a file, it just merges that file. You can have multiple instances of this option in a command.

You can specify that VR-Vantage not load the model definitions in the *./appData/definitions* directory. To do this, start VR-Vantage with the `--noAppDataModelDefs` command-line option.

32.10. Exporting Model Definitions

You can export all model definitions and you can export individual model definitions. This allows you to create your own library of custom model definitions that you can easily migrate to newer releases of VR-Vantage or copy to other VR-Vantage installs.

! The list of model definitions in a default VR-Vantage installation is created by merging multiple model definition files. If you export all model definitions, they do not get exported back to their original files. They get exported to the file you select.

To export all model definitions:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Model Definition Editor page (Figure 32-1).
3. Click the Export All button (). The Export Model Definitions dialog box opens.
4. Select the model definition file to save to or type a new file name.
5. Click Save.

32.10.1. Exporting Selected Model Definitions

To export selected model definitions:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Model Definition Editor page (Figure 32-1).
3. Optionally, filter the model definitions in the list. (For details, please see "Filtering the List of Model Definitions," on page 32-8.)
4. Select the model definitions you want to save.
5. Click the Export Selected button (). The Export Model Definitions dialog box opens.
6. Select a file or type a name for the model definition file.
7. Click Save.



33. Element Definitions

This chapter explains how to create and edit element definitions. "Adding a Model to VR-Vantage," on page 38-2 walks you through the process of adding a model.

Creating and Editing Element Definitions 33-2
How Entity, Unit, and Tactical Graphics Element Definitions are Organized 33-2
Leaf File Descriptions 33-3
Work Flow Issues for Creating and Editing Element Definitions 33-4
Creating an Element Definition 33-5
Editing an Element Definition 33-7
Adding a Visual Definition 33-7
Editing a Visual Definition 33-8
Adding an Attribute to a Visual Definition 33-11
Deleting an Attribute from a Visual Definition 33-12
Deleting an Element Definition 33-13
Saving Element Definitions 33-13
Importing Element Definitions 33-14
Importing a Hierarchy File 33-14
Importing and Merging Element Definitions 33-15
Importing Element Definitions from the Command Line 33-15
Exporting Element Definitions 33-16
Exporting Element Definitions to .elem Files 33-16
Exporting Entity, Unit, or Tactical Graphics Element Definitions 33-17

33.1. Creating and Editing Element Definitions

Element definitions define how VR-Vantage renders scene elements (simulation objects, tactical graphics, interactions, and so on).

Element definitions are organized in a tree-style hierarchy. At any level of the hierarchy an element definition inherits the visual definitions of its parents. This makes it possible to assign a visual definition to many elements at one time. You can change the inherited parameter values for a specific element at any level of the hierarchy. However, in most cases, you cannot delete an inherited visual definition.

You can create and edit element definitions. The procedures are the same for each element type except for the save procedure. Element definitions are not saved automatically. You must save them explicitly.



If you are connected to an exercise, changes to element definitions do not take effect until you disconnect and reconnect.

33.2. How Entity, Unit, and Tactical Graphics Element Definitions are Organized

Entity, unit, and tactical graphics element definitions are stored in two types of files, a hierarchy file (*.hier*) and one or more leaf definition files (*.leaf*). The hierarchy file contains the hierarchy of categories for the element definitions, such as Entity → Surface → Boat and the attribute inheritance information associated with each level of the hierarchy. The leaf files contain the definitions for each element at the leaf level[1].

Other types of element definitions do not save separate hierarchy and leaf files. They are saved to a file with the extension *.elem*.

When you add or edit entity, unit, and tactical graphics element definitions, you must save the changes to the hierarchy and leaf files separately. If you have only made leaf level changes, you do not have to save the hierarchy file.

You can create multiple hierarchy and leaf level files to meet your needs. However you can only load one hierarchy file at a time. When VR-Vantage starts up, it loads the files in the various element definition directories. If it finds more than one hierarchy file for an element type, it only loads the first one (in alphabetical order). If it finds multiple leaf files, it loads them in alphabetical order. If there are any duplicate entries, the last file loaded takes precedence.

1. Technically, the leaf level of a tree structure is a node with no children. For element definitions, the leaf level consists of specific definitions. A category that has no members would be part of the hierarchy, not the leaf file.

33.2.1. Leaf File Descriptions

The element definitions for entity, unit, and tactical graphics are organized into multiple leaf files. This organization is to accommodate the data structure of VR-Forces, MAK's computer generated forces application. By default, VR-Vantage loads all of the element definitions so that it can render whatever simulation objects VR-Forces sends. The element definitions are organized as follows:

- *./appData/definitions/Aggregate/ElementDefinitions*. Element definitions for hierarchical units such as platoons and companies:
  - *EntityLevelAggregateElementDefinitions.leaf*. Units used in entity-level scenarios.
  - *coreAggregateElementDefinitions.leaf*. Element definitions for objects that do not have *.entity* files in VR-Forces.
  - *AggregateLevelAggregateElementDefinitions.leaf*. Units used in aggregate-level scenarios.
  - *baseAggregateElementDefinitions.leaf*. Unit elements used in both entity-level and aggregate level scenarios.
- *./appData/definitions/Entity/ElementDefinitions*. Element definitions for individual simulation objects such as humans, ships, and cars:
  - *EntityLevelEntityElementDefinitions.leaf*. Element definitions used in entity-level scenarios.
  - *coreEntityElementDefinitions.leaf*. Element definitions for objects that do not have *.entity* files in VR-Forces.
  - *baseEntityElementDefinitions.leaf*. Element definitions used in both entity-level and aggregate level scenarios.
- *./appData/definitions/TacticalGraphics/ElementDefinitions*. Element definitions for tactical graphics:
  - *EntityLevelTacticalGraphicsElementDefinitions.leaf*. Tactical graphics used in entity-level scenarios.
  - *coreTacticalGraphicsElementDefinitions.leaf*. Element definitions for tactical graphics that do not have *.entity* files in VR-Forces.
  - *AggregateLevelTacticalGraphicsElementDefinitions.leaf*. Tactical graphics used in aggregate-level scenarios.
  - *baseTacticalGraphicsElementDefinitions.leaf*. Tactical graphics used in both entity-level and aggregate level scenarios.

33.3. Work Flow Issues for Creating and Editing Element Definitions

A default VR-Vantage installation has multiple element definition files for entities, units, and tactical graphics. When you start VR-Vantage they all get loaded. When you open the Visual Model Editors dialog box, all of the element definitions are listed. However, the editor does not keep track of which element definitions come from which element definition file.



Interaction and smoke element definitions are loaded from just one file. They do not have hierarchy and leaf files. You cannot export specific element definitions. When you export them, all definitions get exported to the selected file. Therefore, except for the need to explicitly save all changes, the considerations in this section do not apply to them.

When you create or edit element definitions, VR-Vantage does not automatically save them. Therefore, you are responsible for saving new and edited element definitions to the correct location. This section describes the issues to consider when you work with entity, unit, and tactical graphics element definitions to get the results you want and avoid problems.

- If you have created new element definitions, the preferred work flow is to export them to a new leaf file. This is the easiest way to avoid overwriting other element definition files and makes them easily portable to newer releases of VR-Vantage.

  If you want to add the new element definitions to an existing element definitions file:

  a. Load the hierarchy file. This removes all element definitions.

  b. Load the element definition leaf file you want to edit.

  c. Add the new element definitions.

  d. Export all changes back to the original element definition leaf file.

- If you have edited existing element definitions, you can export them to their original leaf file or to a new leaf file:
  - If you export to a new file, make sure that the file name is alphabetically later in alphabetical order than the original element definition file. This ensures that when the new element definition is loaded it will overwrite the original one.
  - If you want to export to the original element definition file, you must plan ahead:

    a. Load the hierarchy file. This removes all element definitions.

    b. Load the element definition leaf file you want to edit.

    c. Make your changes.

    d. Export all changes back to the original element definition file.

For more information about exporting element definitions, please see "Exporting Element Definitions," on page 33-16.

! Do not use the Export All Leaves option if you are editing the combined element definition files that get loaded by default unless you are very certain about how you want to use the resultant file. If you export to an existing element definitions file, it will conflict with the other files in that directory.

33.4. Creating an Element Definition

You can create new element definitions. Entity, unit and tactical graphics element definitions get exported to *.leaf* files. Smoke and interaction element definitions get exported to *.elem* files.

! VR-Vantage does not save changes to element definitions automatically. If you want to keep changes, you must export them to a file.

To create an element definition:

1. If you are adding an entity, unit, or tactical graphic element definition, decide if you want to export the new element definition to its own leaf file or to an existing file. (For details about the element definition work flow, please see "Work Flow Issues for Creating and Editing Element Definitions," on page 33-4.)
2. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
3. Select the page for the type of element definition that you want to create.
4. If you want to export the new element definition to an existing leaf file, use the Load and Replace procedure to reload the hierarchy. (This deletes all element definitions.) Then load the element definition leaf file. (For details, please see "Importing Element Definitions," on page 33-14.)
5. Expand the element definition hierarchy to the level at which you want to add an element definition and select the parent element (Figure 33-1).

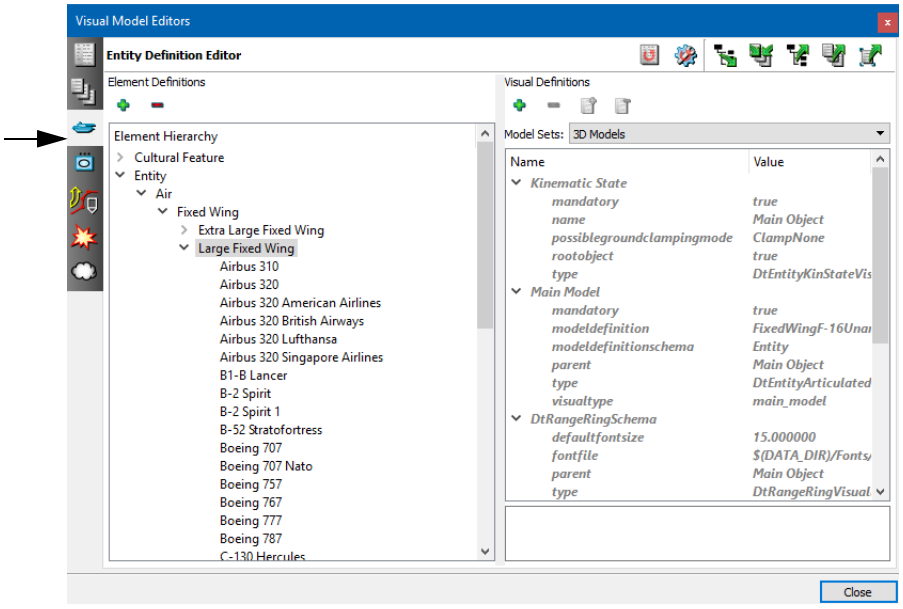


Figure 33-1. Entity Definition Editor page

6. Click the Add button (). The Create Element Definition dialog box opens.
7. Type a name for the element definition. (You cannot edit the name after you create it, so try to avoid typing errors.)
8. Click OK. The new element is added to the list.
9. Optionally, edit the visual definitions assigned to the element. For details, please see "Editing a Visual Definition," on page 33-8.
10. Optionally, add a visual definition to the element. For details, please see "Adding a Visual Definition," on page 33-7.
11. Export the element definition. If you made any changes to an entity, unit, or tactical graphics hierarchy, you must export the hierarchy also. (For details, please see "Exporting Element Definitions," on page 33-16.)

33.5. Editing an Element Definition

An element definition consists of a list of visual definitions for that element. You can add visual definitions and you can edit their parameters. In most cases, you cannot delete an inherited visual definition.

! VR-Vantage does not save changes to element definitions automatically. If you want to keep changes, you must export them to a file.

33.5.1. Adding a Visual Definition

To add a visual definition to an element definition:

1. If you are editing an entity, unit, or tactical graphic element definition, decide if you want to export the edited element definition to its own leaf file or to an existing file. (For details about the element definition work flow, please see "Work Flow Issues for Creating and Editing Element Definitions," on page 33-4.)
2. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
3. If you want to export the edited element definition to an existing leaf file, use the Load and Replace procedure to reload the hierarchy. (This deletes all element definitions.) Then load the element definition leaf file. (For details, please see "Importing Element Definitions," on page 33-14.)
4. Select the page for the type of element definition that you want to edit (Figure 33-1).
5. Select the element definition that you want to edit.
6. Optionally, on the Model Sets list, select the model set you want to use. (VR-Vantage Stealth only.)
7. Click the Add button (+) next to the Visual Definitions label. The Create Visualizer Definition dialog box displays a list of available visual definitions.

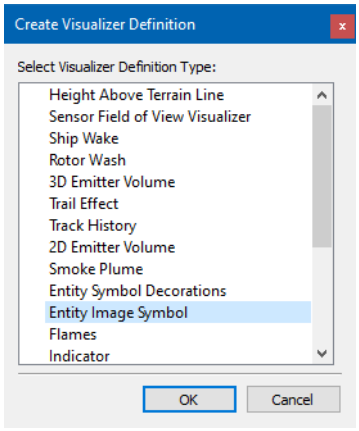


Figure 33-2. Create Visualizer Definition dialog box

8. Select the visual definition that you want to add.
9. Click OK. The visual definition is added to the element.
10. Optionally, edit the parameters for the visual definition. For details, please see "Editing a Visual Definition," on page 33-8.
11. When you have finished editing the element definition, export it. (For details, please see "Exporting Element Definitions," on page 33-16.)

33.5.2. Editing a Visual Definition

Visual definitions and their attributes are displayed either as gray italicized text or as black text (Figure 33-3). Italicized text indicates that the visual definition or attribute is inherited from a parent element. Visual definitions and attributes that are not italicized are particular to that element. (These entries could be inherited by child elements.)

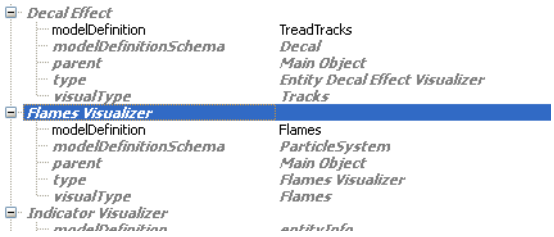


Figure 33-3. Visual definition entry

To edit a visual definition parameter:

1. If you are editing an entity, unit, or tactical graphic element definition, decide if you want to export the edited element definition to its own leaf file or to an existing file. (For details about the element definition work flow, please see "Work Flow Issues for Creating and Editing Element Definitions," on page 33-4.)
2. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
3. Select the page for the type of element definition that you want to edit (Figure 33-1).
4. If you want to export the edited element definition to an existing leaf file, use the Load and Replace procedure to reload the hierarchy. (This deletes all element definitions.) Then load the element definition leaf file. (For details, please see "Importing Element Definitions," on page 33-14.)
5. Select the element definition that you want to edit.
6. In the Visual Definitions pane, click the value of the attribute that you want to edit. Depending on the attribute, a list of available values is displayed, a dialog box is displayed (Figure 33-4), or the field becomes editable.

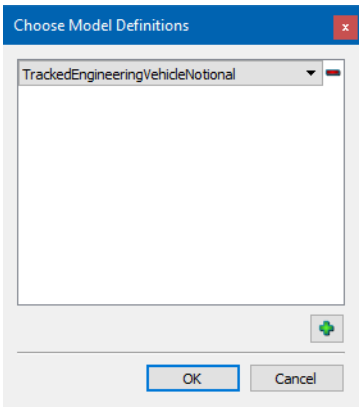


Figure 33-4. Choose Model Definitions dialog box

7. For a list, select the new value. For an editable field, type a new value. For a dialog box:
   a. If the dialog box window does not have a list, click the Add button (+). A list of values is added to the dialog box window.
   b. Select the value that you want.
   c. Click OK.
8. When you have finished editing the element definition, export it. (For details, please see "Exporting Element Definitions," on page 33-16.)

Mapping to Multiple Model Definitions or Schema

There may be cases where you want to map a visual definition to multiple model definitions or multiple schema. If you do this, VR-Vantage randomly assigns the model definition or schema to instances of the object.

To map a visual definition attribute to multiple values:

1. If you are editing an entity, unit, or tactical graphic element definition, decide if you want to export the edited element definition to its own leaf file or to an existing file. (For details about the element definition work flow, please see "Work Flow Issues for Creating and Editing Element Definitions," on page 33-4.)
2. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
3. Select the page for the type of element definition that you want to edit (Figure 33-1).
4. If you want to export the edited element definition to an existing leaf file, use the Load and Replace procedure to reload the hierarchy. (This deletes all element definitions.) Then load the element definition leaf file. (For details, please see "Importing Element Definitions," on page 33-14.)
5. Select the element definition that you want to edit.
6. In the Visual Definitions pane, click the value of the modeldefinition or modeldefinitionschema parameter that you want to edit. A dialog box opens. If the attribute already has values, they are listed in the dialog box window as a series of lists (Figure 33-5).

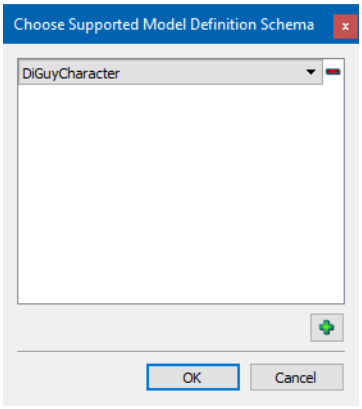


Figure 33-5. Parameter with multiple values

7. To add a value, click the Add button (+). A new list is added to the window.
8. Select a value from the new list.
9. To remove a value, click the Delete button (−) next to the list for that value.
10. Click OK. The visual definitions list does not show all of the values, but clicking it opens the dialog box again and shows the selected values.

11. When you have finished editing the element definition, export it. (For details, please see "Exporting Element Definitions," on page 33-16.)

33.5.3. Adding an Attribute to a Visual Definition

Each visual definition has a set of attributes. Some of them may be optional, so they are not specified as part of the factory definitions. You can add optional attributes to a visual definition.

To add an attribute to a visual definition:

1. If you are editing an entity, unit, or tactical graphic element definition, decide if you want to export the edited element definition to its own leaf file or to an existing file. (For details about the element definition work flow, please see "Work Flow Issues for Creating and Editing Element Definitions," on page 33-4.)
2. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
3. Select the page for the type of element definition that you want to edit (Figure 33-1).
4. If you want to export the edited element definition to an existing leaf file, use the Load and Replace procedure to reload the hierarchy. (This deletes all element definitions.) Then load the element definition leaf file. (For details, please see "Importing Element Definitions," on page 33-14.)
5. Select the visual definition that you want to edit.
6. Click the Add Attribute button (). The Add Visualizer Definition Attribute window opens. It lists the attributes that are available for the visual definition (Figure 33-6).

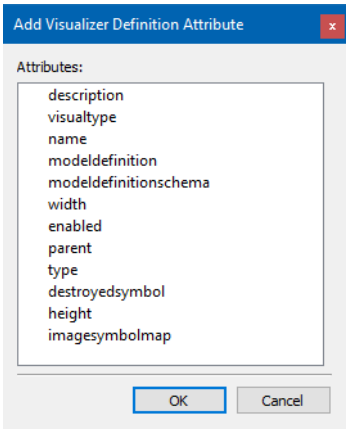


Figure 33-6. Add Visualizer Definition Attribute window

7. Select the attribute that you want to add.
8. Click OK. The attribute is added to the visual definition.

9. Specify a value for the attribute. This might require typing a value, selecting a value from a list, or selecting a value in another window.
10. When you have finished editing the element definition, export it. (For details, please see "Exporting Element Definitions," on page 33-16.)

33.5.4. Deleting an Attribute from a Visual Definition

You can delete individual attributes from a visual definition.

! The Delete button above the list of visual definitions deletes entire visual definitions. Do not click it thinking it will delete just the selected attribute.

To delete an attribute:

1. If you are editing an entity, unit, or tactical graphic element definition, decide if you want to export the edited element definition to its own leaf file or to an existing file. (For details about the element definition work flow, please see "Work Flow Issues for Creating and Editing Element Definitions," on page 33-4.)
2. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
3. Select the page for the type of element definition that you want to edit (Figure 33-1).
4. If you want to export the edited element definition to an existing leaf file, use the Load and Replace procedure to reload the hierarchy. (This deletes all element definitions.) Then load the element definition leaf file. (For details, please see "Importing Element Definitions," on page 33-14.)
5. Select the visual definition that you want to edit.
6. Select the attribute that you want to delete.
7. Click the Delete Attribute button ( ).
8. When you have finished editing the element definition, export it. (For details, please see "Exporting Element Definitions," on page 33-16.)

33.6. Deleting an Element Definition

VR-Vantage does not save changes to element definitions automatically. If you want to keep changes, you must export them to a file.



You cannot delete an element definition if it is being used.

To delete an element definition:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the page for the type of element definition that you want to delete (Figure 33-1).
3. If you are deleting an entity, unit, or tactical graphic element definition, use the Load and Replace procedure to reload the hierarchy. (This deletes all element definitions.) Then load the element definition leaf file. (For details, please see "Importing Element Definitions," on page 33-14.)
4. Select the element definition that you want to delete.
5. Click the Delete button (−) next to the Entity Definitions label.
6. Confirm the deletion.
7. Export the element definition back to the original element definition file. (For details, please see "Exporting Element Definitions," on page 33-16.)

33.7. Saving Element Definitions

Changes to element definitions do not get saved automatically. You must explicitly export the changes to a file. For details, please see "Exporting Element Definitions," on page 33-16.

33.8. Importing Element Definitions

You can import element definitions and replace the loaded element definitions. You can import element definitions and merge them with the loaded element definitions.

If you want to import entity, unit, or tactical graphics element definitions, you can import hierarchy files, leaf files, or both. If you import a hierarchy file, it removes all leaf file entries and replaces the previously loaded hierarchy.

When you import a leaf file, it is merged according to the criteria described in "Managing VR-Vantage Settings," on page 3-15.

You can also import or merge files in the legacy *.elem* format. If you import and replace entity, unit, or tactical graphics element definitions using an *.elem* file, the hierarchy and all element definitions are replaced with those in the elem file. If you merge an *.elem* file, the merge is additive only. Entities with names that already exist are skipped.



Interaction and smoke element definitions use the *.elem* format.

33.8.1. Importing a Hierarchy File

You can replace the current hierarchy with a different one by importing a hierarchy file or a file in the legacy *.elem* format.



When you import a hierarchy file, all leaf files are removed.

To import a hierarchy or element definition file and replace the existing definitions:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the page for the type of element definition you want to import.
3. Click the Import and Replace button (). The Import Element Definitions dialog box opens.
4. Select the hierarchy file or element definition file (extension *.elem* or *.hier*) you want to import.
5. Click Open.

33.8.2. Importing and Merging Element Definitions

You can import and merge element definitions from leaf files or legacy element definition (*.elem*) files.



You cannot import a leaf file unless a hierarchy is already loaded.

To merge element definitions:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the page for the type of element definition you want to import.
3. Click the Import and Merge button (). The Merge Element Definitions dialog box opens.
4. Select the element definition file (extension *.elem* or *.leaf*) you want to merge.
5. Click Open.

33.8.3. Importing Element Definitions from the Command Line

You can import and merge entity, unit, and tactical graphics element definitions when you start up VR-Vantage using the following command-line options:

- `--entityDefsDir` *`directory`* merges in all *.hier* and *.leaf* files in the specified directory. The directory must be *./appData/definitions/Entity/ElementDefinitions*, *./appData/definitions/Aggregate/ElementDefinitions*, or *./appData/definitions/TacticalGraphics/ElementDefinitions* or be subordinate to one of them.
- `--mergeHierarchyEntityDef` *`filename`* imports the specified *.hier* file.
- `--mergeLeafEntityDef` *`filename`* merges in the specified *.leaf* file.
- `--mergeHierarchyTacticalGraphicsDef` *`filename`* merges in the specified tactical graphics *.hier* file.
- `--mergeLeafTacticalGraphicsDef` *`filename`* merges in the specified tactical graphics *.leaf* file.
- `--mergeHierarchyUnitDef` *`filename`* merges in the specified unit *.hier* file.
- `--mergeLeafUnitDef` *`filename`* merges in the specified unit *.leaf* file.

These commands merge element definitions into the default element definitions, which are usually those in *./appData/definitions*. You can also specify that the element definitions in *./appData/definitions* not be loaded. In that case, only those files specified on the command line would get loaded. If none are specified, no element definitions would be loaded.

To specify that the element definitions in *./appData/definitions* not be loaded, start VR-Vantage with the `--noAppDataEntityDefs`, `--noAppDataUnitDefs`, or `--noAppDataTacticalGraphicsDefs` command-line options.

33.9. Exporting Element Definitions

Smoke and interaction element definitions get exported to a file with the extension *.elem*. Entity, unit, and tactical graphics element definitions, get saved to separate hierarchy (*.hier*) and leaf (*.leaf*) files.

You cannot save portions of the hierarchy. The entire hierarchy gets exported to file.

When the Visual Model Editors dialog box loads entity, unit, and tactical graphics element definitions it merges several files. When you export all element definitions, they do not get exported back to these files. They get exported to the file you specify. If you keep the new file in the same directory as the installed files, there might be conflicts between them when VR-Vantage loads element definitions. The element definition files are sorted into alphabetical order and loaded in that order. In the case of conflicts, the last file loaded wins. Therefore:

- If you are only adding or editing a few element definitions, save them to one or more new leaf files rather than overwriting one of the default leaf files.
- When you export to new leaf files, consider carefully how you name them.
- If you save all leaf files, consider removing the installed files from the definitions directory and just using your newly saved leaf file or save your file to a different directory and use command line options to load it instead of the installed files.

33.9.1. Exporting Element Definitions to *.elem* Files

When you export interaction or smoke element definitions, all element definitions get exported to one file with the extension *.elem*.

To export interaction or smoke element definitions:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the page for the type of element definition you want to export.
3. Click the Export All button (). The Export Element Definitions dialog box opens.
4. Specify a file to export to.
5. Click Save.

33.9.2. Exporting Entity, Unit, or Tactical Graphics Element Definitions

If you want to export entity, unit, or tactical graphics element definitions, you must export the hierarchy and leaf information separately. If the hierarchy has not changed, you do not need to export it.

To export a hierarchy file:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the page for the type of element definition you want to export.
3. Click the Export Hierarchy button (). The Export Element Definitions dialog box opens.
4. Select the hierarchy file you want to export to or type the name of a new file.
5. Click Save.

To export all definitions to a leaf file:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the page for the type of element definition you want to export.
3. Click the Export All Leaves button (). The Export Element Definitions dialog box opens.
4. Select the leaf file you want to export to or type the name of a new file.
5. Click Save.

To export selected definitions to a leaf file:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the page for the type of element definition you want to export.
3. Select the element definitions you want to export.
4. Click the Export Selected Leaves button (). The Export Element Definitions dialog box opens.
5. Select the leaf file you want to export to or type the name of a new file.
6. Click Save.



34. Mapping Object Types to Element Definitions

This chapter explains how to map element definitions to DIS and HLA entity types. For information about how to create element definitions, please see Chapter 33, *Element Definitions*. For information about mapping CIGI models and components, please see Chapter 41, *Mapping CIGI Models and Components*. Chapter 38, *Model Tutorials* walks you through the process of adding a model.

Introduction to Object Type Mapping 34-2
Work Flow Issues for Object Type Mappings 34-2
Adding an Object Type Mapping 34-4
Editing an Object Type Mapping 34-5
Filtering the Element Definition List 34-7
Deleting an Object Type Mapping 34-7
Saving Object Type Mappings 34-8
Importing Object Type Mappings 34-8
Exporting Object Type Mappings 34-9
Starting without Default Object Type Mappings 34-9
How VR-Vantage Maps DI-Guy Models 34-10

34.1. Introduction to Object Type Mapping

In order to visualize a scene element (entity, unit, detonation, and so on), VR-Vantage must map its object type enumeration to an element definition. This chapter explains how to do that.

You map object types to element definitions in the Simulation Object Type Mappings dialog box. It has pages for mapping:

- Cockpit displays.
- Detonation effects.
- Entities.
- Firing effects.
- Tactical graphics.
- Tactical smoke.
- Units.

The mapping process is the same for all object types. You can add new object type mappings, edit existing mappings, and delete mappings.



- You can map more than one element definition to an object type. If you do this, when entities of this type appear in a simulation, VR-Vantage randomly assigns one of the element definitions to the entity.
- You can map an element definition to more than one object type.

34.2. Work Flow Issues for Object Type Mappings

A default VR-Vantage installation has multiple object type mapping files. When you start VR-Vantage they all get loaded. When you open the Simulation Object Type Mappings dialog box, all of the mappings are listed. However, the dialog box does not keep track of which mappings come from which mapping file. When you create or edit object type mappings, VR-Vantage does not automatically save them. Therefore, you are responsible for saving new and edited mappings to the correct location.

The default object type mapping files are stored in *./appDate/definitions/<object_type>/TypeMap*, where <object_type> is Aggregate, Detonation, Entity, Fire, Smoke, or Tactical Graphics. Cockpit type mappings are in *./appData/settings/stealth/default_CockpitTypeMap.mhtx*. Detonation, fire, cockpits, and smoke have one default mappings file. The other object types each have four files (to accommodate VR-Forces, MAK's CGF application), which contain the following types of mappings:

- Aggregate level mappings. Mappings for simulation objects in VR-Forces aggregate-level scenarios.
- Entity level mappings. Mappings for simulation objects in VR-Forces entity-level scenarios.
- Base mappings. Mappings for simulation objects used by both aggregate-level and entity-level scenarios.
- Core mappings. Mappings for simulation objects for which VR-Forces does not have *.entity* files.

Before you add, delete, or edit a simulation object type mapping, consider which of the following work flows best meets your need:

- If you have created new model mappings, the preferred work flow is to export them to a new file. This is the easiest way to avoid overwriting other mappings and makes them easily portable to newer releases of VR-Vantage.

  If you want to add the new mappings to an existing mappings file:

  a. Use the Import and Replace option to load the mapping file you want to edit. (For details, please see "Importing Object Type Mappings," on page 34-8.)

  b. Add the new mappings.

  c. Export all changes back to the original mapping file.

- If you have edited existing mappings, you can export them to their original file or to a new file:
  - If you export to a new file, make sure that the file name is alphabetically later in alphabetical order than the original mapping file. This ensures that when the new mapping is loaded it will overwrite the original one.
  - If you want to export to the original mapping file:

    a. Use the Import and Replace option to load the mapping file you want to edit. (For details, please see "Importing Object Type Mappings," on page 34-8.)

    b. Make your changes.

    c. Export all changes back to the original mapping file.

- If you want to delete a mapping, you must delete it from the source mapping file. Follow the work flow for editing an original mapping file.

For more information about how to export mappings, please see "Exporting Object Type Mappings," on page 34-9.

34.3. Adding an Object Type Mapping

This procedure describes how to map an element definition to an object type enumeration. The element definition must already exist. The procedure is the same for any page on the Simulation Object Type Mappings dialog box.

To add an object type mapping:

1. Decide whether you want to save the new object type mapping to its own file or to an existing object type mapping file, as described in "Work Flow Issues for Object Type Mappings," on page 34-2.
2. Choose **Settings** → **Simulation Object Type Mappings**. The Simulation Object Type Mappings dialog box opens.
3. Select the Entity Mapping Settings page (Figure 34-1). The page shows a list of object type enumerations in the Entity Type column. The entity definition mapped to the object type is listed in the Entity Definition column.

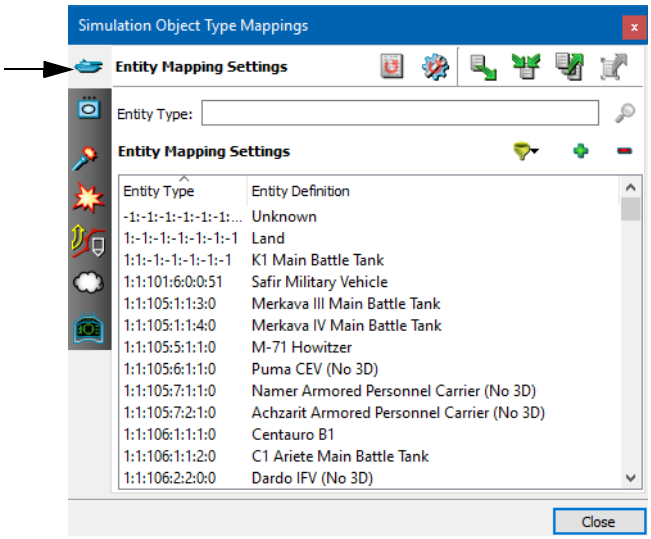


Figure 34-1. Entity Mapping Settings page

4. If you want to save the new object type mapping to an existing file, use the Import and Replace procedure to load the object type mapping file that you want to save to. (For details, please see "Importing Object Type Mappings," on page 34-8.)
5. Click the Add button (+). The Create New Entity Type Model Mapping dialog box opens (Figure 34-2). It lists a default object type enumeration.

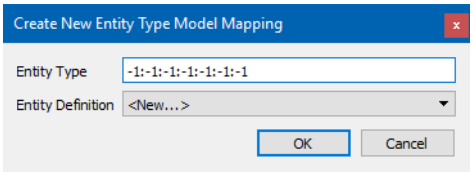


Figure 34-2. Create New Entity Type Model Mapping dialog box

6. In the Entity Type text box, type the enumeration you want to map to a model. (Separate each value with a colon(:).)
7. In the Entity Definition list, select the entity definition that you want to map to this object type.
8. Click OK.
9. The new model mapping is added to the list.
10. Export the new model mapping to a new or existing file. (For details, please see "Exporting Object Type Mappings," on page 34-9.)

34.4. Editing an Object Type Mapping

This procedure describes how to edit an object type mapping for a simulation object. The procedure is the same for any page on the Simulation Object Type Mappings dialog box.

To edit an object type mapping:

1. Decide whether you want to save the edited object type mapping to its own file or to an existing object type mapping file, as described in "Work Flow Issues for Object Type Mappings," on page 34-2
2. Choose **Settings** → **Simulation Object Type Mappings**. The Simulation Object Type Mappings dialog box opens.
3. Select the Entity Mapping Settings page (Figure 34-1).
4. If you want to save the edited object type mapping to an existing file, use the Import and Replace procedure to load the object type mapping file that you want to save to. (For details, please see "Importing Object Type Mappings," on page 34-8.)
5. Select the model mapping that you want to edit. If you know the entity enumeration that you want to edit, you can type the enumeration in the Entity Type text box and click the Search button (🔎). VR-Vantage highlights the entry for that enumeration, if it is present.
6. To change the enumeration associated with an element definition:
   a. In the Entity Type column, click the enumeration to make it editable.
   b. Change the values of the enumeration.
   c. Click someplace else in the window.

7. To change the element definition associated with an enumeration:
   a. Optionally, filter the list of available element definitions. (For details, please see "Filtering the List of Model Definitions," on page 32-8.)
   b. In the Entity Definition column, click the element definition to make it an editable list (Figure 34-3).
   c. Select the new element definition from the list.
   d. Click someplace else in the window.

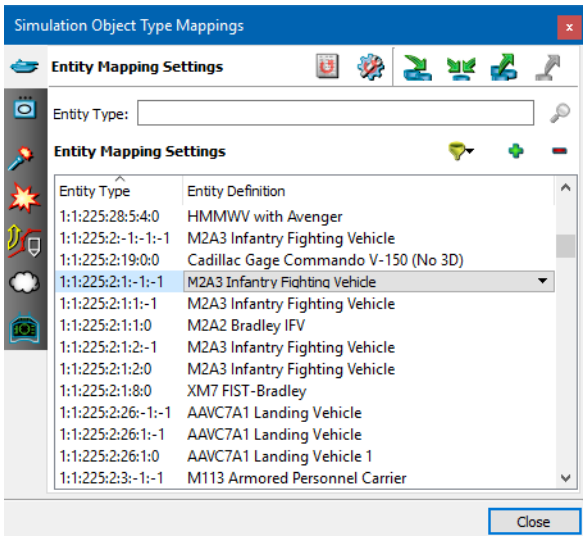


Figure 34-3. Editable entity definition

8. Export the model mapping to a new or existing file. (For details, please see "Exporting Object Type Mappings," on page 34-9.)

34.5. Filtering the Element Definition List

You can filter the list of element definitions that are available in the Entity Definition (Figure 34-3) list for mapping to entity types.

> Filtering does not change the list of simulation objects in the Entity Mapping Settings window. It just changes the list of simulation objects in the Entity Definition list when you change a specific mapping.

To filter the list of element definitions:

1. On any page of the Simulation Object Type Mappings dialog box, click the Filter button (▾). A menu is displayed. A check mark indicates the type of element definition that will be displayed when you make a element definition entry editable. (You may have to expand the hierarchy to see which elements are checked.)
2. Select the element definition type that you want to see in the list or select All to see all element definitions.

34.6. Deleting an Object Type Mapping

The procedure for deleting an object type mapping is the same for any page on the Simulation Object Type Mappings dialog box.

To delete an object type mapping:

1. Choose **Settings** → **Simulation Object Type Mappings**. The Simulation Object Type Mappings dialog box opens.
2. Select the Entity Mapping Settings page (Figure 34-1).
3. Use the Import and Replace procedure to load the object type mapping file that you want to delete a mapping from. (For details, please see "Importing Object Type Mappings," on page 34-8.)
4. Select the model mapping that you want to delete. If you know the entity enumeration that you want to edit, you can type the enumeration in the Entity Type text box and click the Search button (🔎). VR-Vantage highlights the entry for that enumeration if it is present.
5. Click the Delete button (−).
6. Export the model mapping to the file you imported. (For details, please see "Exporting Object Type Mappings," on page 34-9.)

34.7. Saving Object Type Mappings

New and changed simulation object type mappings do not get saved automatically. You must export them to the appropriate file. For details, please see "Exporting Object Type Mappings," on page 34-9.

34.8. Importing Object Type Mappings

You can import simulation object type mapping files. An imported file replaces or merges with the previously loaded mappings.

! When you merge type mappings, mappings from the file being merged in overwrite matching mappings that have already been loaded. If you have multiple models mapped to an object enumeration, all of the mappings are replaced by the new mappings.

To import simulation object type mappings:

1. Choose **Settings** → **Simulation Object Type Mappings**. The Simulation Object Type Mappings dialog box opens.
2. Select the page for the type of simulation object whose mappings you want to import.
3. To replace the current set of mappings, click the Import and Replace button (). The Import Mapping Settings dialog box opens.

   To merge mappings, click the Import and Merge button (). The Merge Mapping Settings dialog box opens.
4. Select the settings file you want to import.
5. Click Open.

34.9. Exporting Object Type Mappings

You must export all changes to simulation object type mappings. You can export all mappings to a file or you can select specific mappings and export them.

! If you have not explicitly loaded and replaced the list of object type mappings, do not export them to an existing file. The existing file will be replaced either by all of the mappings loaded in the Simulation Object Type Mappings dialog box or by the selected object type mappings.

To export all simulation object type mappings to one file:

1. Choose **Settings** → **Simulation Object Type Mappings**. The Simulation Object Type Mappings dialog box opens.
2. Select the page for the type of simulation object whose mappings you want to export.
3. Click the Export All button (). The Export Mapping Settings dialog box opens.
4. Specify the file to export to.
5. Click Save.

To export selected object type mappings to a file:

1. Choose **Settings** → **Simulation Object Type Mappings**. The Simulation Object Type Mappings dialog box opens.
2. Select the page for the type of simulation object whose mappings you want to export.
3. Click the Export Selected button (). The Export Mapping Settings dialog box opens.
4. Specify the file to export to.
5. Click Save.

34.10. Starting without Default Object Type Mappings

When you start VR-Vantage, you can specify that it not load the entity, unit, or tactical graphics type mappings in *./appData/definitions* or *./appData/settings/<application>*. To do this, start VR-Vantage with the `--noAppDataEntityTypeMap`, `--noAppDataEntityTypeMap`, or `--noAppDataEntityTypeMap` command-line option.

34.11. How VR-Vantage Maps DI-Guy Models

When VR-Vantage discovers a lifeform entity on the DIS or HLA networks, it chooses the animations to play for a DI-Guy based on the lifeform's speed, position, and similar attributes. However, if VR-Vantage receives a special DI-Guy state message (PDU) that dictates the lifeform's character, appearance, or animation, VR-Vantage will no longer try to guess and will expect to be told how to render that lifeform for the remainder of the simulation.



35. Wakes, 2D Icons, Cockpits, and Other Models

The previous chapters in this manual provide a general introduction to model schemas, model definitions, and element definitions. This chapter describes how to configure several specific types of visual models.

Configuring 2D Icons 35-2
Editing Font-Based 2D Icons 35-2
Using Images for 2D Icons 35-3
Adding Images for Simulation Object Icons 35-9
Adding New Cockpit Display Models 35-10
Installing a Cockpit DLL 35-11
Creating a Model Definition for a Cockpit 35-11
Configuring Wakes 35-14
Configuring Tidal Stream Wakes 35-16
Configuring Light Points 35-16
Configuring Light Point Type Models 35-18
Configuring Light Point Group Models 35-18
Adding a New Light Point Type 35-19
Adding Wind-based Controls to Models 35-20
Using Texture Atlases to Skin Models 35-21
Configuring Texture Mapping in the Model Definition 35-21
Metafiles for Texture Mapping 35-22
Flipping DDS Textures for a Model 35-23
Configuring SpeedTree Trees 35-24
Randomizing the Size of SpeedTrees 35-25
Displaying SpeedTrees in Planar Reflections 35-26
Best Practices for Creating Models for VR-Vantage 35-27
Compressing Model Files 35-28

35.1. Configuring 2D Icons

The 2D icons in VR-Vantage PVD and the Plan View mode of VR-Vantage Stealth are configured in visual definitions. Font-based icons are configured in the Military Symbol Icon visual definition. Image-based icons use the Entity Image Symbol visual definition.

35.1.1. Editing Font-Based 2D Icons

By default, 2D icons are configured in the Military Symbol Icon visual definition, which is part of a simulation object's entity definition. This visual definition uses font-based icons. You can change a simulation object's icon by changing the fillGlyph and outlineGlyph attributes.

To change a simulation object's icon:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Entity Definition Editor page (Figure 35-1).
3. Select the simulation object whose icon you want to edit.
4. In the Model Sets list, select 2D Icons.
5. To change the fill glyph:
   a. Under the Military Symbology Icon visual definition, select the fillGlyph attribute (Figure 35-1).

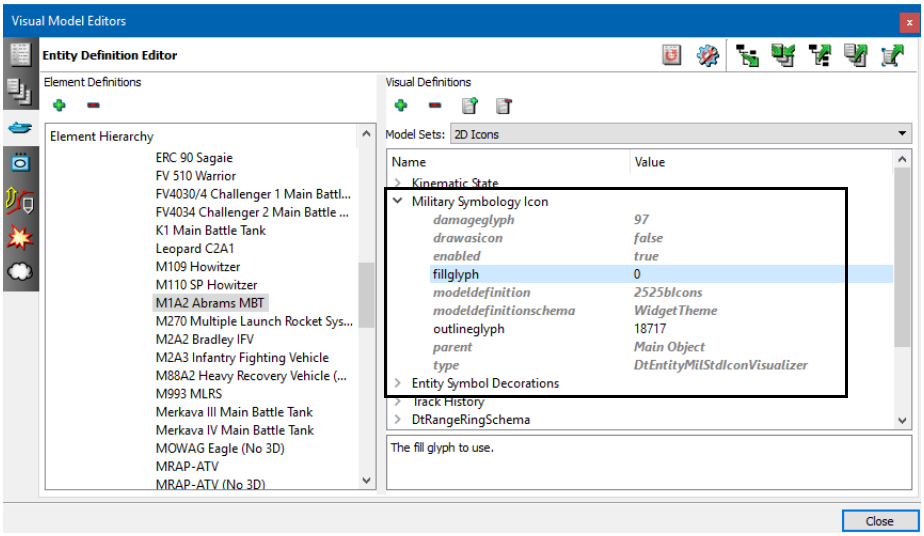


Figure 35-1. Military Symbol Icon visual definition

   b. Click the value in the Value column. The Choose Font Glyph dialog box opens (Figure 35-2).

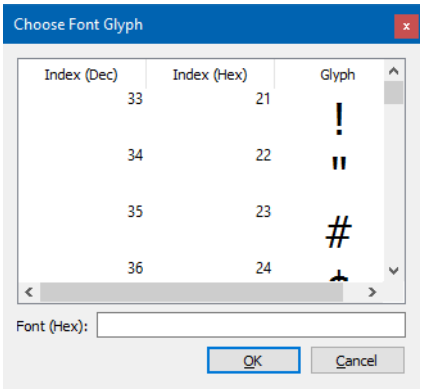


Figure 35-2. Choose Font Glyph dialog box

c. Select the glyph with the fill pattern you want to use. Click OK.

6. To change the outline glyph, follow the same procedure as changing the fill glyph.

35.1.2. Using Images for 2D Icons

By default, VR-Vantage uses TrueType fonts for 2D icons. You can replace the font-based icons with images using the Entity Image Symbol visual definition.



If you configure images for icons, be sure to save the element definitions to a file other than the default settings. If you revert to factory defaults or otherwise switch back to font-based icons and you have not saved the image mappings, you will not be able to recover them without repeating the entire mapping process again.

To use images for entity icons:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Entity Definition Editor page (Figure 35-1). (If you want to change the icon for a unit, select the Unit Definition Editor page. To change the icon for a waypoint, select the Tactical Graphics page.)
3. In the Model Sets list, select 2D Icons.
4. In the Element Definitions list, select Entity. (If you are editing unit images, in the Element Definitions list select Aggregate. If you are editing tactical graphics images, in the Element Definitions list select Waypoint. For all other tactical graphics, skip to step 8.)
5. In the Visual Definitions list, select Military Symbology Icon (Figure 35-3).

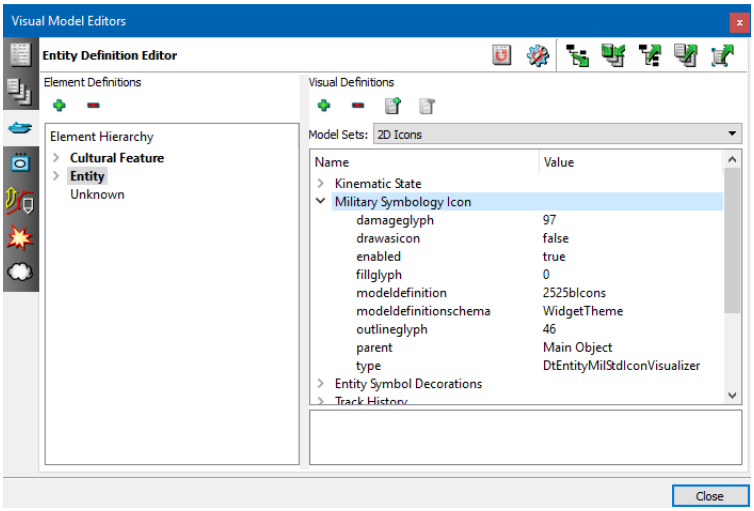


Figure 35-3. Entity element definition with Military Symbology Icon visual definition

6. Click the Delete button (−). The visual definition gets deleted.
7. Click the Add button (+). The Create Visualizer Definition window opens. It displays a list of visual definitions (Figure 35-4).

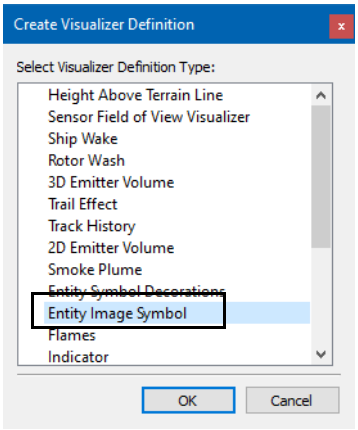


Figure 35-4. Create Visualizer Definition window

8. Select Entity Image Symbol. (If you are configuring images for units, select Aggregate Image Symbol; for tactical graphics, select Tactical Graphic Image Symbol.)
9. Click OK.
10. In the Visual Definitions list, select Entity Image Symbol (Figure 35-5).

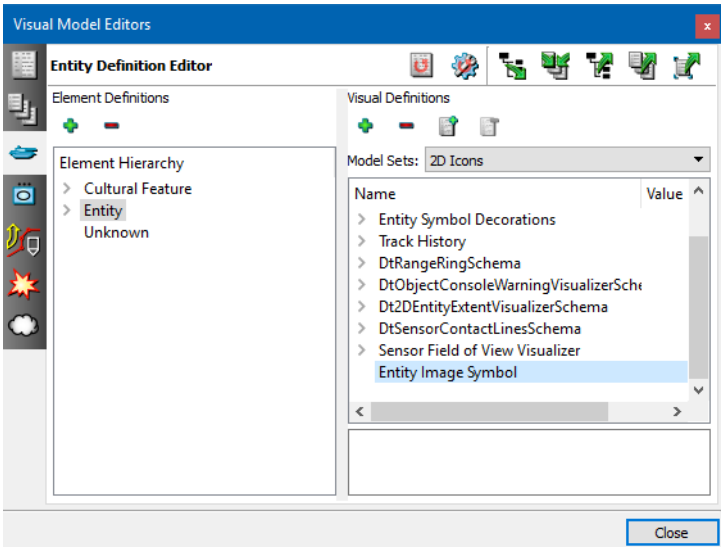


Figure 35-5. Entity element definition with Entity Image Symbol visual definition

11. Click the Add Attribute button (). The Add Visualizer Definition Attribute window is displayed (Figure 35-6).

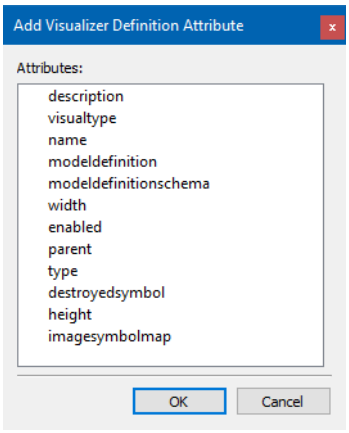


Figure 35-6. Add Visualizer Definition Attribute window

12. In the attribute list, select **parent**.
13. Click OK. The **parent** attribute is added to the visual definition.
14. Click the value field to make it editable.
15. Type Main Model and press **Enter**.
16. Click the Add Attribute button.
17. In the attribute list, select **type**. The value is filled in automatically.
18. Click the Add Attribute button.

19. In the attribute list, select modeldefinitionschema. The Choose Supported Model Definition Schema dialog box opens (Figure 35-7).

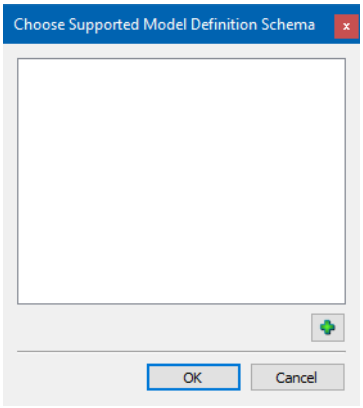


Figure 35-7. Choose Supported Model Definition Schema dialog box

20. Click the Add button. A list is added to the dialog box.
21. In the list, select WidgetTheme (Figure 35-8).

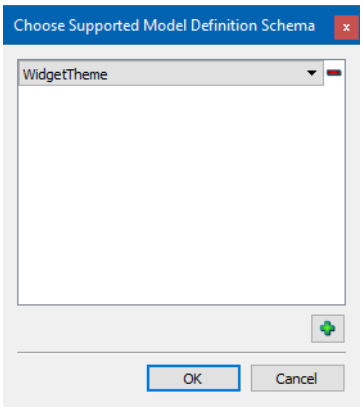


Figure 35-8. Completed Choose Supported Model Definition Schema dialog box

22. Click OK.
23. Click the Add Attribute button.
24. In the attribute list, select imagesymbolmap. The 2D Symbol Image Mapper dialog box opens (Figure 35-9).

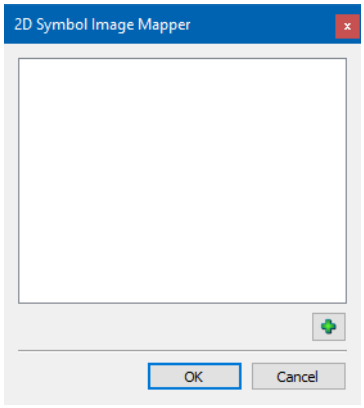


Figure 35-9. 2D Symbol Image Mapper dialog box

25. Click the Add button. A list is added to the dialog box. It lists the forces for which you can map images. and the images that you can apply to a particular force
26. Select a force and an image from the lists (Figure 35-10).

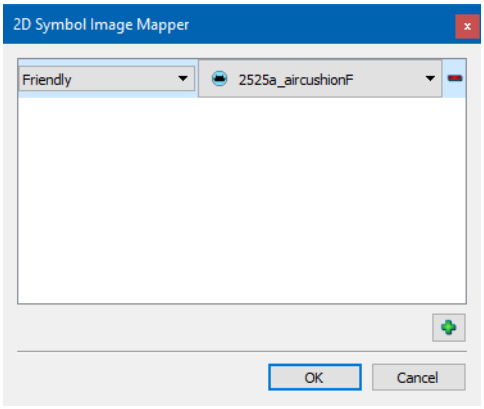


Figure 35-10. 2D Symbol Image Mapper dialog box with force chosen

27. Add an entry for each force that you want to represent with an image.
28. Click OK.
29. Click the Add Attribute button.
30. In the attribute list, select **destroyedsymbol**. The **destroyedsymbol** attribute is added to the visual definition.
31. In the **destroyedsymbol** list, select Destroyed. The completed visual definition should look similar to Figure 35-11.

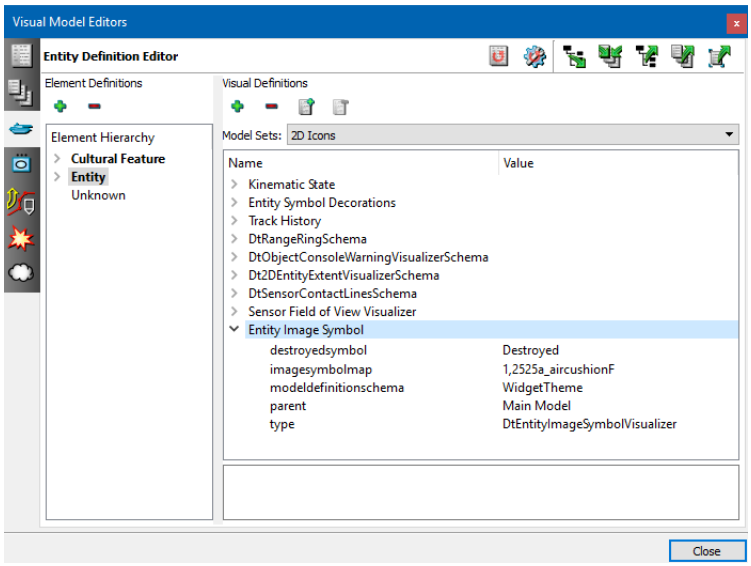


Figure 35-11. Entity Image Symbol visual definition

The settings for the Entity Image Symbol visual definition are inherited by each entity in the hierarchy from the base Entity definition. Unless you configure the various entity categories, all entities use the images you configured for the Entity category. Therefore, at a minimum, change the **imagesymbolmap** values for each force at each generic object type, such as Fixed Wing and Life Form, to use the proper image for that object type. Optionally, you can specify an image for each entity definition at the leaf level.



When you edit the Entity Image Symbol visual definition attributes, be sure that you select the 2D Icons model set.

35.1.3. Adding Images for Simulation Object Icons

VR-Vantage includes a representative set of MIL-STD 2525a images. They are in *./data/images*. If you want to add your own images, you can do so by adding a new model definition or changing the image mapping for an existing model definition. The new images are automatically added to the list of images for the imagesymbolmap in the Entity Image Symbol visual definition.

To add images for entities:

1. Copy the new images to *./data/images*.
2. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
3. Select the Model Definition Editor page (Figure 32-1).
4. In the Schema list, select WidgetTheme.
5. If you want to change the image used for an existing model definition:
   a. Select the model definition that you want to change, for example, 2525a_groundF.
   b. Change the file specified for the backgroundImage attribute.

   If you want to specify an image for a new model definition:
   a. Create a new model definition, as described in "Creating and Editing Model Definitions," on page 32-2. It is recommended that you copy an existing model definition for a 2525a symbol.
   b. Specify the image that you want to use for the new model definition.

35.2. Adding New Cockpit Display Models

VR-Vantage supports cockpit displays created with GL Studio. A cockpit is implemented through a Reusable Software Object (RSO) DLL. A cockpit's values are changed through updaters. An updater is associated with a cockpit attribute through the cockpit model definition. Figure 35-12 shows the model definition for a cockpit.

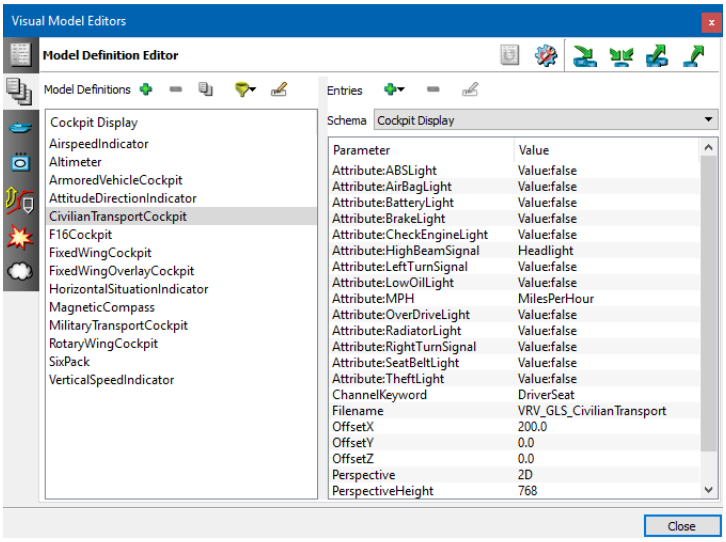


Figure 35-12. Cockpit model definition

Cockpit updaters are associated with an attribute name that is known to the RSO. The value you provide for a cockpit-specific attribute is the name of an updater for dynamically changing values or a constant value. Constant values are commonly used to configure a cockpit when it is loaded. VR-Vantage provides a set of commonly used updaters. You can add additional updaters or override existing updaters through a plug-in.



The GL Studio license included with VR-Vantage only covers the cockpits delivered with the product. If you add your own cockpits, you will need GL Studio run-time licenses.

The general procedure for adding a new cockpit model is:

1. Install the cockpit's DLL.
2. Create a model definition for the cockpit.
3. Map object types to the model definition, as described in "Adding an Object Type Mapping," on page 34-4.

35.2.1. Installing a Cockpit DLL

If you create your own cockpit, it is implemented as an RSO DLL that you must add to the VR-Vantage file structure.

➤ To install a new cockpit, place the DLL for the cockpit in *./data/Huds* and create a model definition for it.

35.2.2. Creating a Model Definition for a Cockpit

Cockpit models are configured using schemas and model definitions, just like any other model in VR-Vantage. This section provides additional details specific to cockpits.

To create a model definition for a cockpit:

1. Add a new model definition, as described in "Creating a Model Definition," on page 32-5. Be sure to select Cockpit Display as the schema. A new model definition is added to the display (Figure 35-13). The message window lists the parameters that you must add. These are the parameters defined by the schema.

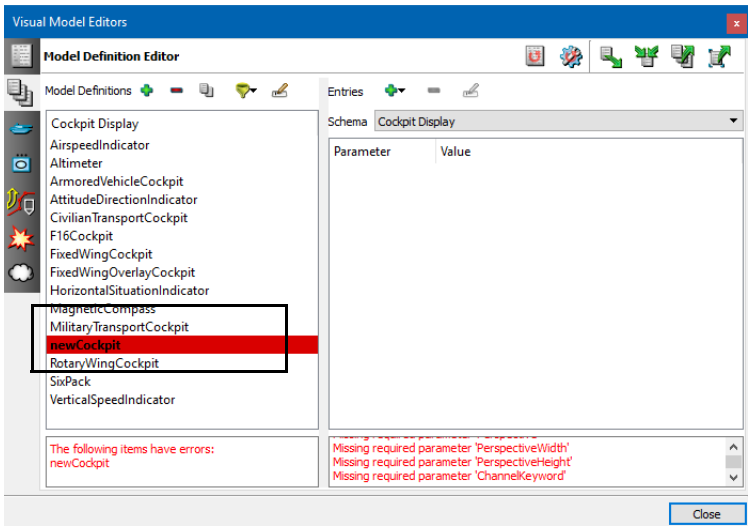


Figure 35-13. New cockpit model definition

2. Add the required parameters, as described in "Adding Parameters to a Model Definition," on page 32-8. Table 35-1 describes the required parameters.

Table 35-1: Required parameters for cockpit models

Parameter	Description
Filename	The name of the DLL without the file extension.
RSOName	The name of the class to be loaded from the DLL. Default: *filename*Class.
Perspective	2D or 3D. This parameter refers to the perspective of the cockpit display, not the terrain.
PerspectiveWidth PerspectiveHeight	Adjusts the orthogonal projection for a 2D cockpit. Adjusts the perspective projection for a true 3D cockpit.
ChannelKeyword	A keyword that tells a channel to display any object that has the keyword in its definition. Default: showCockpits. (This keyword means that any channel that has the showCockpits keyword will show cockpits.)
Offset-X Offset-Y Offset-Z	Moves the cockpit offset around the screen. The OffsetX value shifts the cockpit left or right. The OffsetY value shifts it up or down. OffsetZ shifts a 3D cockpit forward and back. It has no effect on 2D cockpits.

3. Add the parameters required by the cockpit RSO. To do this, add an Attribute parameter, as follows:
   a. Add an Attribute parameter, as described in "Adding Parameters to a Model Definition," on page 32-8.
   b. Double-click the parameter to make its name editable.
   c. Add a colon and the name of an RSO attribute that you need to map. For example, Attribute:Heading.
   d. In the Value column, click the value to make it editable. Type the name of an updater or specify a constant value in the format Value:*value*. You can specify an updater that you have added through a plug-in or one of the updaters provided with VR-Vantage, as described in Table 35-2.

4. Map an object type to your definition, as described in "Adding an Object Type Mapping," on page 34-4.

Table 35-2: Updaters provided

Updater	Description
AOA	Angle of Attack. Returns the pitch of the entity clamped (5 to 30 degrees).
FeetAGL	Feet Above Ground Level. Returns the feet above the terrain (or buildings below).
FeetMSL	Feet Above Mean Sea Level. Returns the altitude above the sea level.
FeetPerMinute	Feet Per Minute. Returns the climb rate.
KilometersPerHour	Speed of the entity in kilometers per hour.
MilesPerHour	Speed of the entity in miles per hour.
KnotsPerHour	Speed of the entity in nautical miles per hour.
Pitch	Returns the pitch of the entity clamped (-180 to 180 degrees).
Roll	Returns the roll of the entity clamped (-180 to 180 degrees).
Yaw	Returns the yaw of the entity clamped (0 to 360 degrees).
Headlight	Returns the headlight state of the entity as true or false.
Value	This updater sends any value to the cockpit once and then never again. It is used to configure initial setup and to turn display components on or off.

35.3. Configuring Wakes

When dynamic ocean is enabled, VR-Vantage simulates wakes on surface entities. Wakes are configured in the Ship Wake visual definition of an entity's element definition. If you add your own model, you may want to change the default configuration values for its wake so that it more closely matches the dimensions and velocity of the object type being modeled.

To configure a ship wake:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Entity Definition Editor page (Figure 35-14).

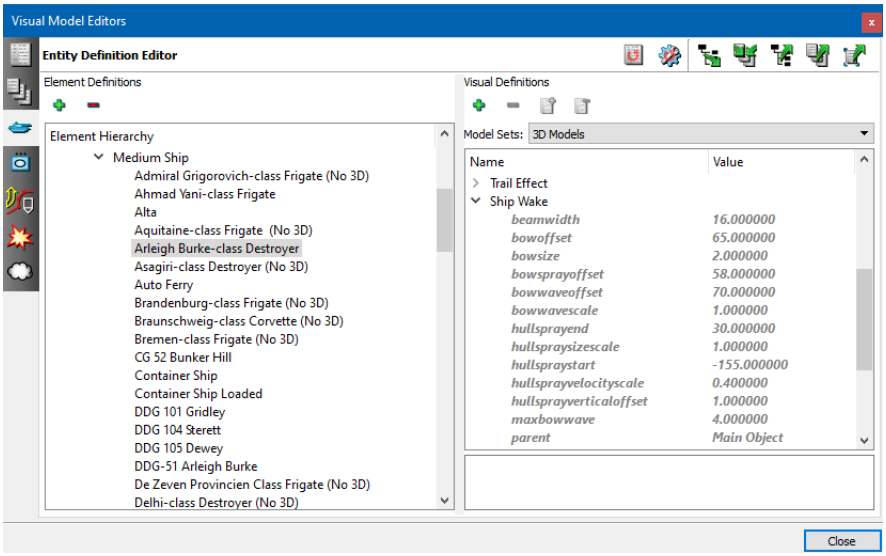


Figure 35-14. Ship Wake visual definition

3. In the Element Hierarchy, expand the Entity list to the Surface level and select the entity that you want to configure.
4. In the Visual Definitions list, select and expand the Ship Wake visual definition.
5. Edit the parameters. Table 35-3 describes each parameter.

Table 35-3: Ship Wake visual definition attributes

Parameter	Description
beamwidth	The width of the ship at its widest point.
bowsize	The size of the bow in world units. This affects the wavelength of the bow wave and the initial spread of spray particles at the bow. Use 0 for a pointy bow.
bowsprayoffset	An offset from the ship position from which to emit spray particles.

Table 35-3: Ship Wake visual definition attributes

Parameter	Description
bowwaveoffset	An offset from the ship position, along the direction of travel, at which bow waves will originate.
bowwavescale	A scaling factor for the bow wave's amplitude at the bow of the ship.
draft	The depth of the hull underwater.
hullsprayend	The offset from the ship at which hull spray effects end.
hullspraysizescale	A scaling factor applied to the hull spray particles.
hullspraystart	The offset from the ship at which hull spray effects begin.
hullsprayvelocityscale	The initial velocity of hull spray particles as a percent of the ship velocity (0-1).
hullsprayverticaloffset	A vertical offset to the starting point of new hull spray particles.
loddistance	The distance from the camera beyond which wakes are not drawn. Cannot be greater than the value set by max-kelvin-wake-distance in the MAK Overrides section of *./data/Environment/Ocean/Triton.config*.
maxbowwave	The maximum amplitude of the bow wave, or -1.0 for unbounded.
numhullsprays	The number of spray particles emitted periodically along the hull of the ship.
parent	The visualizer's parent.
propwashenabled	Enable or disable propeller backwash effects.
propwashoffset	An offset from the ship position from which to generate propeller backwash effects.
shiplength	A value, that in combination with the entity's speed, is used to determine the length of the bow wake.
sprayenabled	If enabled, the wake emits spray particles.
spraysizescale	A scaling factor applied to the size of the bow spray particles and their random spread.
sprayvelocityscale	A scaling factor for spray effects at the bow of the ship. This is applied to the initial velocity of the spray particles.
sternwaveenabled	Enable or disable the stern wake.
sternwaveoffset	An offset for the stern wakes.
type	The type of visualizer.
wakeoffset	An offset to the model's origin that all other offsets are relative to. Changing this offset lets you move all of the wake effects in tandem if the entity model changes.

35.3.1. Configuring Tidal Stream Wakes

Tidal stream wakes are supported as stand-alone models and are automatically attached to streamed buoys and beacons. Model definitions for wakes to be used with each of the supported buoy and beacon models are included (BuoysPillarWake, BeaconsLatticeWake, and so on). You may want to edit them. We recommend wake lengths of at least 20m for best results. (Some of the model definitions specify values less than that.)

By default, VR-Vantage starts with a tidal stream current speed of 0. To see tidal stream wakes, change the speed on the Scene Settings dialog box, Environment Conditions Settings page. For more information, please see "Configuring Marine Conditions," on page 11-12.

You can also change the speed and direction of the tidal stream current using a CIGI Component Control or Short Component Control message. The message uses a component class of Regional Sea Surface(4), a component ID of 1, and two float component data fields — the first is the tidal stream current speed (m/s), and the second is the current direction (0-2PI radians, clockwise from North). The Region ID is ignored.

35.4. Configuring Light Points

VR-Vantage supports configuration of light points using light point types. A light point type is an optional configuration for a light point that allows consistent behavior across models and terrains.

Light point type settings are configured in *./appData/importConfig/Lights_DefaultValues_3_2.csv*. You can edit this file and add additional light point types. Table 35-4 describes the parameters specified in *Lights_DefaultValues_3_2.csv*.

Table 35-4: Light point type settings

Parameter	Descriptions
Code	Integer which specifies the light point type for this row.
Description	String which specifies a human readable description for the light point type. It is currently unused in VR-Vantage.
Intensity	Floating point number which specifies the intensity of the light point. Usually (0-1.0) but does support values higher than 1.0.
Red	Floating point number which specifies the red color value for the light point. Usually (0-1.0) but does support values higher than 1.0.
Green	Floating point number which specifies the green color value for the light point. Usually (0-1.0) but does support values higher than 1.0.
Blue	Floating point number which specifies the blue color value for the light point. Usually (0-1.0) but does support values higher than 1.0.

Table 35-4: Light point type settings

Parameter	Descriptions
Directionality	String which specifies the directionality of the light point. Valid strings: ◆ Omni-directional - Light point will have the same intensity when viewed from any direction. ◆ Directional - Light point will have the regularly specified intensity when viewed from the within the lobe and the residual intensity when viewed outside the lobe. ◆ Bidirectional - Light point will have the regularly specified intensity when viewed from the lobe or another lobe in the opposite direction and the residual intensity when viewed outside the two lobes.
Horizontal Lobe	Floating point value in degrees for the angle that describes the horizontal lobe away from the light point's direction. Only used when the Directionality is set to Directional or Bidirectional. (0.0 - 360.0)
Vertical Lobe	Floating point value in degrees for the angle that describes the vertical lobe away from the light point's direction. Only used when the Directionality is set to Directional or Bidirectional. (0.0 - 360.0)
Residual Intensity	Floating point value which specifies the intensity of the light point when the light point is viewed from a direction that is outside the lobe. Only used when the Directionality is set to Directional or Bidirectional. Usually (0.0 - 1.0) but does support values greater than 1.0.
Frequency	Floating point value which specifies the number of times per second that the light point should blink. 0.0 will turn off blinking.
Duty Cycle	Floating point value which specifies the fraction of time that the light point should be on. (0.0 - 1.0)
Min Pixel Size	Positive floating point value which specifies the minimum number of pixels for the light point's diameter.
Max Pixel Size	Positive floating point value which specifies the maximum number of pixels for the light point's diameter.
Max Visible Distance	Positive floating point value which specifies the maximum distance in meters that the light point should be visible. This will only have an effect if the light point is bright enough to still be visible from this distance.
Radius	Positive floating point value which specifies the size of the light point in meters.

35.4.1. Configuring Light Point Type Models

You can configure light point types for a particular light point in the OpenFlight model using Creator in a couple of different ways.

The simplest way is to use the light point palette manager in Creator to specify the Appearance name for a particular light point appearance index. The name can either be a CDB type name, which can be set using the **Edit → CDB Light point Palette Names** menu or it can be an integer value that is one of the Codes specified in the *Lights_DefaultValues_3_2.csv*.

The CDB names are in *./appData/importConfig/Lights.xml*, which has a hierarchy defined using XML. For example the light point type with code 1 would have the CDB name of \Light\Platform based on the hierarchy in *Lights.xml*.

35.4.2. Configuring Light Point Group Models

You can group light points into light point groups, which let you control the color of a group of light points. These groups can group light points for things like runways, airports, parks, and cities.

Light point groups are automatically generated for the procedural airports and the CDB database airports using the runway and airport names.

A terrain that has an OpenFlight airport generated by the Creator Airport or Runway wizards will also automatically have light point groups if the airport name follows the format "Airport - `{name here}`" where `{name here}` specifies the airport name.

The default grouping name style for airports and runways is `{Airport Name}:{Runway Name}:{optional type}` where `{optional type}` is Approach or Runway.

Another way to set the light point group is to use comments on the light point group node. If the comment contains `@dis light_group {group name}` where `{group name}` is the name for this light point group, then all light points under that particular light point group node will be in that light point group.

You can set a light point group color using the VR-Vantage view control messages, CIGI component control messages, or the VR-Vantage developers toolkit.

35.4.3. Adding a New Light Point Type

You can add new light point types to the default configuration files. To add a new light point type, you add an entry to the end of *./appData/importConfig/Lights_DefaultValues_3_2.csv* and to *./appData/importConfig/Lights.xml*. It can be added anywhere in the *Lights.xml* hierarchy, but it is recommended that you add it to the end of the `Lights` element.

The following example shows how to add light point type 488 to the configuration.

1. Add the following to the bottom of *Light_DefaultValues_3_2.csv*:

```
488, Red Custom,0.6,1,0,0,Omnidirectional,57.2958,57.2958,0,0,0.5,1,20,
  100000,0.2
```

2. Change the bottom of the *Lights.xml* from:

```
  </Light>
</Lights>
```

   To:

```
  </Light>
  <Light type="Red_Custom" code="488">
    <Description>Red Custom Light Type</Description>
  </Light>
</Lights>
```

3. Create a model that uses the new custom light point type.

35.5. Adding Wind-based Controls to Models

VR-Vantage can simulate the movement of flags and windsocks based on the current simulated wind speed and direction on terrains. It can also simulate movement of flags and windsocks on models added through connections or programmatically (this is not supported for props or external references). These features require models with appropriate switches and articulations, as well as model-specific or terrain-specific metafiles.

For generic information about the metafile format and syntax, please see section 8.6, Transforming Nodes Programmatically, in VR-Vantage *Developers Guide*.

The metafiles for windsocks and flags have sections that configure wind control of nodes, as follows:

- Control switch nodes based on wind speed.
- Control the heading (Z-Axis) of articulating (degree of freedom) nodes based on wind direction.

The metafile used for wind features supports two features — WindSwitchPart and WindDirPart.

- WindSwitchPart. The WindSwitchPart feature is used to change the state of a switch node based on the current wind speed. The flag models used by MAK include a switch with three states. This feature has the following parameters:
  - **WindSpeed**. A list of wind speeds (in km/h).
  - **WindState**. A list of the switch state values that correspond to the values in the **WindSpeed** parameter.

  The number of items in each list should be the same (and only contain switch states supported by the node specified in the `Linkage` entry.
- WindDirPart. The WindDirPart feature is used to modify the heading value (Z-Axis) of an articulation (degree of freedom) node based on wind direction. It has the following parameters:
  - **RotOffsetDeg**. Allows a rotation offset to be applied.
  - **RotDir**. Indicates if the node should be rotated in a clockwise or counter-clockwise direction.

Metafiles used to configure flags and windsocks include *./data/Terrain/Hawaii/Geometry/OpenFlightAlaMonana/AlaMoanaGeocentric.meta*, *./data/Structures/Poles/FlagPole/FlagPoleFlag.meta*, and *./data/Structures/Poles/WindSock.meta*.

35.6. Using Texture Atlases to Skin Models

When VR-Vantage renders a vehicle model it takes into account information about the geometry of the vehicle, which specifies the dimensions, shape and so on, and textures, which specify the surface characteristics (in other words, how the vehicle looks.) If several models have the exact same geometry, but only differ in how they look (their textures), you can save disk space and some processing time by using the same geometry file for all of the models and a texture atlas to specify the textures to use for each model. (A texture atlas is a composite from multiple textures with the corresponding 3D models having uv coordinates shifted.)

For example, VR-Vantage has four different versions of the Airbus 320 jet, one each for American Airlines, British Airways, Lufthansa, and Singapore Airlines. Instead of having four geometry files with associated textures, you can use one geometry file for all four models and reference the appropriate texture in a texture atlas.

Texture skinning requires a MEDF (geometry) file and a texture atlas that has been prepared for the entities that will share the texture atlas. The portions of the texture for a given model are accessed via UV offsets into the texture atlas. This information is stored in a metafile (*.meta*) that references a *.dtxc* file. You must also specify the metafile in the model definition for each model that is to use it and a configName parameter that is used to identify the correct texture to use.

35.6.1. Configuring Texture Mapping in the Model Definition

To use a texture atlas for a model, you must provide values for the metaFileName and configName parameters in the model definition.

To configure texture mapping in a model definition:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Model Definition Editor page (Figure 32-1).
3. Optionally, filter the model definitions in the list. (For details, please see "Filtering the List of Model Definitions," on page 32-8.)
4. Select the model definition that you want to edit.
5. Click the Add Parameter button (). A list of parameters is displayed.
6. Select metaFileName. The metaFileName parameter is added to the model definition.
7. In the value column specify the location of the metafile you want to use for this model.
8. Click the Add Parameter button and select configName. The configName parameter is added to the model definition.
9. In the value column type a string that VR-Vantage should look for in the meta file to identify the texture pattern to use.

10. Export the model definitions.

35.6.2. Metafiles for Texture Mapping

VR-Vantage uses metafiles to configure texture skinning. For conceptual information about metafile format and syntax, please see section 8.6, Transforming Nodes Programmatically, in VR-Vantage *Developers Guide*.

A metafile is associated with a terrain or model in one of the following ways:

- Place it in the same directory as the terrain or model's MEDF file. The meta file must have the same name as the MEDF file, but with a *.meta* extension.
- Specify it in the **metaFileName** parameter of the model definition used to import the model.

VR-Vantage metafiles support two features that are relevant to texture mapping:

- UvAtlasMapper. Defines a texture atlas for texture skinning.
- SwitchStateMapper. Specifies the child of a switch node to use. For example you could have an M1A1 with mine rollers, with mine plows, or DU armor, depending on the switch node used.

The following example shows the meta file for the Airbus 320 jet *Airbus-320a.meta*:

```
{
  "NodeFeatures": [
    {
      "Linkage": {
        "Comment": "Traversal with match up this feature based on
          NodeName match OR AnnotationName match",
        "NodeName": "",
        "AnnotationName": "@dis annotation texture"
      },
      "FeatureData": [
        {
          "FeatureName": "UvAtlasMapper",
          "AtlasFile": "$(DATA_DIR)\\Vehicles\\FixedWing\\Airbus320a\\
                       Airbus320a.dtxc",
          "Mappings": [
            {
              "ConfigName": "AmericanAirlines",
              "AtlasKeys": {
                "Pattern": "AmericanAirlines"
                }
            },
            {
              "ConfigName": "BritishAirways",
              "AtlasKeys": {
                "Pattern": "BritishAirways"
              }
            },
            {
              "ConfigName": "Lufthansa",
              "AtlasKeys": {
                "Pattern": "Lufthansa"
              }
```

```
            },
            {
              "ConfigName": "SingaporeAirlines",
              "AtlasKeys": {
                "Pattern": "SingaporeAirlines"
              }
            }
          ]
        }
      ]
    }
  ]
}
```

When VR-Vantage loads an Airbus 320, the **metaFileName** parameter in the model definition points it to the *Airbust320a.meta* file. The **configName** parameter in the model definition specifies which texture to use, for example, AmericanAirlines. VR-Vantage reads *Airbust320a.meta* and associates the **configName** string, with the pattern value, also AmericanAirlines. Then it looks in *Airbus320a.dtxc* and gets the UV offset of AmericanAirlines so it knows which textures to use from the texture atlas.

The UV offsets for these models are configured in *Airbus320a.dtxc*. (The file contains explanatory comments.)

```
texture,COLPAT,AmericanAirlines,0.0,0.0
texture,COLPAT,Lufthansa,0.0,0.189612
texture,COLPAT,SingaporeAirlines,0.0,0.344823
texture,COLPAT,BritishAirways,0.0,0.500039
```

35.7. Flipping DDS Textures for a Model

If a model is displayed upside down because its DDS textures are not what VR-Vantage expects, you can specify that the model be flipped. (For more information about DDS textures, please see "Displaying DDS Textures Correctly," on page 29-6.)

To flip DDS textures for a model:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Model Definition Editor page (Figure 32-1).
3. Optionally, filter the model definitions in the list.
4. Select the model for which you want to change the DDS parameter.
5. If the model has the **flip DDS textures** parameter in its parameters list, set it to 1.

   If the model does not have the **flip DDS textures** parameter in its parameters list, click the Add Parameter button () to see if this is a valid parameter for this model. If it is, you can add the parameter and set it to 1.
6. Export the updated setting.

35.8. Configuring SpeedTree Trees

SpeedTree trees are high resolution models of trees. To improve performance, you can set clipping plane and level-of-detail (LOD) values to vary the resolution of the trees placed in the scene based on their distance from the observer. For details about clipping planes, please see "Setting the Clipping Planes," on page 6-14. Table 35-5 describes the settings you can change.

Table 35-5: SpeedTree settings

Parameter	Description
Clipping distance	The distance at which trees are clipped out of the scene.
Fade from High LOD to Low LOD	The range (near to far) in which high LOD trees are gradually replaced with low LOD trees.
Fade from Low LOD to Billboard	The range (near to far) in which low LOD trees are gradually replaced with billboard trees. This range is usually greater than the range for high LOD to low LOD.
Ambient Scalar	Multiplies the built-in ambient lighting value in each SpeedTree by the specified amount.
Diffuse Scalar	Multiplies the built-in diffuse lighting value in each SpeedTree by the specified amount.
Specular Scalar	Multiplies the built-in specular lighting value in each SpeedTree by the specified amount.

To configure SpeedTree trees:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the SpeedTree Settings page (Figure 35-15).

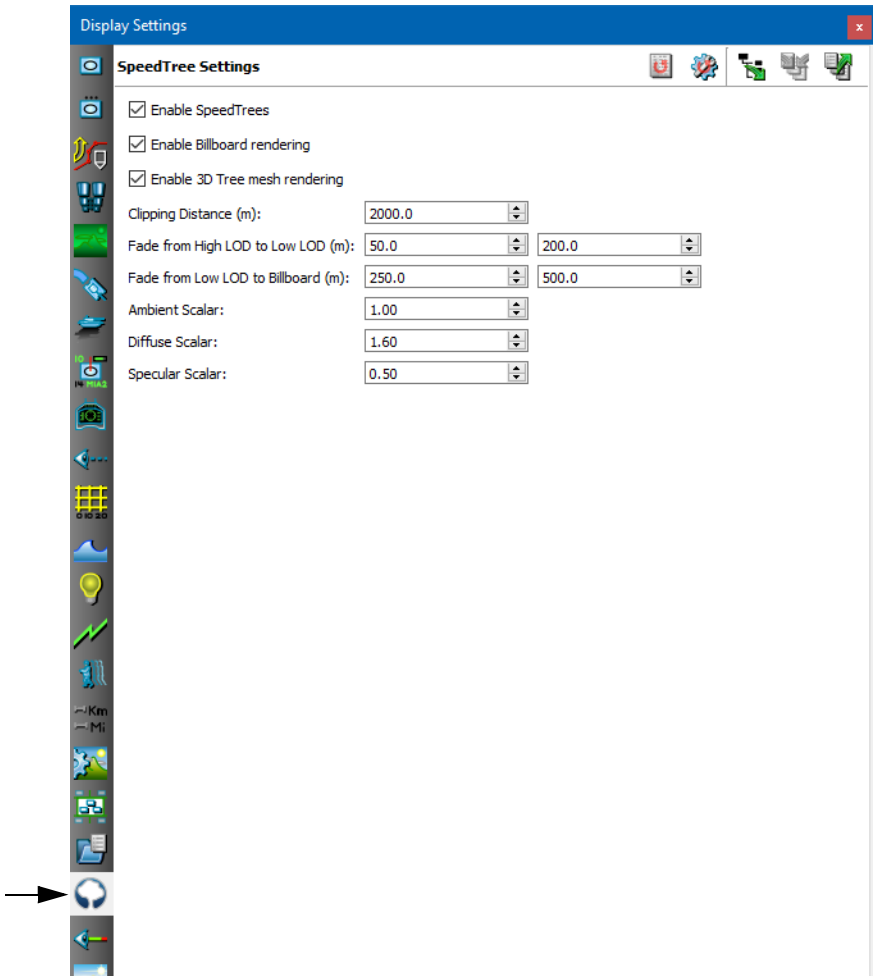


Figure 35-15. SpeedTree Settings page

3. Update parameters as desired.
4. Click Close.

35.8.1. Randomizing the Size of SpeedTrees

SpeedTrees can be created as props or streamed into the scene using an earth file. They have a default size. You can configure VR-Vantage to scale the size of SpeedTrees by a fixed amount or you can scale them randomly within a specified range. The scale applied to a SpeedTree is the product of several factors.

If a SpeedTree is a prop, it is scaled as the product of the following settings:

- The **scale** attribute in its model definition. If the **scale** attribute is not part of the model definition, the value defaults to 1.0.
- A random scale factor as specified on the SpeedTree Randomization page on the Terrain Settings dialog box, if enabled.
- A feet-to-meters conversion factor of 0.3048.

If a SpeedTree is a streamed in using an earth file, it is scaled as the product of the following settings:

- The **scale** attribute in its model definition. If the scale attribute is not part of the model definition, the value defaults to 1.0.
- The **marker-scale** attribute in the `model` element that configures the trees.
- A random scale factor as specified on the SpeedTree Randomization page on the Terrain Settings dialog box, if enabled.
- A feet-to-meters conversion factor of 0.3048.

You can also randomize the orientation of SpeedTrees.

To specify randomization of SpeedTree scaling:

1. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
2. Select the SpeedTree Randomization page (Figure 35-16).

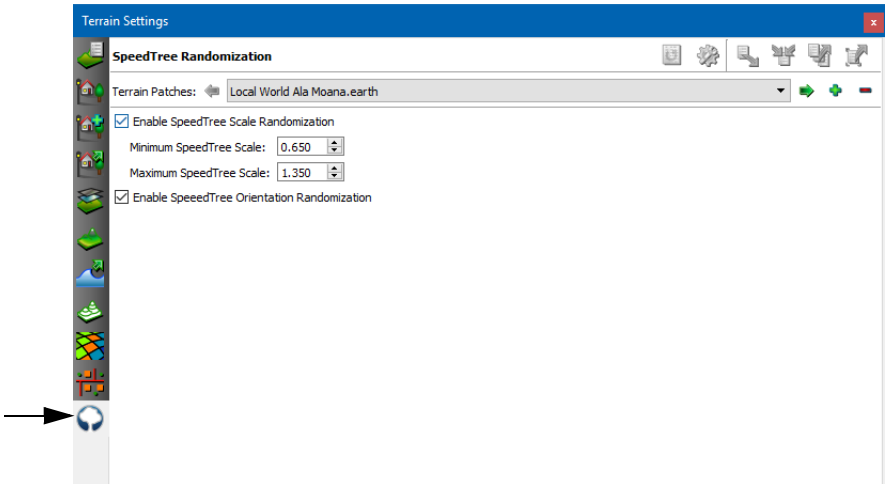


Figure 35-16. SpeedTree Randomization page

3. To enable randomization of the SpeedTree scale, select the Enable SpeedTree Scale Randomization check box.
4. If you enabled Enable SpeedTree Scale Randomization, specify values for the Minimum Tree Scale and Maximum Tree Scale.
5. To enable randomization of tree orientation, select the Enable SpeedTree Orientation check box.

35.8.2. Displaying SpeedTrees in Planar Reflections

By default, SpeedTrees are not displayed in planar reflections. You can enable their display. For details, please see "Configuring Ocean Planar Reflections," on page 13-8.

35.9. Best Practices for Creating Models for VR-Vantage

If you build your own models for use in VR-Vantage, following these guidelines will result in models that work efficiently:

- Keep all textures to powers of two, for example 256x256, 128x512, or 2048x2048:
  - Make sure textures use as many of the pixels as possible and that there are no large black areas in the texture.
  - Keep alpha (transparency) to a minimum.
- Minimize the number of textures:
  - Create a texture atlas that uses multiple textures combined into one.
  - Group polygons together that use the same texture.
  - Use the same textures for all LOD nodes.
- Minimize the number of nodes:
  - Keep the hierarchy as flat as possible.
  - Move all LOD nodes to the top of the hierarchy.
  - Minimize unnecessary transforms.
  - When you reuse geometry in the model, use instancing.
- Minimize the number of polygons that have different attributes:
  - Use the same material on polygons whenever possible.
  - Use two polygons with flipped normals instead of using the double sided attribute.
  - Use the same color on polygons whenever possible.
  - Group polygons with similar attributes.
- Consider using a more aggressive LOD structure:
  - Reduce the switchout distance to be smaller so that the LODs switch sooner.
  - Minimize the number of LODs.
  - Make sure that models switch out completely at a reasonable distance.
- Reduce the number of external references:
  - Merge external reference files.
  - Create instances from external merged reference files.

35.10. Compressing Model Files

The models and image files provided with VR-Vantage are shipped in compressed formats (MAK Encrypted Data Format (MEDF) and MAK Encrypted Image Format (MEIF)). You may want to compress your models to save space or protect intellectual property. You can convert supported formats to MEDF and MEIF with *./bin64/medfTool.exe*. The compression tool supports the 3D modeling formats supported by VR-Vantage, such as OpenFlight, 3DS, and OBJ, and raster formats such as RGB and PNG.



When VR-Vantage processes a request to load a a file, such as an external reference in a terrain, it checks the cache and the directory of the requested file for an MEDF version of the file. If it does not exist, it loads the file in the original format.

The syntax for medfTool is as follows:

```
medfTool [-q simple_filename ... -p <integer> -s directory
   ... -z optimization -b -i -o -m extensions ...
   -r #comment -x extensions ... --norecurse
   --directory directory -f filename -- -v -h]
```

Table 35-6 describes the arguments.

Table 35-6: medfTool arguments

Argument	Description
`(-- | --ignore_rest)`	Ignore all arguments after this one.
`(-b | --ignoreNonBase-Tex)`	Ignores any non-base texture (diffuse texture) for a model. This is used when you do not want any multiple diffuse textures or any effect textures associated with it.
`--directory` *directory*	Specifies the directory to search for image and geometry files to convert into MEDF and MEIF. All subdirectories are visited unless the `--norecurse` command line option is used.
`(-f | --file)` *filename*	Convert just this file. The extension of the file must be specified using `-x` or `-m` to indicate it is an image or a geometry file.
`(-h | --help)`	Displays usage information.
`(-i | --ignore_errors)`	Ignore errors in the encryption process and continue without exiting.
`(-m | --image_exten-sions)` *extension*	Specifies the extensions of image formats which will be converted into MEIF format. Use this argument to encrypt a directory. Can be used multiple times.
`--norecurse`	Do not recurse through subdirectories. Default: recurse through subdirectories.

Table 35-6: medfTool arguments

Argument	Description
(-o | --onlyIfNew)	Only generates an encrypted file if the expected output file does not exist.
(-p | --threads) *threads*	Specifies the number of threads to run. Specifying zero results in the number of threads equaling the number of logical processors the host machine has.
(-q | --forceOpaque) *simple_filename*	Specifies a filename and extension, but no path, to force polygons with these filenames to be opaque. Accepted multiple times.
-r *#comment*	Removes any any node tagged with the comment from the scene graph in a model. This is used in special cases to remove nodes that are not needed. Example, `-r #missionfunctiononly`.
(-s | --search) *directory*	Adds a directory to the search path when resolving external references. During processing, any external references not found are removed from the generated files. Files in the search path are not converted.
(-v | --version)	Display version information and exit.
(-x | --extensions) *extension*	Specifies the extensions of geometry file formats that will be converted into MEDF format. Use this argument to encrypt a directory. Can be used multiple times.
(-z | --optimize) *optimization*	Specifies the optimization level. Maximum optimization may provide better optimization than the default setting, but it is significantly slower. Possible values are: ◆ 0 = None. ◆ 1 = Default. ◆ 2 = Maximum optimization.

The following example recursively compresses all FLT, RGB, RGBA, PNG, and DDS files in *%MAK_DATADIR%/Terrain*:

```
%MAK_IGDIR%/bin64/medfTool.exe
  --directory %MAK_DATADIR%/Terrain --extensions flt
  --image_extensions rgb --image_extensions rgba
  --image_extensions png --image_extensions dds
```

The following example compresses all FLT, RGB, RGBA, PNG, and DDS files in *%MAK_DATADIR%/Lifeforms*. It does not compress files in subdirectories.

```
%MAK_IGDIR%/bin64/medfTool.exe
  --directory %MAK_DATADIR%/Lifeforms --norecurse
  --extensions flt --image_extensions rgb
  --image_extensions rgba --image_extensions png
  --image_extensions dds
```



36. Configuring Emitter Volumes

This chapter explains how to configure display of emitter volumes.

Configuring Emitter Volumes 36-2
Configuring Emitter Volume Color 36-2
Controlling Emitter Volume Radius 36-3
Configuring Emitter Volume Segments 36-4

36.1. Configuring Emitter Volumes

You can configure how VR-Vantage colors emitter volumes, how they are drawn, and how they are scaled for visibility. You can configure emitter volumes by editing the DefaultEllipsoidArc model definition or by creating customized ellipsoid arc model definitions and applying them to different entity models.



In order for an entity to display emitter volumes in 2D observer modes and 3D observer modes, it must have a 2D Emitter Volume visualizer in its 2D model set and 3D Emitter Volume visualizer in its 3D model sets. The DefaultEllipsoidArc model definition is an attribute of the visualizer.

36.1.1. Configuring Emitter Volume Color

By default, emitter volumes are displayed using fixed colors for high frequency and low frequency. You can configure emitter volumes to be colored based on the emission frequency or pulse rate. When emitter volume color represents frequency, color is determined by the frequency field of the EM Emission update message. In both cases, you specify a low frequency or pulse rate value and a high value.

If the emitter frequency or pulse rate is lower than your specified low value, the emitter volume is red. As frequency rises above the specified low value, emitter volume color changes, progressing through the colors of the visible spectrum until the specified high value is reached. At the high value and above, the emitter volume is violet.

To change coloring from the default, you must add the appropriate attributes to the emitter volume visualizer for the entities whose model definitions you want to change. Table 36-1 describes the emitter volume visualizer attributes that affect emitter volume color.

Table 36-1: Emitter volume visualizer attributes

Attribute	Description
colorByFrequency	If True, color by frequency instead of using static colors.
lowFrequency	The low frequency threshold for coloring by frequency. Default: 800 MHz.
highFrequency	The high frequency threshold for coloring by frequency. Default: 1 GHz.
usePulseFrequency	If colorByFrequency is True and usePulseFrequency is True, color emitter volumes by pulse frequency.

Table 36-1: Emitter volume visualizer attributes

Attribute	Description
lowPulseFrequency	The low pulse rate threshold for coloring by frequency. Default: 100 Hz.
highPulseFrequency	The high pulse rate threshold for coloring by frequency. Default: 3 KHz.

36.1.2. Controlling Emitter Volume Radius

A emitter volume's radius (that is, the distance it extends from the emitter) is based in part on emitter power and in part on elevation and azimuth sweep data obtained from EM-Emission state messages. A higher emitter power value results in a longer radius, while a wider field of view results in a shorter radius (because the radiated power is distributed over a wider area).

If VR-Vantage relied exclusively on these values, the range of sizes among different emitter types could become too wide to be practical, with some emissions being barely visible, and other emissions covering the entire virtual world. Therefore, a configurable exponent weight and scale factor have been added to help fine tune the emitter beam radii. The radius exponent weight (a value between 0 and 1) slows the growth of the radius when power or area is greater than 1 and increases the growth of the radius when power or area is less than 0. The scale factor increases the radii of all beams by the given value.

To change the scale factor and exponent from the default, you must add one or both of the following attributes to the emitter volume visualizer for the entities whose model definitions you want to change:

- linearScale. The amount of linear scaling applied to all emitter volumes. Default: 100.
- exponentialScale. The amount of exponential scaling applied to emitter volumes. Default: 0.1.

How VR-Vantage Calculates the Radius

VR-Vantage calculates emitter volume radius by approximating the spread of the emissions based on the elevation and azimuth sweeps. If the approximate coverage is zero, it sets the value to 1. Then it applies the linear and exponential scaling values. The code for implementing this is as follows:

```
double tempVar = 8*elevationSweep*sin(2*azimuthSweep);
if (tempVar <=0) tempVar=1.0;
scale = linearScale * pow(sqrt(power/tempVar), exponentialScale );
```

36.1.3. Configuring Emitter Volume Segments

VR-Vantage displays an emitter volume as a group of spherical sections projecting from the sensing device. By default, for each 10 degrees (or portion thereof) in the emitter's field of view, VR-Vantage draws one segment. Figure 36-1 shows segments set to 10° (the default) and 30°.

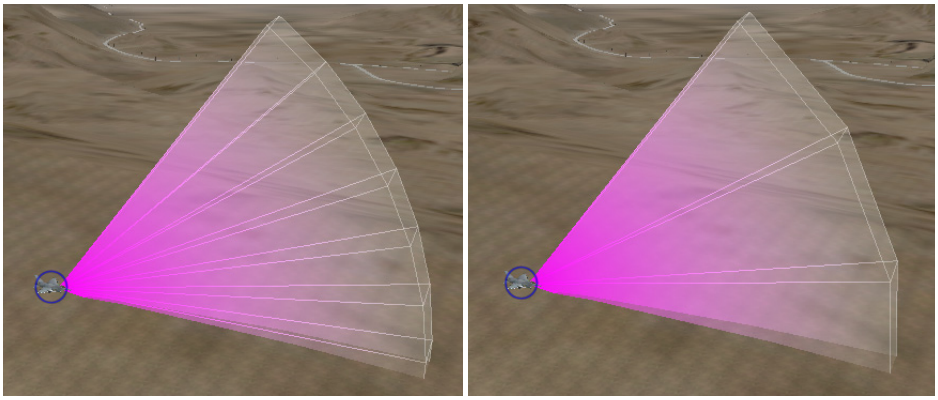


Figure 36-1. Emitter volume segments

Emitter volume segments are managed by the EllipsoidArc schema and model definitions based on it. VR-Vantage ships with one ellipsoid arc model definition – DefaultEllipsoidArc. You can edit this model definition or create copies of it and customize them. If you want to have different emitter volumes for different entities, assign a different ellipsoid arc model definition to each one.

Specifying the Segment Angle

To specify the segment angle:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Model Definition Editor page.
3. Click the Filter button (▾) and select EllipsoidArc. The ellipsoid arc model definitions are listed in the window.
4. Select DefaultEllipsoidArc (Figure 36-2).

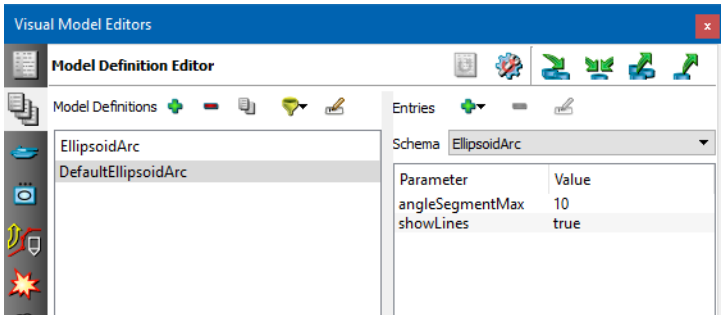


Figure 36-2. EllipsoidArc model definition

5. Change the value of the **angleSegmentMax** parameter. The minimum value is 10 (degrees); the maximum is 45.
6. Export the updated setting.

Displaying Segment Outlines

By default, VR-Vantage displays opaque outlines showing the segment edges along the surface of the emitter volume.

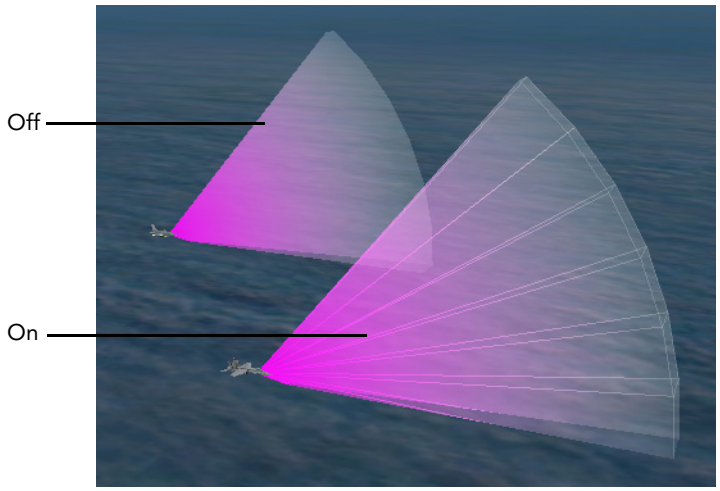


Figure 36-3. Segment outlines

To display or hide segment outlines:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Model Definition Editor page.
3. Click the Filter button (▾) and select EllipsoidArc. The ellipsoid arc model definitions are listed in the window.
4. Select DefaultEllipsoidArc.
5. Change the value of the **showLines** parameter.
6. Export the updated setting.

Adding a Custom EllipsoidArc Model Definition

If you want to customize the segment angle and segment outlines for a particular entity without changing the values for other entities, you must create a separate ellipsoid arc model definition and add it to the entity's model definition.

To create a custom ellipsoid arc model definition:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Model Definition Editor page.
3. Click the Filter button () and select EllipsoidArc. The ellipsoid arc model definitions are listed in the window.
4. Select DefaultEllipsoidArc.
5. Click the Duplicate Model Definition button (). The Duplicate Model Definition dialog box opens with a default definition name.
6. Accept the generated name or type a new name for the model definition.
7. Click OK. The model definition is added to the list.
8. Change the parameter values for the new model definition as desired.
9. Select the Entity Definition Editor page.
10. Select the entity whose emitter volume you want to change.
11. Select the 3D Sensor Volume visualizer for the entity.
12. Click the Value column for the modeldefinition attribute. The Choose Model Definition dialog box opens. It has a list of model definitions.
13. Select the model definition that you just created.
14. Click OK.
15. Export the updated setting.
16. If you are connected to an exercise, disconnect and then reconnect to enable the new model definition.



37. Adding DI-Guy Models to VR-Vantage

This chapter explains how to take DI-Guy models created with the DI-Guy SDK and integrate them into VR-Vantage.

Adding New DI-Guy Models to VR-Vantage 37-2
Copy DI-Guy Files to the Data Directory 37-2
Add Model Definitions and Element Definitions 37-2

37.1. Adding New DI-Guy Models to VR-Vantage

VR-Vantage supports DI-Guy characters and includes many lifeform characters. However, MAK customers sometimes want to add their own DI-Guy characters. The general process for adding a new character is as follows:

1. Create the new character using DI-Guy software and copy the required files to the *./data* directory.
2. Add a model definition and element definition for the new character.

37.1.1. Copy DI-Guy Files to the Data Directory

VR-Vantage stores files for DI-Guy characters in *./data/Lifeforms/DIGuy* and its subdirectories. After you create a new character using DI-Guy software, you must copy the files for this character to these directories.

DI-Guy documentation recommends that all customer content be placed in *$(DIGUY)/custom*. To follow this convention, create a directory called *./data/Lifeforms/DIGuy/custom* and put your files in the appropriate subdirectories under it.

VR-Vantage can visualize custom DI-Guy characters based on DIS enumerations mapped to element definitions or by recognizing custom DI-Guy PDUs.

- If you want to map your custom characters to element definitions, follow the procedures in this chapter.
- If you want to visualize characters based on custom DI-Guy PDUs without creating element definitions or object type mappings, place the DI-Guy files in the *./data/Lifeforms/DIGuy/custom* directory as described in the previous paragraph. No further configuration is needed.

37.1.2. Add Model Definitions and Element Definitions

To visualize a DI-Guy character in VR-Vantage, you must add a model definition and element definition for each character.

Create a DI-Guy Model Definition

To create a model definition for a DI-Guy:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Model Definition Editor page.
3. Click the Add button (+) next to the Model Definitions label. The Add Model Definition dialog box opens.
4. Type a name for the new model definition. It must end with the letters DIG.
5. In the Schema list, select DiGuyCharacter (Figure 37-1).

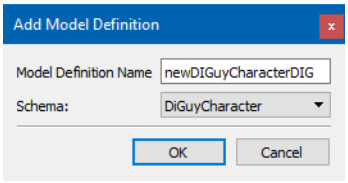


Figure 37-1. Add Model Definition dialog box

6. Click OK. The new model definition is added. It is highlighted in red because it is missing parameters (Figure 37-2).

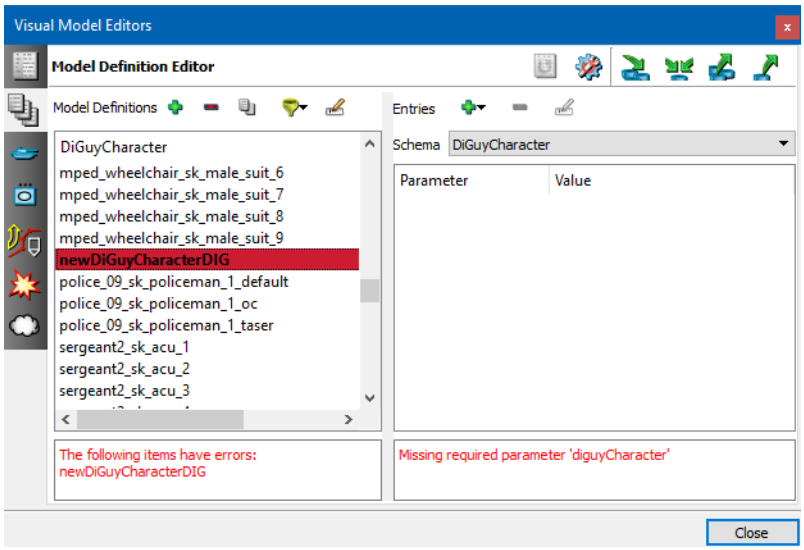


Figure 37-2. New model definition

7. Click the Add Parameter button (). A list of parameters is displayed (Figure 37-3).

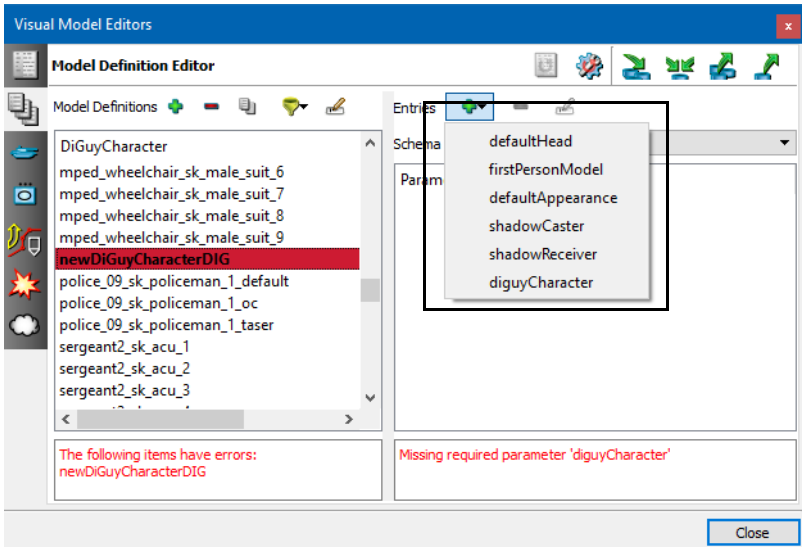


Figure 37-3. Add parameter list

8. Select defaultAppearance. It is added to the parameter list.
9. Click the value for the parameter. The field becomes editable.
10. Type the default appearance name for this character (Figure 37-4).
11. Add a diguyCharacter parameter.
12. Click the value for the parameter. The field becomes editable.
13. Type the character name for this character.
14. Add the shadowCaster and shadowReceiver parameters (Figure 37-4).
15. Export the new model definition.

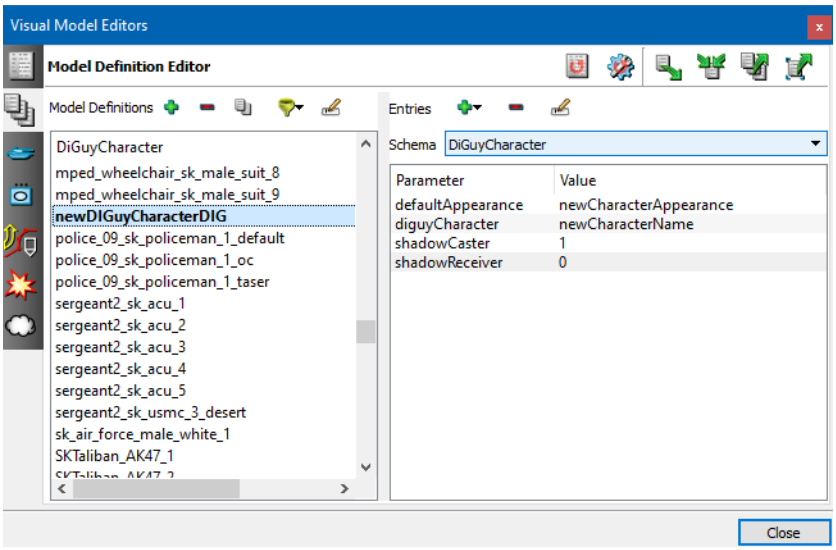


Figure 37-4. DI-Guy character parameters

Add an Element Definition

To create an element definition:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Entity Definition Editor page.
3. Expand the element definition hierarchy to the Life Form level and select the type of life form you are adding (Figure 37-5).

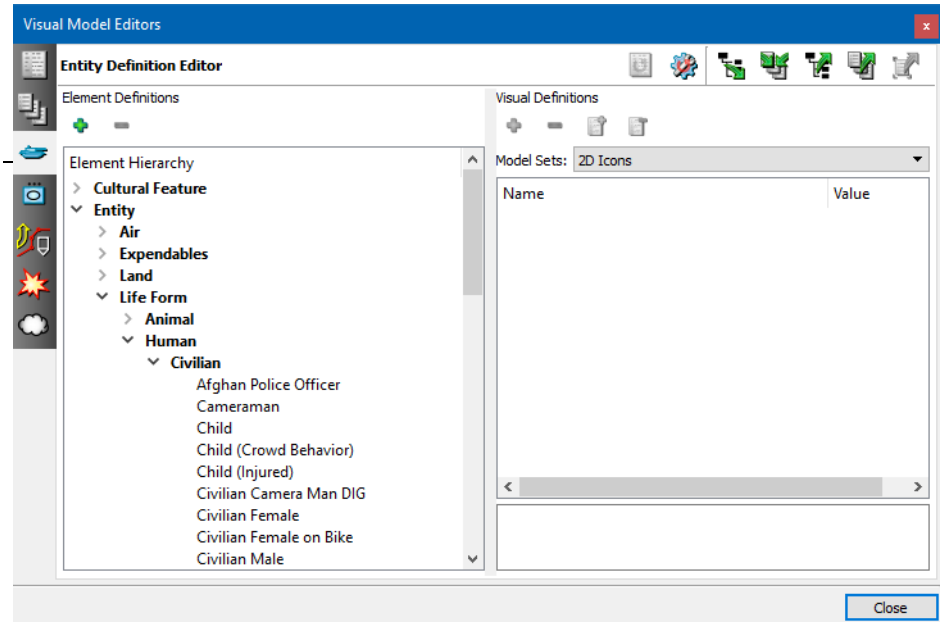


Figure 37-5. Entity Definition Editor page

4. Click the Add button (). The Create Element Definition dialog box opens.
5. Type a name for the element definition. (You cannot edit the name after you create it, so try to avoid typing errors.)
6. Click OK. The new element is added to the list (Figure 37-6).

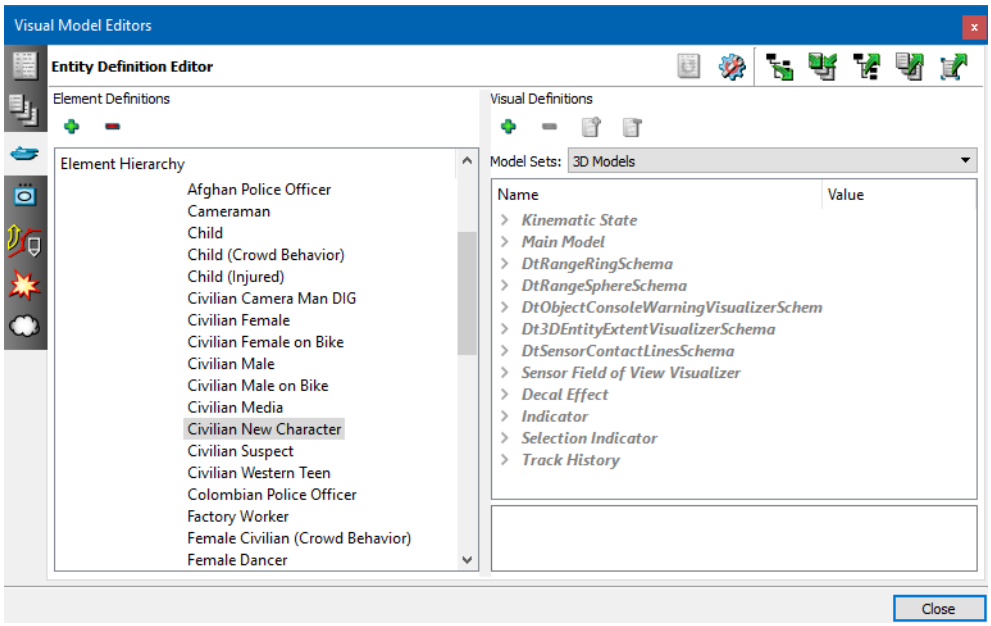


Figure 37-6. New entity definition added

7. In the Visual Definitions list, select Main Model (Figure 37-7).

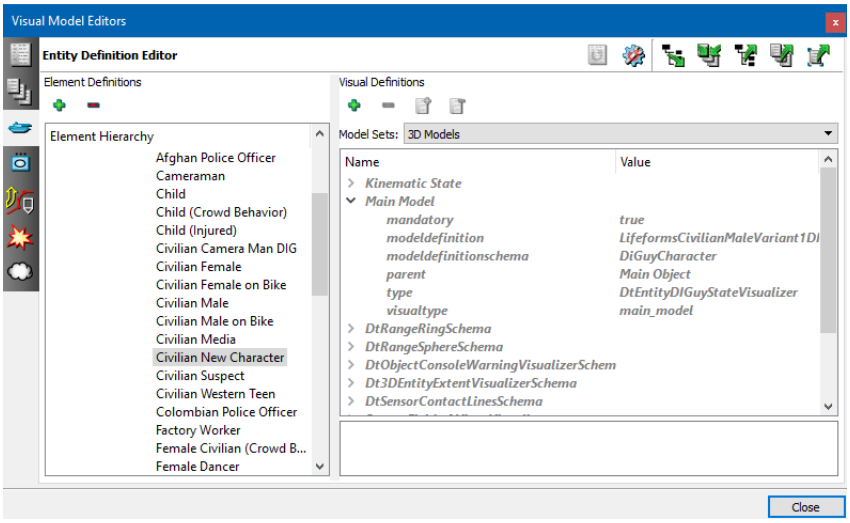


Figure 37-7. Main Model visual definition

8. Click the Value for the **modeldefinition** attribute. The Choose Model Definition dialog box opens.
9. Click the Add button (+). A list is added to the window (Figure 37-8).

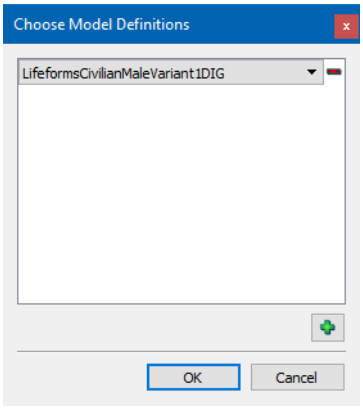


Figure 37-8. Choose Model Definition dialog box

10. Select the model definition that you created for this character (Figure 37-9).

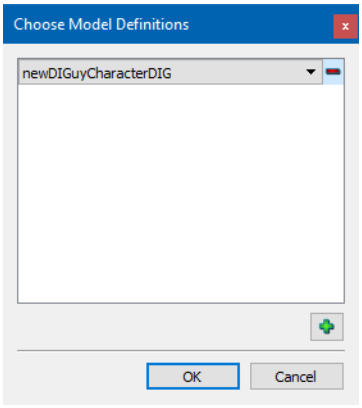


Figure 37-9. New model definition selected

11. Click OK. The model definition is updated (Figure 37-10).

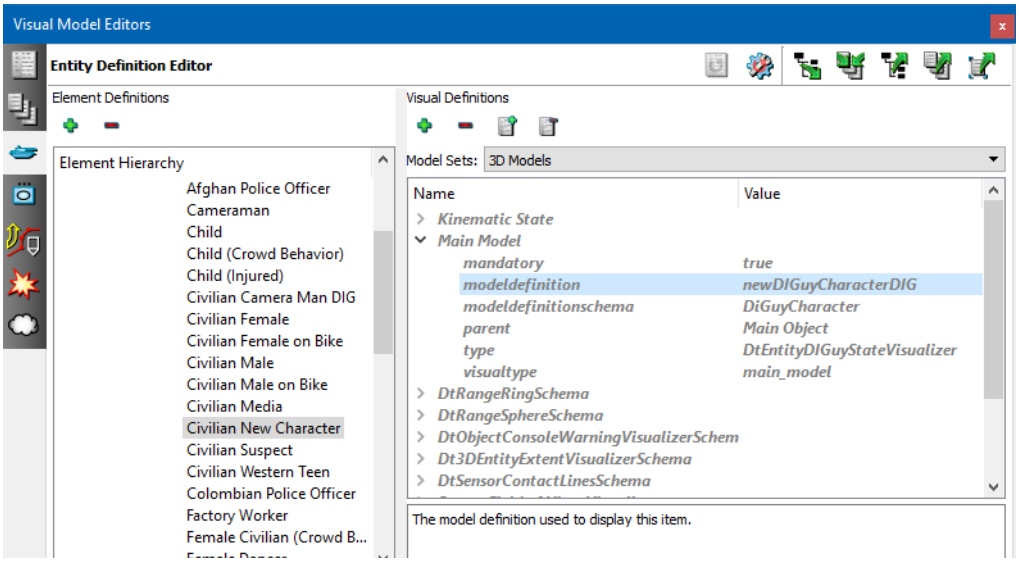


Figure 37-10. Completed entity definition

12. Select the value column for the modeldefinitionschema attribute. The Choose Supported Model definition Schema dialog box opens.
13. Click the Add button (). A list is added to the window.
14. Select DiGuyCharacter.
15. Click OK.
16. Close the dialog box.
17. Export the new element definition.



38. Model Tutorials

Although the individual procedures for managing model definitions and model mappings are fairly straightforward, understanding how they fit together to help you achieve your visualization goals is not always obvious. The tutorial in this chapter demonstrates how to combine various procedures to add models.

Adding a Model to VR-Vantage 38-2
Create a Model Definition 38-2
Add an Element Definition 38-5
Add a Model Mapping 38-8
Test the Model Mapping 38-10

38.1. Adding a Model to VR-Vantage

This tutorial shows how to add a new model to a VR-Vantage application. To help you with the tutorial, VR-Vantage includes a model of an F/A-18 Hornet that is not mapped to any entities.

This tutorial has you save the new model definition and element defintion to new files.

To see a set of brief videos that demonstrate this tutorial, choose **All Programs** → **MAK Technologies** → **VR-Vantage 2.3** → **Documentation** → **VR-Vantage Video Tutorials**, run *./doc/vrvantage_tutorialvideos.htm*, or click the camera icon next to the procedure whose video you want to view.

The basic steps are:

1. Create a model definition ("Create a Model Definition," on page 38-2).
2. Add an element definition ("Add an Element Definition," on page 38-5).
3. Map the new model to a simulation object or object type enumeration ("Add a Model Mapping," on page 38-8).
4. Test the new model ("Test the Model Mapping," on page 38-10).

38.1.1. Create a Model Definition

A model definition is an instance of a schema. It specifies values for the parameters required by the schema.

To create the model definition:



1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Model Definition Editor page (Figure 38-1).

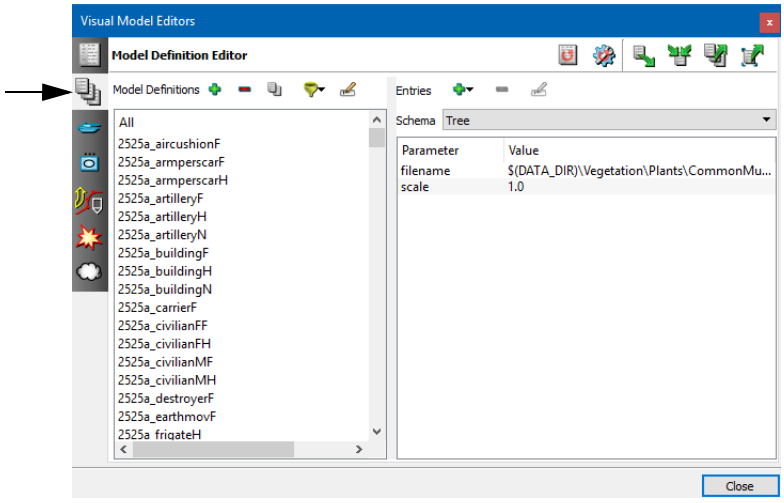


Figure 38-1. Model Definition Editor page

3. Optionally, filter the model definitions in the list to just show entity model definitions. (For details, please see "Filtering the List of Model Definitions," on page 32-8.)
4. Click the Add button (+) next to the Model Definitions label. The Add Model Definition dialog box opens (Figure 38-2).

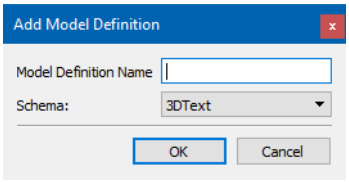


Figure 38-2. Add Model Definition dialog box

5. Type a name for the new model definition, for example, myF18.
6. If you filtered the model definition list, the Schema box says Entity. If not, in the Schema list, select Entity. The dialog box now looks like Figure 38-3.

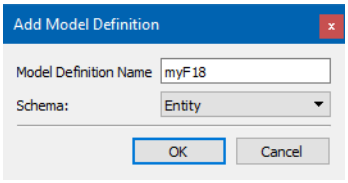


Figure 38-3. New model definition

7. Click OK. The new definition is added to the list (Figure 38-4). It does not have any parameters specified yet, so VR-Vantage prints an error message. Now you must specify the model file for this model definition.

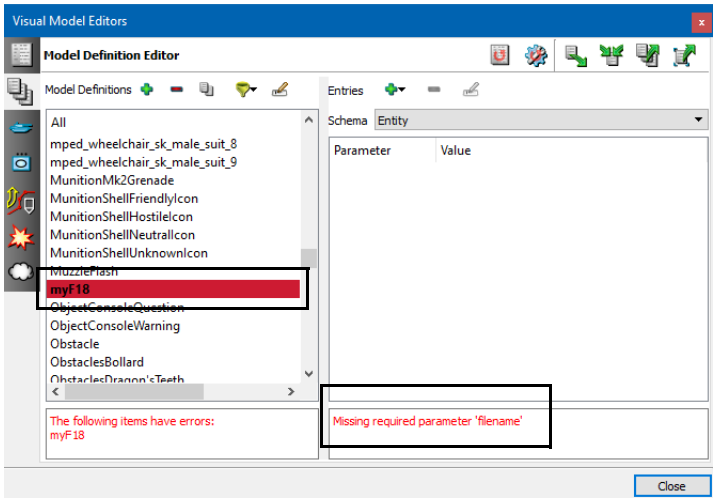


Figure 38-4. New model definition in list

8. Select the new model definition.

9. Click the Add Parameter button (). A list of parameters is displayed (Figure 38-5).

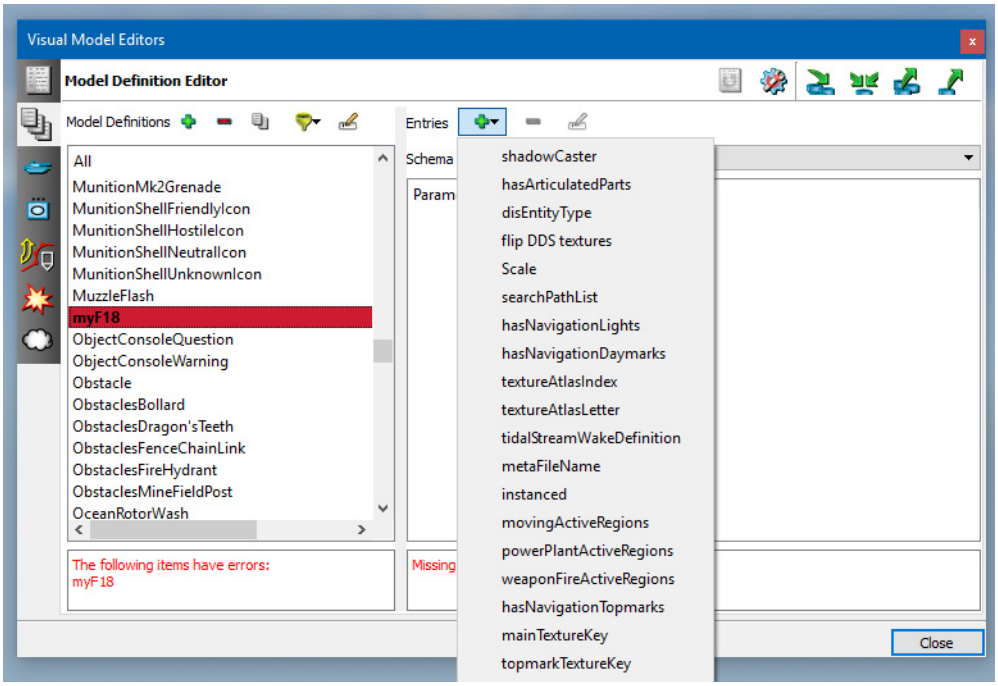


Figure 38-5. List of parameters

10. Select **filename**. It is added to the parameter list.
11. Click the value for the parameter. A file selection dialog box opens.
12. Navigate to *./data/Vehicles/FixedWing/FA-18_Hornet* and select *FA-18_Hornet.flt*.
13. Click OK. The value is updated (Figure 38-6).

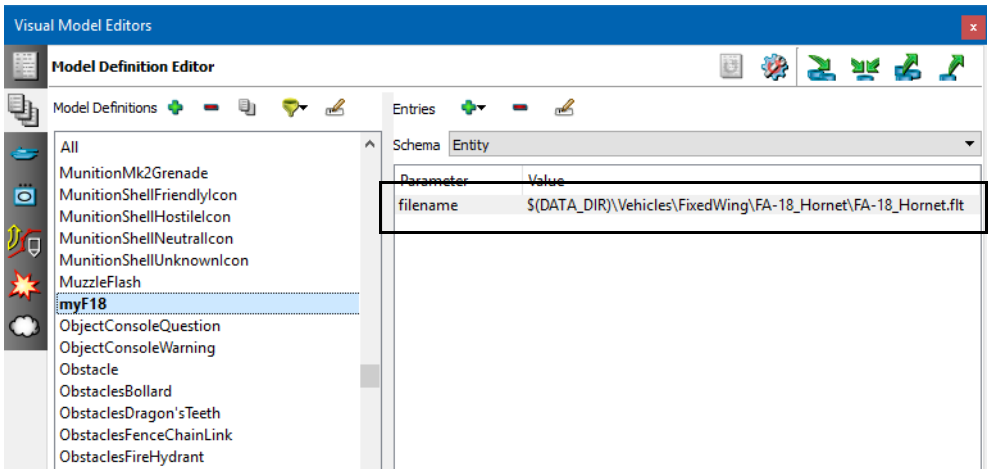


Figure 38-6. Model definition parameter

14. In the Model Definition list, make sure myF18 is selected.
15. Click the Export Selected button ( ). The Export Model Definitions dialog box opens.
16. In the File Name box, type a name for the model definition, such as myF18-ModelDefinition.
17. Click Save.

38.1.2. Add an Element Definition

An element definition configures the visualizers used to display an element and its effects. You must specify the model definition to use for the Model Visualizer.

To add an element definition:

1. Choose **Settings** → **Visual Model Editors**. The Visual Model Editors dialog box opens.
2. Select the Entity Definition Editor page (Figure 38-7).

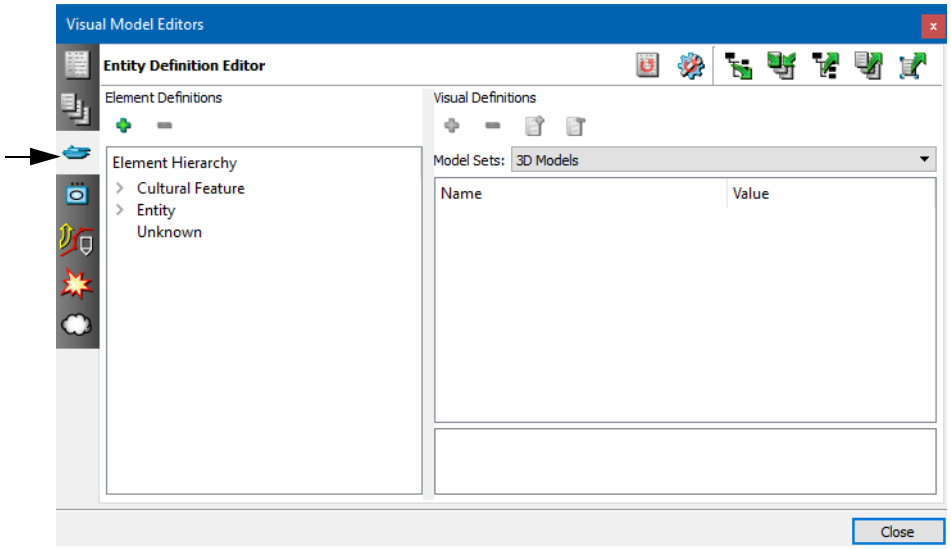


Figure 38-7. Entity Definition Editor page

3. Expand the Entity section of the hierarchy and then expand the Air section.
4. Expand Fixed Wing.
5. Expand Medium Fixed Wing.
6. Select Medium Fixed Wing.
7. Click the Add button ( ) next to the Element Definitions label. The Create Element Definition dialog box opens.
8. Type a name for the new entity, for example, myF18.

9. Click OK. An entry called myF18 is added to the list of fixed wing entities (Figure 38-8). It has the same list of visualizers as the fixed wing parent level.

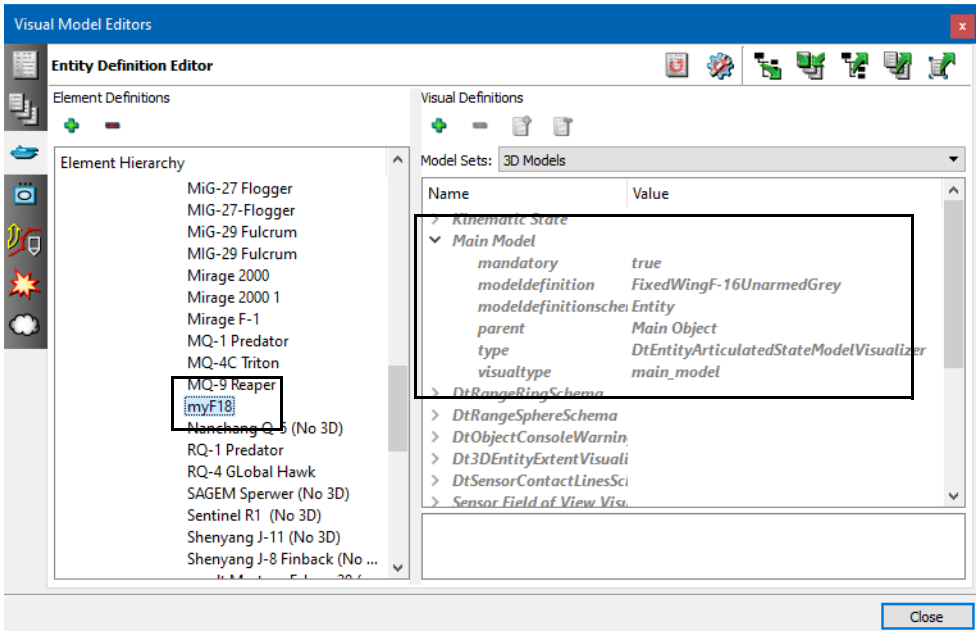


Figure 38-8. New entity definition

10. Scroll through the Visual Definitions list until you find the Main Model visualizer. This visualizer specifies the model definition that this element should use. It has the model definition of the parent element. We need to change it to use the myF18 model definition.
11. Click the Value column for the **modelDefinition** attribute to make it editable. The Choose Model Definitions dialog box opens (Figure 38-9). It shows the current model definition as a selection in a list.

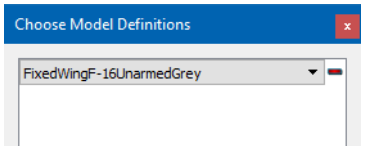


Figure 38-9. Choose Model Definitions dialog box with list of definitions

12. Click the Down Arrow to expand the model definition list and select the new model definition (myF18).
13. Click OK. The model definition is now specified (Figure 38-10).

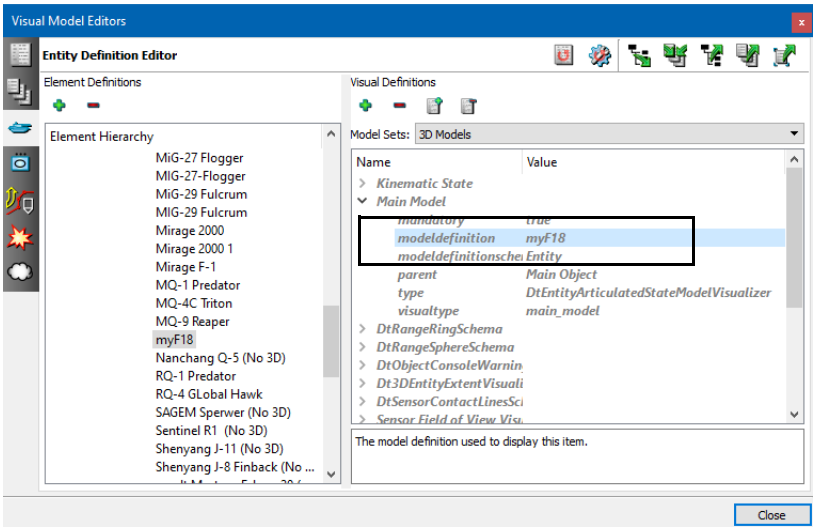


Figure 38-10. Model Definition for Model Visualizer

14. In the Element Definitions list, make sure myF18 is selected.
15. Click the Export Selected Leaves button (). The Export Element Definitions dialog box opens.
16. In the File Name box, type a name, such as myF18ElementDefinition.
17. Click Save.
18. Close the Visual Model Editors dialog box.

38.1.3. Add a Model Mapping

Now, you have to map the new model definition to an object type enumeration.

To add a model mapping:

1. Choose **Settings** → **Simulation Object Type Mappings**. The Simulation Object Type Mappings dialog box opens.
2. Select the Entity Mapping Settings page (Figure 38-11).

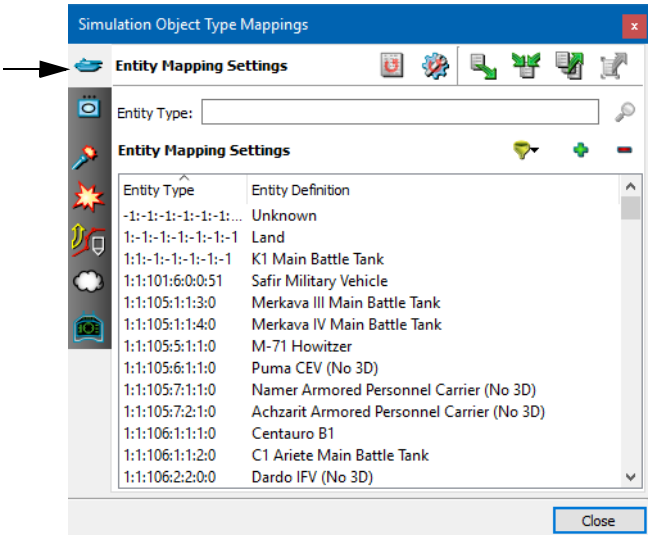


Figure 38-11. Entity Mapping Settings page

3. Click the Filter button ( ).
4. In the list, select **Entity** → **Air** → **Fixed Wing** → **Medium Fixed Wing**. (This does not change the Entity Mapping Settings list; it filters the list in the Create New Entity Type Model Mapping dialog box.)
5. Click the Add button ( ). The Create New Entity Type Model Mapping dialog box opens (Figure 38-12). It lists a default entity type enumeration.

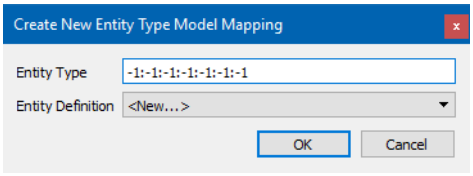


Figure 38-12. Create New Entity Type Model Mapping dialog box

6. In the Entity Type text box, type the enumeration you want to map to the new model mapping. (The enumeration for an F/A18 is 1:2:225:1:9:1:-1.)

VR-Vantage already has an F/A18 mapped to this enumeration. However you can map multiple model definitions to an enumeration. When an object type that has multiple model definitions mapped to it is discovered in a simulation, VR-Vantage randomly assigns a model definition from among those mapped to the object type.

7. In the Entity Definition list, select the model definition that you added in the previous procedure (myF18) (Figure 38-13).

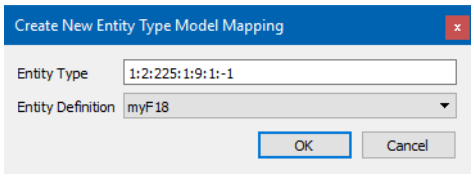


Figure 38-13. Create new model mapping

8. Click OK.
9. The new model mapping is added to the list.

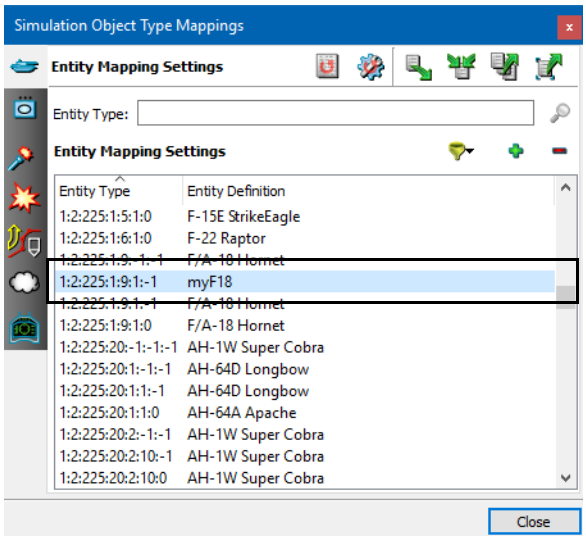


Figure 38-14. New model mapping

10. Make sure that the mapping is selected in the list.
11. Click the Export Selected button ( ). The Export Mapping Settings dialog box opens.
12. In the File Name box, type myF18Mapping.
13. Click Save.

38.1.4. Test the Model Mapping

It is a good idea to test the new model mapping to see if the correct model is being displayed. You can test a model by running a simulation that has the entity or object type, or you can perform the simple test described here.

To test a model:



1. Open a terrain database that has easily identifiable props. For the purposes of this tutorial, open the Makland terrain.
2. In the VR-Vantage window, move the observer so that you are viewing the palace on top of the hill near the town in Makland.
3. Click the palace to select it (Figure 38-15).

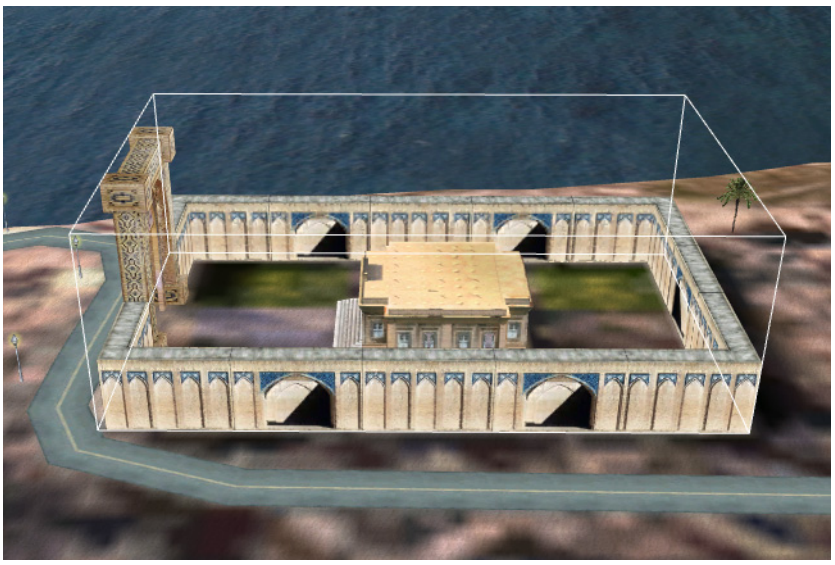


Figure 38-15. Palace

4. Choose **Settings** → **Terrain**. The Terrain Settings dialog box opens.
5. Select the Edit Existing Props page. The list of props shows that Building26, the palace, is selected (Figure 38-16).

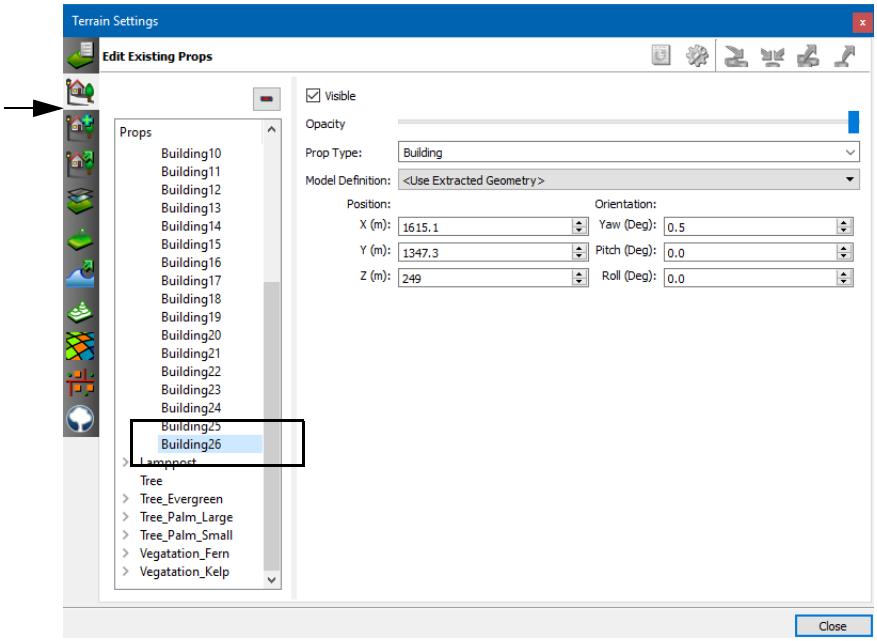


Figure 38-16. Edit Existing Props page

6. In the Model Definition list, select the model definition that you added earlier in this tutorial (Figure 38-17). The model of the palace is replaced with the model for the F/A-18 (Figure 38-18). If this is the model that you wanted to add, then you know that you completed the procedure correctly. If this is not the model, you expected, check the filename you specified for the model definition.

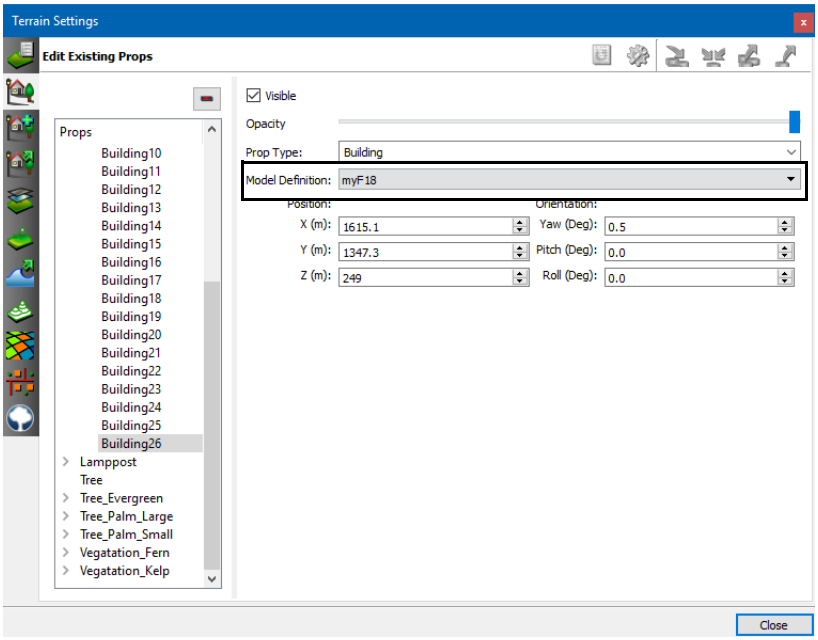


Figure 38-17. Changed model definition

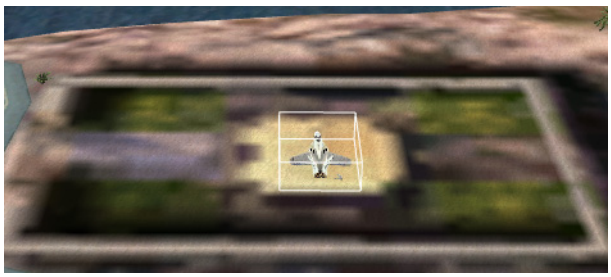


Figure 38-18. F/A-18 replaces the palace

7. If you are going to use this terrain for simulations, change the model definition back to Use Extended Geometry.



39. The WRM Specification (DIS Notes)

Some OpenFlight files contain information specified by the *WRM Entity Model Specification for DIS Interoperability*. This information makes it possible for VR-Vantage to depict the state of an entity's articulated and attached parts correctly and to switch among different entity appearances.

This chapter contains information that can help you build entity models that comply with the WRM specification, or add that compliance to existing models.

The WRM specification was developed by MAK Technologies (now VT MAK), Paradigm Simulation Inc., MultiGen Inc., and SAIC, for the ARPA War Breaker project. (Paradigm Simulation and MultiGen merged and were later acquired by Presagis.)



Although the WRM specification was developed for DIS, it is used in the HLA RPR FOM.

Introduction 39-2
The OpenFlight File Format 39-2
- Node Name and Comment Fields 39-2
- External References and Instancing 39-3

Modeling for DIS Interoperability 39-3
- The DIS Attribute Lexicon (DAL) 39-3
- DAL Keywords 39-4
- Model Coordinate Systems 39-5

Articulated Parts 39-7
Entity Appearances 39-8
Animation 39-11
DIS Keyword Values 39-12

39.1. Introduction

In order for visualization applications such as VR-Vantage to change the state of a model in response to network messages, they must have certain information about the model. For example, which piece of the model's geometry is associated with which articulated part, and which piece of a model is associated with a particular entity appearance attribute.

This chapter describes a method of encoding information in an OpenFlight format model so that applications can make these associations. For the most part, information is associated with a MultiGen node by including it as a comment associated with the node. Comments that contain this information begin with the token, `@dis`.

VR-Vantage depicts the movement of articulated parts and changes in entity appearances only when using models constructed to this specification.

This chapter represents a subset of the Model Specification for DIS Interoperability (version 001). For more details, please refer to that document.

Although VR-Vantage implements only a subset of the functionality enabled by the WRM Model Specification, we recommend that models conform to as much of the specification as is appropriate for the model, to promote compatibility with future versions of VR-Vantage, and with visualization applications from other vendors.

39.2. The OpenFlight File Format

The OpenFlight file format organizes geometry into the following types of nodes:

- Face nodes represent individual polygons.
- Object nodes represent face nodes that are logically or geographically related. Object nodes can have only face nodes (polygons) as children.
- Level of detail (LOD) nodes provide a means to select alternate geometries based on a user-definable range. Using lower resolution geometry for objects that are far from the observer saves memory and improves performance.
- Degree of Freedom (DOF) nodes specify translational or rotational freedom, or both, for portions of a model. They can be used for DIS articulated parts or animated special effects. They specify a type of transformation and its limits, such as "rotate about the x axis between 45 and 135 degrees". The OpenFlight approach to degrees of freedom is hierarchical.
- Group nodes represent logically-related or geographically-related object nodes. Group nodes are the most general, and may have any type of nodes, except face nodes, as children.

39.2.1. Node Name and Comment Fields

Each node contains a comment field for user-definable data. The WRM Specification defines keywords that, when placed in a comment field, provide the information that DIS applications need to properly render a model.

39.2.2. External References and Instancing

The following OpenFlight constructs provide a means to avoid duplicating massive amounts of geometry:

- An external reference (or xref) is a pointer to another OpenFlight file that contains additional geometry.
- A transformation matrix is usually attached to an xref to locate the position and orientation of the referenced object's coordinate system geometry. Each instance may have a transformation matrix controlling the position and orientation of the child's coordinate system.

39.3. Modeling for DIS Interoperability

The WRM Specification provides uniform standards for locating and orienting the model coordinate system and a way to map DIS codes to associated models and parts. By conforming to these standards, applications that support the WRM Specification do not need customized software to interpret the multiple potential model formats that might otherwise exist.

39.3.1. The DIS Attribute Lexicon (DAL)

DIS attributes are specified using the following syntax:

```
@dis keyword values # comment
```

where:

- `@dis` must precede the keyword.
- *`keyword`* is one of the DAL keywords.
- values can be a number or string. Depending on the requirements of the keyword, the values specified can be an individual value, a comma separated list, or a range.
- `#` indicates the start of a comment. Comments are case sensitive. A comment continues until the next `@dis` entry.

Remember that all comments will be parsed by VR-Vantage, so spelling counts.

39.3.2. DAL Keywords

The WRM Spec supports the following keywords:

- `articulated_part` *`part`*. Identifies the movable parts of a model with a unique DIS Articulated Part code. This keyword is placed in a Group node that is a child to the DOF node that identifies an articulated part, and above the geometry that is going to be articulated. This keyword is always followed by the motion keyword. The articulated part codes are specified in DIS Standard SISO-REF-010-2006 Enumerations and Bit Encoded Values Articulated Parts. They are listed in Table 39-12.

  This keyword is always followed by the switch keyword.

- `switch` *`appearance`*. Identifies a model appearance attribute with a DIS keyword. This keyword is placed in a Switch node in the model above the geometry that is going to be switched.

  The switch keyword takes the name value in DIS Standard SISO-REF-010-2006 Enumerations and Bit Encoded Values Entity Appearance (General and Specific), joined by the underscore character (_) if the name is multiple words, as its value. Note that frozen_status, power_plant_status, and state are preceded by an abbreviation for their domain, as follows: gm_ for guided munitions and lf_ for life forms. For a list of supported values, please see Table 39-6.

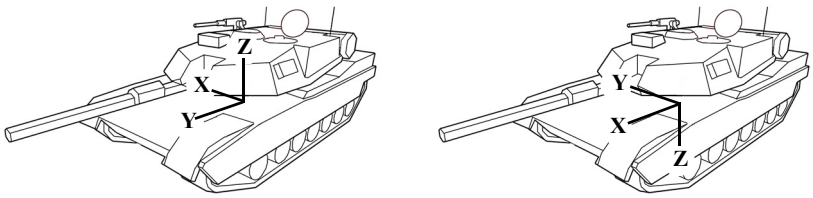
- `state` *`state`*,*`state`*... . Specifies the state of the appearance attribute with a DIS appearance state code. This keyword is placed in a Group node just below the switch node. For a list of supported values, please see Table 39-6.

39.3.3. Model Coordinate Systems

OpenSceneGraph (OSG) and the DIS Standard use different right-hand coordinate systems. Table 39-1 describes the two systems. Figure 39-1 illustrates them.

Table 39-1: Model coordinate systems

Coordinate	OpenSceneGraph	DIS
X	Right	Forward
Y	Forward	Right
Z	Up	Down



OSG coordinate system DIS coordinate system

Figure 39-1. Model coordinate systems

When you create models, it is recommended that you use the OpenSceneGraph coordinate system. The simulation application will convert the coordinates, as follows:

- xmodel = yDIS.
- ymodel = xDIS.
- zmodel = -zDIS.

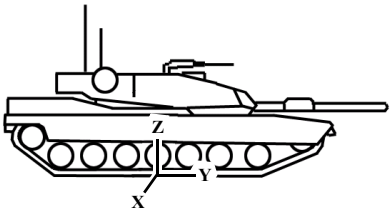
Location of the Origin for Models

The origin of the model's coordinate system should be located as follows, depending on the domain of the entity:

- Land: on the base of the vehicle, on the axis of yaw rotation.
- Surface: On the axis of the center of buoyancy, in the plane of the waterline for unloaded ship.
- Subsurface: On the axis of the center of buoyancy, in the plane of the waterline for surfaced submarine.
- Air and Space: At the center of gravity.

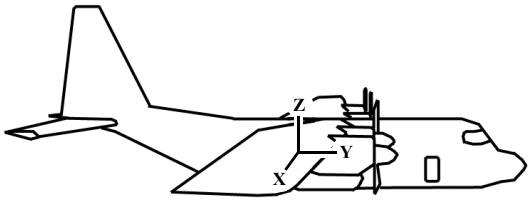
Figure 39-2 illustrates the location of a model's origin for each platform domain.

Land



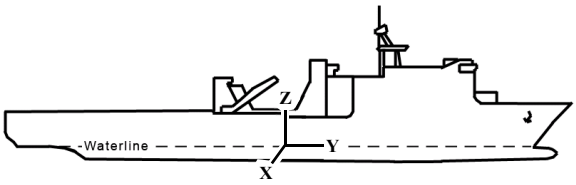
On the plane of ground contact, at the axis of yaw rotation.

Air



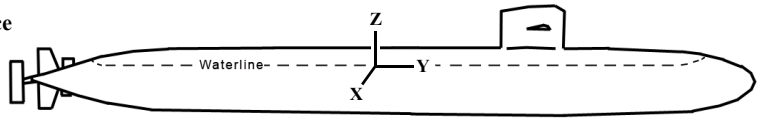
At the center of gravity.

Surface



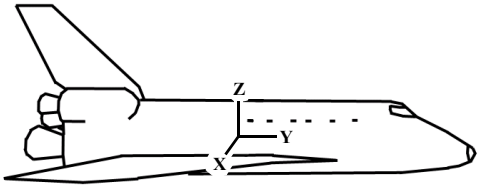
On the axis of the center of buoyancy, in the plane of the waterline for an unloaded ship

Subsurface



On the axis of the center of buoyancy, in the plane of the waterline of a surfaced submari

Space



At the center of gravity.

Figure 39-2. Location of origin by domain

39.4. Articulated Parts

In order for VR-Vantage to properly depict the state of articulated parts, models must conform to the following specifications:

- There must be one DOF node for each articulated part. (This means that if there are several LODs, they should be children of the DOF.) Each articulation DOF should be a descendant of the DOF for the part to which it is attached (unless the parent part is just the base model itself). For example, the DOF for a gun should be a descendant of the DOF for a turret.
- Each articulated part DOF node must have all of the geometry and other nodes associated with the part underneath it. This DOF node must have a comment associated with it. The comment must be of the form:

  `@dis articulated_part` *`part-number`*

  where *`part-number`* is the enumeration for the particular type of part that this part of the model represents. For example, the part number of a turret is 4096, and the part number of a gun is 4416.

  Therefore, the child group of a turret's DOF must be commented with:

  ```
  @dis articulated_part 4096
  ```

  The child group of a gun's DOF must be commented with:

  ```
  @dis articulated_part 4416
  ```

- The WRM Model Specification specifies that information about the legal motions of each articulated part be encoded in the comment field. Currently, VR-Vantage does not use this information.
- Do not include static offsets in the DOF attributes. That is, when you set all of the DOF parameters (X, Y, Z, heading, pitch, roll) to 0, the model should appear in its neutral position.

39.5. Entity Appearances

VR-Vantage supports switching between different pieces of a model's geometry based on appearance attributes. For instance, a tank's turret should look different if the hatch is open or closed. An entire tank should look different if it is damaged or destroyed.

Switch nodes are used to represent a choice between several different children, based on a particular appearance attribute.

We define a switch node as a group node with a comment of the form:

```
@dis switch attribute
```

where *attribute* is the name of the attribute that should be used to select a child. (For example, damage, smoke, hatch, and so on.)

The children of the switch node represent the different portions of a model among which VR-Vantage will select, based on the appearance attribute. If a switch node is commented with:

```
@dis switch hatch
```

it may, for example, have three children containing geometry for a closed hatch, popped hatch, and open hatch.

These children of the switch node must be commented so that VR-Vantage knows which child matches which values of the appearance attribute. The comment is of the form:

```
@dis state value-list
```

value-list is of the form *m[,m]+* where *m* is one of the following:

- An integer representing the enumeration of a value that an appearance attribute may take on, for example, 1 for hatch closed.
- Two integers, a and b, separated by a hyphen, representing all enumerations from a through b, for example, 2-5 for all states whose enumerations are 2, 3, 4, or 5.
- The integer -1, which indicates that this child is the default child, to be used when the attribute state matches no values in any children.

For instance, the child of the hatch switch node that represents a closed hatch might be used when the hatch attribute value is either 0 (not-applicable), 1 (closed), or any value other than those enumerated in other children. This child would be commented:

```
@dis state 0,1,-1
```

The child with the popped hatch geometry should be used when the value is 2 (popped) or 3 (popped with person visible) and is commented:

```
@dis state 2,3
```

The child that contains the open hatch geometry should be used when the value is 4 (open) or 5 (open with person visible) and is commented:

```
@dis state 4,5
```

There may be many switch nodes for the same attribute. For example, there may be several pieces of a tank's geometry that are affected by the destroyed attribute.

The WRM Model Specification also discusses the inheritance of appearance attributes from ancestor nodes. VR-Vantage does not support this functionality. To tell VR-Vantage to, for example, display a camouflaged tank when the paint scheme attribute is set to camouflage, and a normal tank otherwise, you must use the switch node method described previously.

In Figure 39-3, the turret DOF node is attached to the body Group node. The barrel, gun, and hatch DOF nodes are attached to the turret. This allows the turret to be positioned in the coordinate system of the body, and the barrel, gun, and hatch to be positioned in the coordinate system of the turret.

This model also contains a switch node that is flagged with animation. The comment on this node is `@dis switch moving`. This switches between two Group nodes – Stop and Go. Under the Stop node there is a single mesh of the treads of the tank. Under the Go node there is a Group node, with the animation flag set on. This creates a flipbook animation of all nodes below this Group node. There are three meshes in different states of moving: Frame 1, Frame 2 and Frame 3. At runtime OpenSceneGraph will animate these meshes when the velocity is not 0.

If multiple color schemes are needed to represent camouflage, winter, and desert versions, another switch is needed.

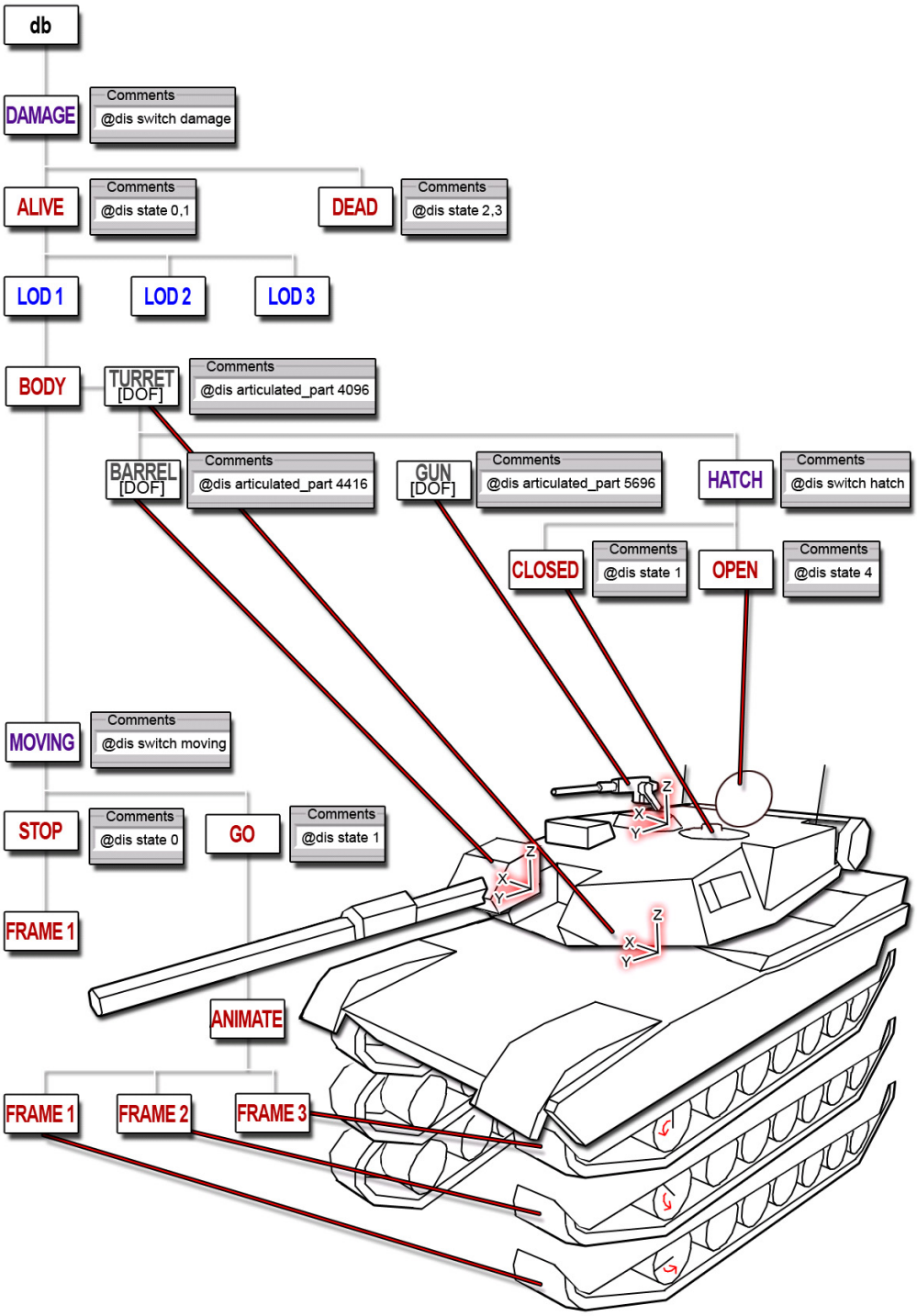


Figure 39-3. 3D model structure

39.6. Animation

By default, animation sequences are displayed as quickly as possible. However, you can configure the animation sequence rate using a WRM animation comment in the animation node's comment field.

The comment syntax is as follows:

```
@dis animation num_frames_per_second [ skip | noskip ]
```

where:

- `animation` tells VR-Vantage that animation instructions follow.
- *`num_frames_per_second`* is the rate at which VR-Vantage should display the frames.
- `skip` and `noskip` indicate whether the elapsed time should be used to choose the next frame, or whether at most one frame should be advanced each frame.

You can also specify how many times to loop through an animation and how long to hold the last frame with the following comments:

- `@dis loopCount` *`count`*, where *`count`* is the number of times to loop through the animation. Default: -1 (loop forever). 0 means do not loop.
- `@dis lastFrameTime` *`time`*, where *`time`* specifies how long, in seconds, to hold the last frame of the animation.

The `loopCount` and `lastFrameTime` comments only work with non-instanced models. You must use them with an `animation` comment, for example:

```
@dis animation 2 skip @dis loopCount 3 @dis lastFrameTime 11.5
```

39.7. DIS Keyword Values

Table 39-6 through Table 39-11 list the values that you can use for the `@dis switch` keyword by domain. For any switch keyword, the supported state values (`@dis state [value 1],[value 1]...`) are listed in the State column. The Bit column lists the appearance bit for setting DIS appearances. (Some bit number are repeated because they are valid only for a particular platform.)

- DIS appearance and states - land domain
- DIS appearance and states - air domain
- DIS appearance and states - surface domain
- DIS appearance and states - subsurface domain
- DIS appearance and states - space domain
- DIS appearance and states - lifeform domain
- DIS appearance and states - munition domain
- DIS appearance and states - environmental domain
- DIS appearance and states - cultural feature
- DIS appearance and states - sensor domain
- DIS Articulations.

Table 39-2: DIS appearance and states - land domain (Sheet 1 of 3)

Values	Description	State/ Description	Bit
blackout_brake_lights	State of blackout brake lights.	0 = Off 1 = On	27
blackout_lights	State of blackout lights.	0 = Off 1 = On	26
brake_lights	State of brake lights.	0 = Off 1 = On	14
camouflage_type	Type of camouflage.	0 = Desert 1 = Winter 2 = Forest 3 = Unused	17-18
concealed	Type of concealment.	0 = Not Concealed 1 = In Concealed Position	19
damage	Damaged appearance of an entity.	0 = None 1 = Slight 2 = Moderate 3 = Destroyed	3-4
fire_power	Characteristics of fire-power kill.	0 = No 1 = Yes	2

Table 39-2: DIS appearance and states - land domain (Sheet 2 of 3)

Values	Description	State/ Description	Bit
flaming	State of flames rising from an entity.	0 = No 1 = Yes	15
frozen_status	Frozen status of an entity.	0 = Not Frozen 1 = Frozen	21
hatch	State of the primary hatch.	0 = Not Applicable 1 = Closed 2 = Popped 3 = Popped and Person Visible 4 = Open 5 = Open and Person Visible 6 = Custom1 7 = Custom2	9-11
head_lights	State of headlights.	0 = No 1 = Yes	12
interior_lights	State of interior lights.	0 = Off 1 = On	29
launcher	State of the platform's primary launcher.	0 = Not Raised 1 = Raised	16
masked_cloaked	Describes whether or not the entity is masked or cloaked.	0 = No 1 = Yes	31
mobility	Characteristics of mobility kill.	0 = No 1 = Yes	1
paint_scheme	Paint scheme of an entity.	0 = Uniform 1 = Camouflage	0
power_plant_status	State of the entity’s power-plant.	0 = Off 1 = On	22
ramp	State of a ramp.	0 = Down 1 = Up	25
smoke	State or location of smoke and vapor.	0 = None 1 = Plume 2 = Engine 3 = Engine and Plume	5-6
spot_lights	Describes whether spot-lights are on or off.	0 = Off 1 = On	28

Table 39-2: DIS appearance and states - land domain (Sheet 3 of 3)

Values	Description	State/ Description	Bit
tail_lights	State of tail lights.	0 = Off 1 = On	13
tent	State of a tent extension.	0 = Not Extended 1 = Extended	24
trailing_effects	Size of the trailing effect for the entity.	0 = None 1 = Small 2 = Medium 3 = Large	7-8

Table 39-3: DIS appearance and states - air domain (Sheet 1 of 2)

Values	Description	State/ Description	Bit
afterburner	State of an air platform's afterburner.	0 = Off 1 = On	16
anticollision_lights	State of anti-collision lights.	0 = Off 1 = On	14
damage	Damaged appearance of an entity.	0 = None 1 = Slight 2 = Moderate 3 = Destroyed	3-4
flaming	State of flames rising from an entity.	0 = No 1 = Yes	15
formation_lights	State of formation lights.	0 = Off 1 = On	24
frozen_status	Frozen status of an entity.	0 = Not Frozen 1 = Frozen	21
hatch	State of the primary hatch.	0 = Not Applicable 1 = Closed 2 = Popped 3 = Popped and Person Visible 4 = Open 5 = Open and Person Visible 6 = Custom1 7 = Custom2	9-11
interior_lights	State of interior lights.	0 = Off 1 = On	29

Table 39-3: DIS appearance and states - air domain (Sheet 2 of 2)

Values	Description	State/ Description	Bit
landing_lights	State of landing lights.	0 = Off 1 = On	12
lights	State of lights.	0 = Off 1 = On	
mobility	Characteristics of mobility kill.	0 = No 1 = Yes	1
navigation_lights	State of navigation lights.	0 = Off 1 = On	13
paint_scheme	Paint scheme of an entity.	0 = Uniform 1 = Camouflage	0
power_plant_status	State of the entity's power-plant.	0 = Off 1 = On	22
smoke	State or location of smoke and vapor.	0 = None 1 = Plume 2 = Engine 3 = Engine and Plume	5-6
spot_lights	Describes whether spot-lights are on or off.	0 = Off 1 = On	28
trailing_effects	Size of the trailing effect for the entity.	0 = None 1 = Small 2 = Medium 3 = Large	7-8

Table 39-4: DIS appearance and states - surface domain (Sheet 1 of 2)

Values	Description	State/ Description	Bit
damage	Damaged appearance of an entity.	0 = None 1 = Slight 2 = Moderate 3 = Destroyed	3-4
flaming	State of flames rising from an entity.	0 = No 1 = Yes	15
frozen_status	Frozen status of an entity.	0 = Not Frozen 1 = Frozen	21
interior_lights	State of interior lights.	0 = Off 1 = On	29

Table 39-4: DIS appearance and states - surface domain (Sheet 2 of 2)

Values	Description	State/ Description	Bit
lights	State of lights.	0 = Off 1 = On	
mobility	Characteristics of mobility kill.	0 = No 1 = Yes	1
paint_scheme	Paint scheme of an entity.	0 = Uniform 1 = Camouflage	0
power_plant_status	State of the entity's power-plant.	0 = Off 1 = On	22
running_lights	State of running lights.	0 = Off 1 = On	12
smoke	State or location of smoke and vapor.	0 = None 1 = Plume 2 = Engine 3 = Engine and Plume	5-6
spot_lights	Describes whether spot-lights are on or off.	0 = Off 1 = On	28
trailing_effects	Size of the trailing effect for the entity.	0 = None 1 = Small 2 = Medium 3 = Large	7-8

Table 39-5: DIS appearance and states - subsurface domain (Sheet 1 of 2)

Values	Description	State/ Description	Bit
damage	Damaged appearance of an entity.	0 = None 1 = Slight 2 = Moderate 3 = Destroyed	3-4
flaming	State of flames rising from an entity.	0 = No 1 = Yes	15
frozen_status	Frozen status of an entity.	0 = Not Frozen 1 = Frozen	21
hatch	State of the primary hatch.	0 = Not Applicable 1 = Closed 2 = Popped	9-11

Table 39-5: DIS appearance and states - subsurface domain (Sheet 2 of 2)

Values	Description	State/ Description	Bit
		3 = Popped and Person Visible 4 = Open 5 = Open and Person Visible 6 = Custom1 7 = Custom2	
mobility	Characteristics of mobility kill.	0 = No 1 = Yes	1
paint_scheme	Paint scheme of an entity.	0 = Uniform 1 = Camouflage	0
power_plant_status	State of the entity's power-plant.	0 = Off 1 = On	22
running_lights	State of running lights.	0 = Off 1 = On	12
smoke	State or location of smoke and vapor.	0 = None 1 = Plume 2 = Engine 3 = Engine and Plume	5-6

Table 39-6: DIS appearance and states - space domain (Sheet 1 of 2)

Values	Description	State/ Description	Bit
damage	Damaged appearance of an entity.	0 = None 1 = Slight 2 = Moderate 3 = Destroyed	3-4
flaming	State of flames rising from an entity.	0 = No 1 = Yes	15
frozen_status	Frozen status of an entity.	0 = Not Frozen 1 = Frozen	21
mobility	Characteristics of mobility kill.	0 = No 1 = Yes	1
paint_scheme	Paint scheme of an entity.	0 = Uniform 1 = Camouflage	0

Table 39-6: DIS appearance and states - space domain (Sheet 2 of 2)

Values	Description	State/ Description	Bit
power_plant_status	State of the entity's power-plant.	0 = Off 1 = On	22
smoke	State or location of smoke and vapor.	0 = None 1 = Plume 2 = Engine 3 = Engine and Plume	5-6

Table 39-7: DIS appearance and states - lifeform domain (Sheet 1 of 2)

Values	Description	State/ Description	Bit
camouflage_type	Type of camouflage.	0 = Desert 1 = Winter 2 = Forest 3 = Unused	17-18
compliance	Compliance of lifeform.	0 = Lifeform None 1 = Detained 2 = Surrender 3 = Using Fists 4 = Verbal Abuse 1 5 = Verbal Abuse 2 6 = Verbal Abuse 3 7 = Passive Resistance 1 8 = Passive Resistance 2 9 = Passive Resistance 3 10 = Non-Lethal Weapon 1 11 = Non-Lethal Weapon 2 12 = Non-Lethal Weapon 3 13 = Non-Lethal Weapon 4 14 = Non-Lethal Weapon 5 15 = Non-Lethal Weapon 6	
concealed	Type of concealment.	0 = Not Concealed 1 = In Concealed Position	19
damage	Damaged appearance of an entity.	0 = None 1 = Slight 2 = Moderate 3 = Destroyed	3-4

Table 39-7: DIS appearance and states - lifeform domain (Sheet 2 of 2)

Values	Description	State/ Description	Bit
flashlight	State of lights.	0 = Off 1 = On	
frozen_status	Frozen status of an entity.	0 = Not Frozen 1 = Frozen	21
life_form_state	Lifeform action.	0 = None 1 = Upright Still 2 = Upright Walking 3 = Upright Running 4 = Kneeling 5 = Prone 6 = Crawling 7 = Swimming 8 = Parachuting 9 = Jumping 10 = Sitting 11 = Squatting 12 = Crouching 13 = Wading 14 = Surrender 15 = Detained	
paint_scheme	Paint scheme of an entity.	0 = Uniform 1 = Camouflage	0
weapon_1	State of primary weapon.	0 = None 1 = Stowed 2 = Deployed 3 = Firing Position	
weapon_2	State of secondary weapon.	0 = None 1 = Stowed 2 = Deployed 3 = Firing Position	

Table 39-8: DIS appearance and states - munition domain

Values	Description	State/ Description	Bit
damage	Damaged appearance of an entity.	0 = None 1 = Slight 2 = Moderate 3 = Destroyed	3-4
flaming	State of flames rising from an entity.	0 = No 1 = Yes	15
frozen_status	Frozen status of an entity.	0 = Not Frozen 1 = Frozen	21
launch_flash	Describes the presence of a guided munitions launch flash.	0 = No 1 = Yes	
masked_cloaked	Describes whether or not the entity is masked or cloaked.	0 = No 1 = Yes	31
power_plant_status	State of the entity's power-plant.	0 = Off 1 = On	22
smoke	State or location of smoke and vapor.	0 = None 1 = Plume 2 = Engine 3 = Engine and Plume	5-6
trailing_effects	Size of the trailing effect for the entity.	0 = None 1 = Small 2 = Medium 3 = Large	7-8

Table 39-9: DIS appearance and states - environmental domain

Values	Description	State/ Description	Bit
density	Density of environmental.	0 = Clear 1 = Hazy 2 = Dense 3 = Very Dense 4 = Opaque 5 = Unused	
frozen_status	Frozen status of an entity.	0 = Not Frozen 1 = Frozen	21
masked_cloaked			

Table 39-10: DIS appearance and states - cultural feature

Values	Description	State/ Description	Bit
damage	Damaged appearance of an entity.	0 = None 1 = Slight 2 = Moderate 3 = Destroyed	3-4
exterior_lights_on	Describes whether the exterior lights are on or off.	0 = Off 1 = On	28
flaming	State of flames rising from an entity.	0 = No 1 = Yes	15
frozen_status	Frozen status of an entity.	0 = Not Frozen 1 = Frozen	21
interior_lights	State of interior lights.	0 = Off 1 = On	29
internal_heat_on	Describes whether the internal heat is on or off.	0 = Off 1 = On	22
masked_cloaked	Describes whether or not the entity is masked or cloaked.	0 = No 1 = Yes	31
smoke	State or location of smoke and vapor.	0 = None 1 = Plume 2 = Engine 3 = Engine and Plume	5-6

Table 39-11: DIS appearance and states - sensor domain (Sheet 1 of 2)

Values	Description	State/ Description	Bit
antenna_raised	Describes whether the antenna is raised or not	0 = Lowered 1 = Raised	16
blackout_lights	State of blackout lights.	0 = Off 1 = On	26
concealed	Type of concealment.	0 = Not Concealed 1 = In Concealed Position	19
camouflage_type	Type of camouflage.	0 = Desert 1 = Winter 2 = Forest 3 = Unused	17-18

Table 39-11: DIS appearance and states - sensor domain (Sheet 2 of 2)

Values	Description	State/ Description	Bit
damage	Damaged appearance of an entity.	0 = None 1 = Slight 2 = Moderate 3 = Destroyed	3-4
flaming	State of flames rising from an entity.	0 = No 1 = Yes	15
frozen_status	Frozen status of an entity.	0 = Not Frozen 1 = Frozen	21
interior_lights	State of interior lights.	0 = Off 1 = On	29
lights	State of lights.	0 = Off 1 = On	
mission_killed	Ability to carry out mission.	0 = No 1 = Yes	2
mobility	Characteristics of mobility kill.	0 = No 1 = Yes	1
paint_scheme	Paint scheme of an entity.	0 = Uniform 1 = Camouflage	0
power_plant_status	State of the entity's power-plant.	0 = Off 1 = On	22
smoke	State or location of smoke and vapor.	0 = None 1 = Plume 2 = Engine 3 = Engine and Plume	5-6
tent	State of a tent extension.	0 = Not Extended 1 = Extended	24
trailing_effects	Size of the trailing effect for the entity.	0 = None 1 = Small 2 = Medium 3 = Large	7-8

Table 39-12 lists the supported values for the `@dis articulated_part` keyword. String values are part of the WRM specification, but are not currently supported.

Table 39-12: DIS Articulations (Sheet 1 of 5)

Value	String	Description
3296	bridge_launcher	Position of the bridge launcher.
3328	bridge_section_1	Position of bridge section 1.
3360	bridge_section_2	Position of bridge section 2.
3392	bridge_section_3	Position of bridge section 3.
7616	fuselage_fold	Fold on plane's body.
1184	helicopter_main_rotor	Describes main rotor rotation.
1216	helicopter_tail_rotor	Describes tail rotor rotation.
3072	landing_gear	Landing gear on a plane.
3520	launcher_arm_primary	Position of the primary launcher arm.
1120	left_aileron	Position of the left aileron.
1056	left_flap	Position of the left flap.
3168	left_weapon_bay_door	Left door to weapon bay.
3552	other	Position of other articulated parts.
3424	primary_blade_1	Position of the primary blade 1.
3456	primary_blade_2	Position of the primary blade 2.
3488	primary_boom	Position of the primary boom.
5056	primary_defense_systems_1	Position of the primary defense systems 1.
5088	primary_defense_systems_2	Position of the primary defense systems 2.
5120	primary_defense_systems_3	Position of the primary defense systems 3.
5152	primary_defense_systems_4	Position of the primary defense systems 4.
5184	primary_defense_systems_5	Position of the primary defense systems 5.
5216	primary_defense_systems_6	Position of the primary defense systems 6.
5248	primary_defense_systems_7	Position of the primary defense systems 7.
5280	primary_defense_systems_8	Position of the primary defense systems 8.
5312	primary_defense_systems_9	Position of the primary defense systems 9.
5344	primary_defense_systems_10	Position of the primary defense systems 10.
4416	primary_gun_number_1	Position of the primary gun number 1.
4448	primary_gun_number_2	Position of the primary gun number 2.
4480	primary_gun_number_3	Position of the primary gun number 3.
4512	primary_gun_number_4	Position of the primary gun number 4.

Table 39-12: DIS Articulations (Sheet 2 of 5)

Value	String	Description
4544	primary_gun_number_5	Position of the primary gun number 5.
4576	primary_gun_number_6	Position of the primary gun number 6.
4608	primary_gun_number_7	Position of the primary gun number 7.
4640	primary_gun_number_8	Position of the primary gun number 8.
4672	primary_gun_number_9	Position of the primary gun number 9.
4704	primary_gun_number_10	Position of the primary gun number 10.
4736	primary_launcher_1	Position of primary launcher 1.
4768	primary_launcher_2	Position of primary launcher 2.
4800	primary_launcher_3	Position of primary launcher 3.
4832	primary_launcher_4	Position of primary launcher 4.
4864	primary_launcher_5	Position of primary launcher 5.
4896	primary_launcher_6	Position of primary launcher 6.
4928	primary_launcher_7	Position of primary launcher 7.
4960	primary_launcher_8	Position of primary launcher 8.
4992	primary_launcher_9	Position of primary launcher 9.
5024	primary_launcher_10	Position of primary launcher 10.
5376	primary_radar_1	Position of the primary radar 1.
5408	primary_radar_2	Position of the primary radar 2.
5440	primary_radar_3	Position of the primary radar 3.
5472	primary_radar_4	Position of the primary radar 4.
5504	primary_radar_5	Position of the primary radar 5.
5536	primary_radar_6	Position of the primary radar 6.
5568	primary_radar_7	Position of the primary radar 7.
5600	primary_radar_8	Position of the primary radar 8.
5632	primary_radar_9	Position of the primary radar 9.
5664	primary_radar_10	Position of the primary radar 10.
4096	primary_turret_number_1	Position of the primary turret number 1.
4128	primary_turret_number_2	Position of the primary turret number 2.
4160	primary_turret_number_3	Position of the primary turret number 3.
4192	primary_turret_number_4	Position of the primary turret number 4.
4224	primary_turret_number_5	Position of the primary turret number 5.
4256	primary_turret_number_6	Position of the primary turret number 6.
4288	primary_turret_number_7	Position of the primary turret number 7.

Table 39-12: DIS Articulations (Sheet 3 of 5)

Value	String	Description
4320	primary_turret_number_8	Position of the primary turret number 8.
4352	primary_turret_number_9	Position of the primary turret number 9.
4384	primary_turret_number_10	Position of the primary turret number 10.
1152	right_aileron	Position of the right aileron.
1088	right_flap	Position of the right flap.
3200	right_weapon_bay_doors	Right door to the weapon bay.
7584	rotor_fold	Fold on helicopters rotors.
1024	rudder	Position of the rudder.
6656	secondary_defense_systems_1	Position of the secondary defense systems 1.
6688	secondary_defense_systems_2	Position of the secondary defense systems 2.
6720	secondary_defense_systems_3	Position of the secondary defense systems 3.
6752	secondary_defense_systems_4	Position of the secondary defense systems 4.
6784	secondary_defense_systems_5	Position of the secondary defense systems 5.
6816	secondary_defense_systems_6	Position of the secondary defense systems 6.
6848	secondary_defense_systems_7	Position of the secondary defense systems 7.
6880	secondary_defense_systems_8	Position of the secondary defense systems 8.
6912	secondary_defense_systems_9	Position of the secondary defense systems 9.
6944	secondary_defense_systems_10	Position of the secondary defense systems 10.
6016	secondary_gun_number_1	Position of the secondary gun number 1.
6048	secondary_gun_number_2	Position of the secondary gun number 2.
6080	secondary_gun_number_3	Position of the secondary gun number 3.
6112	secondary_gun_number_4	Position of the secondary gun number 4.
6144	secondary_gun_number_5	Position of the secondary gun number 5.
6176	secondary_gun_number_6	Position of the secondary gun number 6.
6208	secondary_gun_number_7	Position of the secondary gun number 7.
6240	secondary_gun_number_8	Position of the secondary gun number 8.

Table 39-12: DIS Articulations (Sheet 4 of 5)

Value	String	Description
6272	secondary_gun_number_9	Position of the secondary gun number 9.
6304	secondary_gun_number_10	Position of the secondary gun number 10.
6336	secondary_launcher_1	Position of the secondary launcher 1.
6400	secondary_launcher_3	Position of the secondary launcher 3.
6432	secondary_launcher_4	Position of the secondary launcher 4.
6464	secondary_launcher_5	Position of the secondary launcher 5.
6496	secondary_launcher_6	Position of the secondary launcher 6.
6528	secondary_launcher_7	Position of the secondary launcher 7.
6560	secondary_launcher_8	Position of the secondary launcher 8.
6592	secondary_launcher_9	Position of the secondary launcher 9.
6624	secondary_launcher_10	Position of the secondary launcher 10.
6368	secondary_launcher_2	Position of the secondary launcher 2.
6976	secondary_radar_1	Position of secondary radar 1.
7008	secondary_radar_2	Position of secondary radar 2.
7040	secondary_radar_3	Position of secondary radar 3.
7072	secondary_radar_4	Position of secondary radar 4.
7104	secondary_radar_5	Position of secondary radar 5.
7136	secondary_radar_6	Position of secondary radar 6.
7168	secondary_radar_7	Position of secondary radar 7.
7200	secondary_radar_8	Position of secondary radar 8.
7232	secondary_radar_9	Position of secondary radar 9.
7264	secondary_radar_10	Position of secondary radar 10.
5696	secondary_turret_number_1	Position of the secondary turret number 1.
5728	secondary_turret_number_2	Position of the secondary turret number 2.
5760	secondary_turret_number_3	Position of the secondary turret number 3.
5792	secondary_turret_number_4	Position of the secondary turret number 4.
5824	secondary_turret_number_5	Position of the secondary turret number 5.
5856	secondary_turret_number_6	Position of the secondary turret number 6.
5888	secondary_turret_number_7	Position of the secondary turret number 7.
5920	secondary_turret_number_8	Position of the secondary turret number 8.
5952	secondary_turret_number_9	Position of the secondary turret number 9.
5984	secondary_turret_number_10	Position of the secondary turret number 10.

Table 39-12: DIS Articulations (Sheet 5 of 5)

Value	String	Description
3104	tail_hook	Tail hook on a plane for aircraft carrier landings.
7584	wing_fold	Fold on plane's wings.



IX. Graphics

This section discusses overlays, tactical graphics, and remote graphics.



40. Tactical Graphics and Remote Graphics

Tactical graphics are graphical objects that represent points, lines, and areal objects used for various purposes in simulations. Remote graphics are graphics that are drawn onto the scene by remote programs that use the VR-Vantage Remote Draw API. This chapter explains how to display them in VR-Vantage.

Displaying Tactical Graphics 40-2
Overlays 40-3
Showing and Hiding Tactical Graphics 40-4
Showing and Hiding Tactical Graphics Labels 40-5
Displaying Height-Above-Terrain Lines for Vertices 40-5
Displaying Remote Graphics 40-6
Viewing a List of Remote Graphics 40-7

40.1. Displaying Tactical Graphics

Tactical graphics are graphical objects that represent points, lines, areal objects, text, and symbols used for various purposes in simulations (Figure 40-1 and Figure 40-2). They could represent routes that simulation objects are following, a destination point, or a way to show some sort of relationship, such as height-above-terrain (HAT) lines. VR-Vantage can display tactical graphics published by VR-Forces.

You can configure the display of tactical graphics as follows:

- Enable and disable the display of tactical graphics.
- Show or hide vertex indicators for point, linear, and areal objects.
- Show or hide height-above-terrain lines.
- Display height-above-terrain lines only for selected graphics.

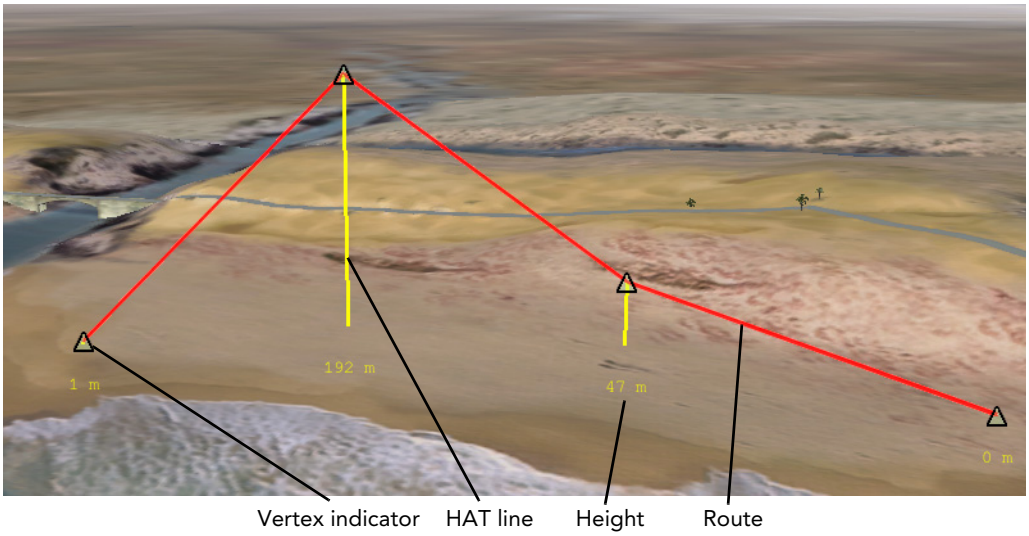


Figure 40-1. Tactical graphics with height-above-terrain lines (3D)

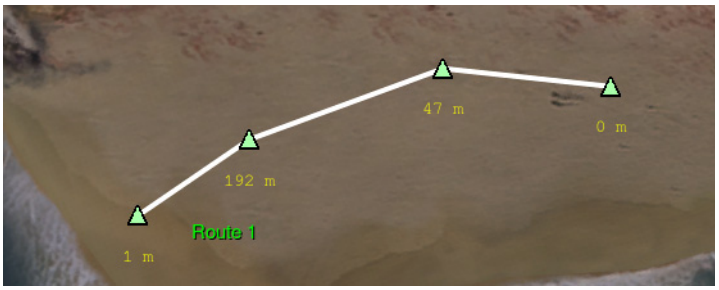


Figure 40-2. Tactical graphic (2D)

40.1.1. Overlays

All tactical graphics are assigned to overlays. Overlays are conceptually similar to clear plastic sheets that you might layer over a map so that you easily add and remove supplementary information.

There is always at least one overlay – the Default overlay. However, VR-Forces users can create multiple overlays. When VR-Vantage receives information about tactical graphics over the network, it lists the overlays and their respective graphics on the Overlays tab of the Objects List Panel (Figure 40-3).

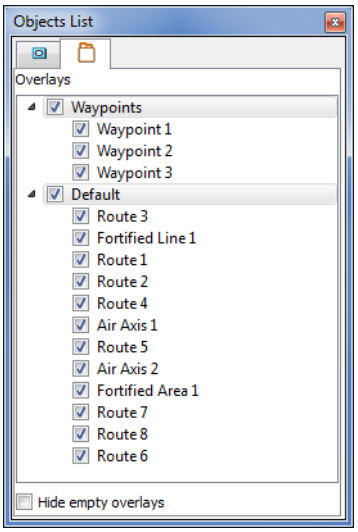


Figure 40-3. Overlays tab

40.1.2. Showing and Hiding Tactical Graphics

You can control the display of tactical graphics from the Display Settings dialog box, Tactical Graphics Display Settings page, from menu and toolbar options, and from the Objects List Panel Overlays tab.

To show or hide tactical graphics:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Tactical Graphics Display Settings page (Figure 40-4).

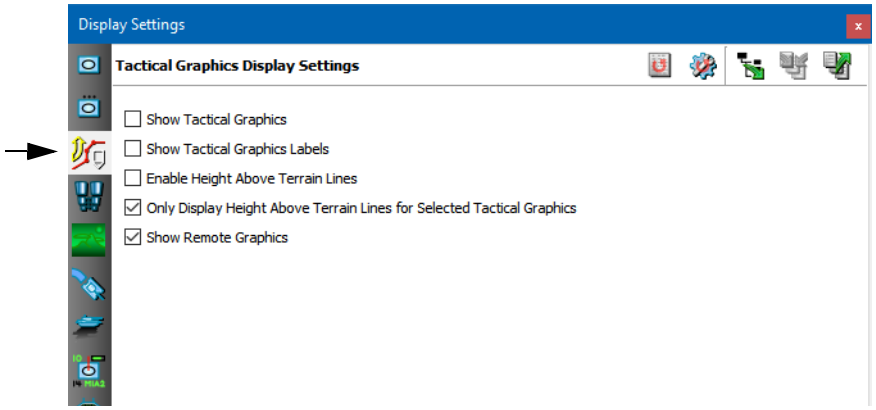


Figure 40-4. Tactical Graphics Display Settings page

3. Select or clear the Show Tactical Graphics check box.

In VR-Vantage Stealth and PVD, you can also enable and disable tactical graphics by choosing **Settings** → **Tactical Graphics** or clicking the Tactical Graphics button ( ) on the Display Settings Toolbar.

Showing and Hiding Tactical Graphics on the Overlays Tab

On the Overlays tab, you can show and hide tactical graphics individually and by overlay.

To control the display of tactical graphics from the Overlays tab:

1. On the Objects List Panel, select the Overlays tab (Figure 40-3).
2. To hide all graphics on a particular overlay, clear the check box for that overlay.
3. To hide a specific tactical graphic, clear its check box.
4. To redisplay a hidden tactical graphic or overlay, select its check box.

40.1.3. Showing and Hiding Tactical Graphics Labels

In addition to displaying the outlines of tactical graphics, you can display graphics (labels) for the vertices of these objects. Vertex labels are enabled in Figure 40-1 and Figure 40-2. In Plan View mode, VR-Vantage also displays a tactical graphic's name. For details about displaying labels for 2D icons, please see "Displaying Labels for 2D Icons," on page 19-7.



Displaying vertex labels can affect performance.

To show or hide vertex labels:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Tactical Graphics Display Settings page (Figure 40-4).
3. Select or clear the Show Tactical Graphics Labels check box.



In VR-Vantage Stealth and PVD, you can also enable and disable tactical graphics labels by choosing **Settings** → **Tactical Graphics Labels**.

40.1.4. Displaying Height-Above-Terrain Lines for Vertices

In 3D observer modes, you can display height-above-terrain (HAT) lines for the vertices of tactical graphics (Figure 40-1). These vertical lines between an object's vertices and the ground help you understand the relationship of the object to the terrain. The height for each vertex is displayed (using whatever units are configured on the Display Units Settings page).

HAT lines are shown only for tactical graphics that could be above or below the ground, such as routes, and waypoints. You can show and hide these lines and you can specify that they be shown for all objects or only for the selected objects.

Use of height above terrain lines can reduce the frame rate.

To show or hide height-above-terrain lines:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Tactical Graphics Display Settings page (Figure 40-4).
3. To display HAT lines, select the Enable Height Above Terrain Lines check box.
4. To only display lines for selected objects, select the Only Display Height Above Terrain Lines For Selected Tactical Graphics check box.

40.2. Displaying Remote Graphics

VR-Vantage can display graphics sent from other applications using the VR-Vantage Remote Draw API (Figure 40-5).

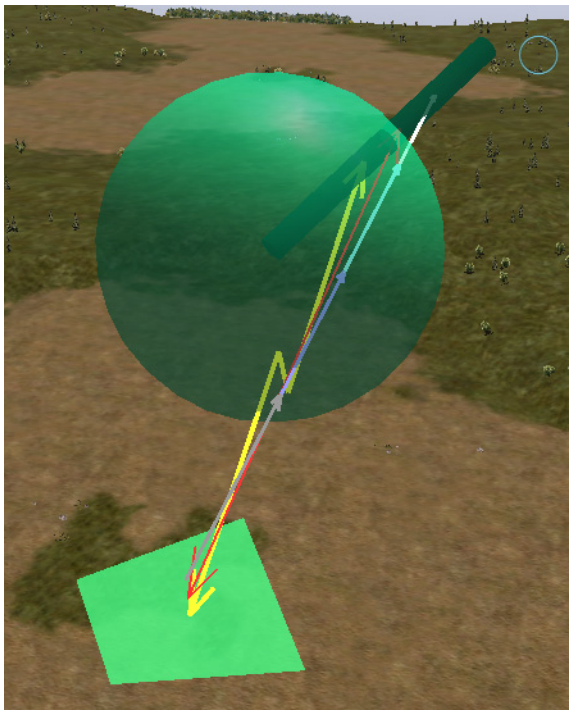


Figure 40-5. Remote graphics

To enable or disable the display of remote graphics:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Tactical Graphics Display Settings page (Figure 40-6).

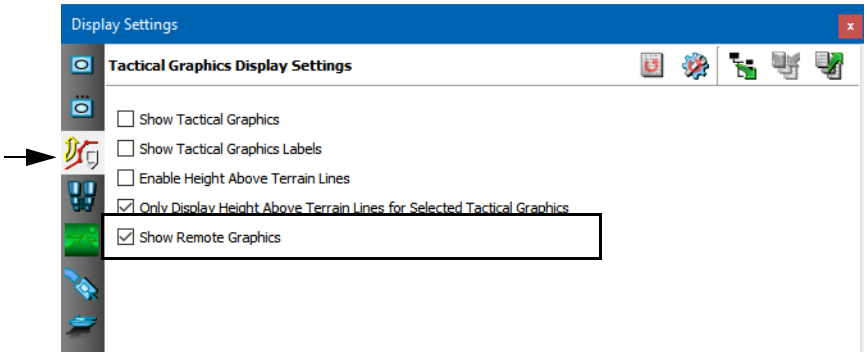


Figure 40-6. Tactical Graphics Display Settings page

3. Select or clear the Show Remote Graphics check box.



In VR-Vantage Stealth and PVD, you can also enable and disable remote graphics by choosing **Settings** → **Remote Graphics** or by clicking the Remote Graphics button ( ) on the Display Settings Toolbar.

40.2.1. Viewing a List of Remote Graphics

Remote graphics can be organized into sets. There is no fixed organization. Labeling of sets is entirely at the discretion of the programmer who develops the application that generates the graphics. The Remote Graphics Panel displays remote graphics in a hierarchy by set.

The VR-Vantage Remote Draw API lets developers send data about objects as attribute:value pairs, for example Speed:30 KM. The attributes are organized by type using attribute group templates. You can enable and disable the display of attribute information by group and by individual attribute. Figure 40-7 illustrates the attributes for the Entity and Packet attribute group templates displayed in Figure 40-8.

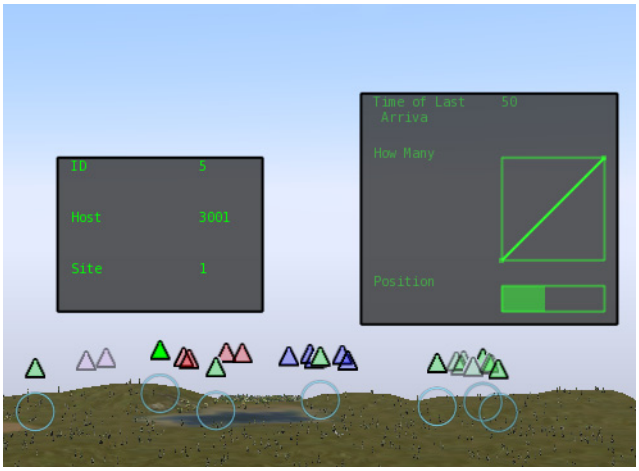


Figure 40-7. Remote graphics attribute group display

To view a list of remote graphics:

1. Choose **View** → **Remote Graphics Panel**. The Remote Graphics Panel is displayed (Figure 40-8). The Remote Graphics Sets tab lists remote graphics by set.

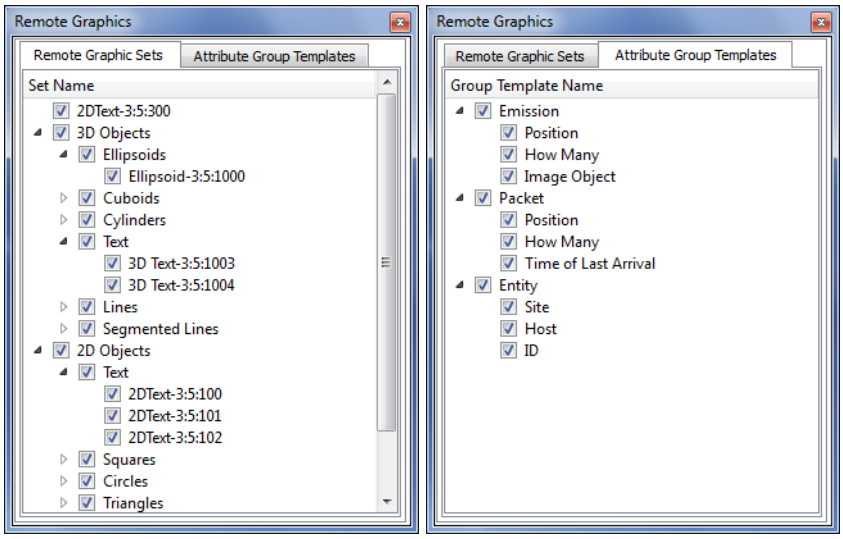


Figure 40-8. Remote Graphics Panel

2. To hide or display a remote graphic set or individual graphic, check or clear its check box in the Remote Graphics Sets list.
3. To view Attribute Group Templates, select the Attribute Group Templates tab. To see the attribute display, mouse over an object for which display data is available.



X. VR-Vantage Support for CIGI

VR-Vantage supports the Common Image Generator Interface (CIGI). The chapters in this section describe the extent of CIGI support. This includes a tutorial for using the included CIGI host application.



41. Mapping CIGI Models and Components

Unlike DIS or HLA, CIGI does not have a standard set of entity type mappings. Therefore, a CIGI connection cannot use the entity type mappings described elsewhere in this manual. VR-Vantage provides a set of mappings (MAK CIGI Mappings) for the entity models it includes. These mappings work with the CIGI host emulator provided with VR-Vantage. You can add new mappings, and you can edit the default mappings to work with other hosts. This chapter provides a minimal introduction to CIGI and the VR-Vantage CIGI driver and explains how to map CIGI components.

Introduction to CIGI 41-2
- The VR-Vantage CIGI Driver 41-2
- CIGI Packet Support 41-3
- Mapping CIGI Input to VR-Vantage 41-3

Mapping Entity Models for CIGI 41-4
- Adding an Articulated Part Mapping to an Entity Model 41-5

Mapping CIGI Components 41-6
- Component Control Message Formats 41-6

Mapping a Database 41-11

Mapping CIGI Views 41-12
- Mapping a CIGI View to an Observer Mode 41-12

Prototypical Host – IG Configurations 41-13

41.1. Introduction to CIGI

CIGI (Common Image Generator Interface) is an interface that allows an image generator (IG) to receive messages from a host to control what is drawn by the IG. CIGI defines packets that are sent between the host and the IG. CIGI is an open source project. It provides an interface control document and an open source library to send and receive CIGI messages: the CIGI Class Library (CCL). The VR-Vantage CIGI driver uses the CCL. VR-Vantage ships with the open-source CIGI Host Emulator. The host emulator and CCL are available at http://cigi.sourceforge.net. Chapter 42, *CIGI Host Emulator Tutorial* describes how to use the host emulator to control VR-Vantage.

41.1.1. The VR-Vantage CIGI Driver

The VR-Vantage CIGI driver supports best-effort (UDP) and reliable (TCP) communication between the CIGI driver and the host. When VR-Vantage uses a best-effort connection, it opens a UDP socket to the host port and tries to exchange data with the host. When it uses a reliable connection, the CIGI driver can act as a TCP client or TCP server. If it runs as a client, on connection the CIGI Driver tries to connect to a TCP server port on the host. If the connection fails, the driver returns to the disconnected state. If it runs as a server, it opens a TCP server socket on the listen port, and it accepts any connections made to that port. Data is then sent on that connection.

CIGI supports two modes of operation: synchronous and asynchronous. In synchronous mode, the IG sends a CIGI Start of Frame packet to the host, which prompts the host to send it data. The data is processed by the IG and is rendered; then the next frame starts. In asynchronous mode, the host sends data at any point and the IG renders the data. Since data may not be received every frame, the IG usually uses dead-reckoning and smoothing to update the positions of the entities until it receives an update.

The VR-Vantage CIGI driver supports synchronous and asynchronous modes. Synchronous mode can be run in single-threaded or multi-threaded mode, as follows:

- In single-threaded mode, the driver runs in the same thread as VR-Vantage, so its rate is bound by the VR-Vantage frame-rate. For example, if the driver is set to run at 60 Hz, but VR-Vantage has a frame rate of 20, the driver will only be able to run at 20 Hz. However, if the driver rate is set to 30 Hz, and VR-Vantage is running at 60, the CIGI Driver will only process CIGI messages at 30 Hz.
- In multi-threaded mode, the driver and VR-Vantage run in separate threads. The driver tries to maintain the desired rate. It sends start of frame messages, processes received messages, and sends messages to the main application to be rendered. Although this is not a true synchronous mode, it is the recommended synchronous mode for performance reasons.

41.1.2. CIGI Packet Support

VR-Vantage can receive the following CIGI packets:

- IG Control.
- Entity Control. Collision Detection is not supported.
- Component Control.
- Short Component Control.
- Articulated Part Control.
- Short Articulated Part Control.
- Rate Control.
- Celestial Sphere Control. Only date/time is supported.
- Weather Control. Only rain and snow are supported.
- View Control.
- View Definition. Mirror, Pixel Replication, and Orthographic projection are not supported.
- HAT/HOT Request.
- Line of Sight Segment Request.
- Line of Sight Vector Request.
- User-defined packet. VR-Vantage supports a user-defined packet for CIGI. For details, please see *DtCigiDiGuyCtrl.h*.

VR-Vantage can send the following packets:

- Start of Frame.
- HAT/HOT Response.
- Line of Sight Response.

For a complete list of supported and unsupported fields for each packet type, please see Chapter 44, *CIGI Packets Supported*.

41.1.3. Mapping CIGI Input to VR-Vantage

You map CIGI input to VR-Vantage models and other data files in the CIGI Mappings dialog box. You can map the following objects and components:

- CIGI entity models.
- CIGI entity components.
- CIGI system components.
- CIGI view components.
- CIGI model states.
- CIGI model animations.
- CIGI database mappings.
- CIGI regional sea surface components.
- CIGI observer mode mappings.
- CIGI active sensor regions.

41.2. Mapping Entity Models for CIGI

To view entity models using CIGI, you must map the visual models to the entity types known to the CIGI host. You can also map articulated parts for entities.

To map an entity model to a CIGI entity type:

1. Choose **Settings** → **CIGI Mappings**. The CIGI Mappings dialog box opens.
2. Select the CIGI Entity Models page (Figure 41-1).

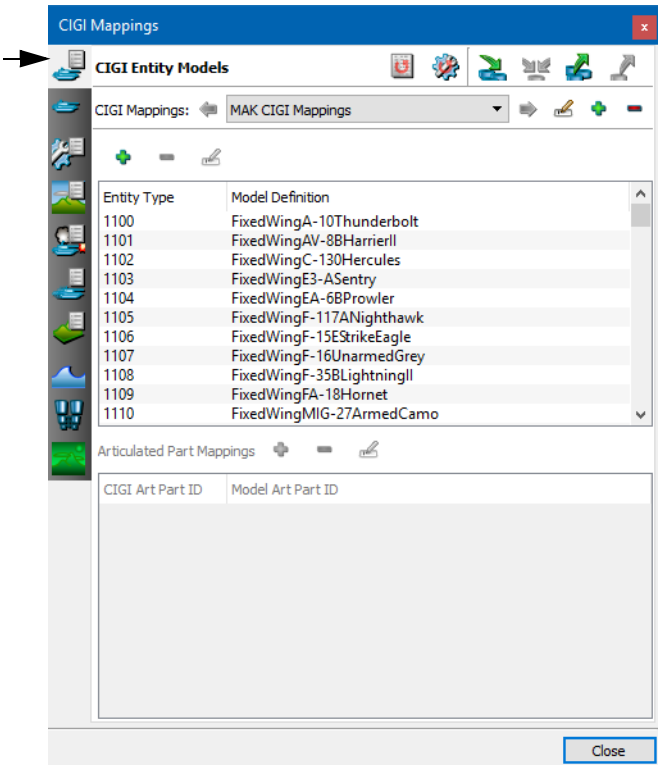


Figure 41-1. CIGI Entity Models page

3. Click the Add button (+) above the list of entity types. The Create New Entity Type Model Mapping dialog box opens (Figure 41-2).

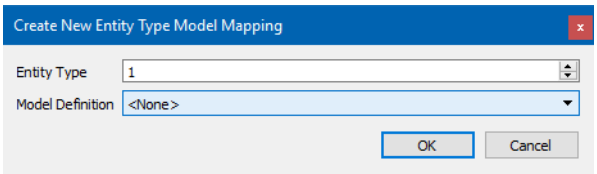


Figure 41-2. Create New Entity Type Model Mapping dialog box

4. Type an integer in the Entity Type box.
5. Select a model definition in the Model Definition list.

6. Click OK. The mapping is added to the list.

41.2.1. Adding an Articulated Part Mapping to an Entity Model

You can map articulated parts to an entity.

To add an articulated part mapping to an entity:

1. Choose **Settings** → **CIGI Mappings**. The CIGI Mappings dialog box opens.
2. Select the CIGI Entity Models page (Figure 41-1).
3. In the list of entity types, select the entity to which you want to add an articulated part. The Articulated Part Mappings window becomes active.
4. Click the Add button (+) next to the Articulated Part Mappings label. The Create New Art Part Mapping dialog box opens (Figure 41-3).

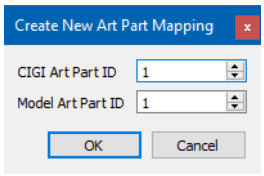


Figure 41-3. Create New Art Part Mapping dialog box

5. In the CIGI Art Part ID box, type an integer. This is an arbitrary number that must match whatever value the CIGI host uses.
6. In the Model Art Part ID box, type an integer. This value is the DIS comment node in the model.
7. Click OK.
8. The mapping is added to the Articulated Part Mappings window.

41.3. Mapping CIGI Components

VR-Vantage can process the following CIGI control messages:

- Entity component control message. Lets you specify model state, animation frame, animation range, and animation speed.
- View component control message. Lets you specify wireframe and whether or not to display the performance statistics overlay.
- System component control message. Lets you specify display of shadows.

The various component pages in the CIGI Mapping Settings dialog box let you specify the component IDs for these messages. The model state mappings specify the data for entity component controls.

For each type of component, if there are two or fewer data fields being used, you can use a component control or short component control message to set the component data. If there are more than two data fields in use, you cannot use a short component control.

The process for mapping components is similar for each component type.

To map a component ID:

1. Choose **Settings** → **CIGI Mappings**. The CIGI Mappings dialog box opens (Figure 41-1).
2. Select the page for the type of component you want to map.
3. Click the Add button (+) above the mapping window. The Create New Component Mapping dialog box opens (Figure 41-4).

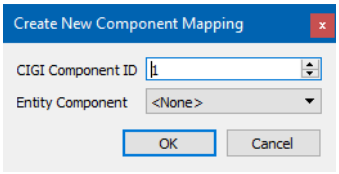


Figure 41-4. Create New Component Mapping dialog box

4. Type an integer value for the component ID. This is the value that will be used in the Component ID field of a component control message.
5. Complete the dialog box as follows:
   - For entity, view, or system components, select the component type from the list. The options vary for the different component types.
   - For animation controls, type the name of an animation.
   - For model state, select an appearance from the list.
6. Click OK. The mapping is added to the window.

41.3.1. Component Control Message Formats

This section describes the fields of the component control messages that VR-Vantage supports.

Entity Component Controls

You can use entity component controls to set the state in entity models and manipulate animations that are built into the model.

Model State

Model states are the equivalent of DIS appearance states. A model state component control message takes a component ID for a model state as configured on the Entity Components page and a model state mapping as configured on the CIGI Model State mappings page.

Table 41-1: Model state component control

Component Class	Entity (0).
Instance ID	Entity ID.
Component ID	ID for Model State, as set on the Entity Components Mappings page.
Component State	The value of the state.
Data 1	Model State, as set in CIGI Model State mappings page (int).
Data 2	Model State, as set in CIGI Model State mappings page (int).



Multiple nodes can be set to the same state through multiple data fields. You can set up to two states with a Short Component Control message or up to six with a Component Control Message.

Animation Control

Entity models can contain named animation nodes. You can set the speed, range, and current frame. The animation node in the model is designated by a string. A mapping from an integer to this string must be specified on the CIGI Model Animations page.

Table 41-2: Animation speed component control

Component Class	Entity (0).
Instance ID	Entity ID.
Component ID	ID for animation speed, as set on the Entity Components Mappings page.
Component State	Animation ID, as set on the Model Animations mappings page.
Data 1	Speed (float).
Data 2	Not applicable.

Table 41-3: Animation range component control

Component Class	Entity (0).
Instance ID	Entity ID.
Component ID	ID for animation range, as set on the Entity Components Mappings page.
Component State	Animation ID, as set on the Model Animations mappings page.
Data 1	First frame index (unsigned int).
Data 2	Last frame index (unsigned int).

Table 41-4: Animation frame component control

Component Class	Entity (0).
Instance ID	Entity ID.
Component ID	ID for animation frame, as set on the Entity Components Mappings page.
Component State	Animation ID, as set on the Model Animations mappings page.
Data 1	Frame index (unsigned int).
Data 2	Not applicable.

System Component Controls

You can enable and disable shadows using the Component State field of a System Component Control message. You can change shadow quality by passing an integer value in the Data 1 field of the component control. Table 41-5 describes the fields and values for the shadow component control.

Table 41-5: Shadows component control

Component Class	System (13).
Instance ID	Not applicable.
Component ID	Set in the System Components mappings page.
Component State	0 — Disable shadows. 1 — Enable shadows.
Data 1	Shadow Quality. Valid values are 1 through 4, with 1 being the worst quality and 4 the best.
Data 2	Not applicable.

View Component Controls

You can enable or disable the performance statistics overlay and wireframe mode using View Component Control messages. The component ID of either component can be set through the CIGI View Components page in the CIGI Mappings dialog box.

Table 41-6: Performance overlay component control

Component Class	View (1).
Instance ID	Not applicable.
Component ID	Set in the View Components mappings page.
Component State	0 — Disable performance statistics overlay. 1 — Enable overlay.
Data 1	Not applicable.
Data 2	Not applicable.

Table 41-7: Wireframe mode component control

Component Class	View (1).
Instance ID	Not applicable.
Component ID	Set in the View Components mappings.
Component State	0 — Disable wireframe mode. 1 — Enable wireframe mode.
Data 1	Not applicable.
Data 2	Not applicable.

Global Terrain Surface Control

You can change the color setting for a light point group using the Global Terrain Surface Control messages. You can set the component ID used to specify that the message is for light point groupson the CIGI Global Terrain Surface Components page in the CIGI Mappings dialog box. You can map the instance ID for a particular light point group to a light point group name on the CIGI Light Point Groups page in the CIGI Mappings dialog box.

Table 41-8 describes the fields and values for the light point group color component control.

Table 41-8: Light point group color component control

Component Class	Global Terrain Surface Control (8)
Instance ID	Set in the Light Point Groups mappings page
Component ID	Set in the Global Terrain Surface Components page.
Component State	Unused.
Data 1	Floating point value for group's color red component.
Data 2	Floating point value for group's color green component.
Data 3	Floating point value for group's color blue component.
Data 4	Floating point value for group's color alpha component.

41.4. Mapping a Database

Similar to mapping components to IDs, in a simulation using CIGI you can map databases to database numbers. If the host sends a control message that specifies a database, VR-Vantage loads it. If a database is not mapped or the host does not send a control message to load a database, you must load the database in VR-Vantage as you normally would for other connection types.

To map a database:

1. Choose **Settings** → **CIGI Mappings**. The CIGI Mappings dialog box opens (Figure 41-1).
2. Select the CIGI Database Mapping page (Figure 41-5).

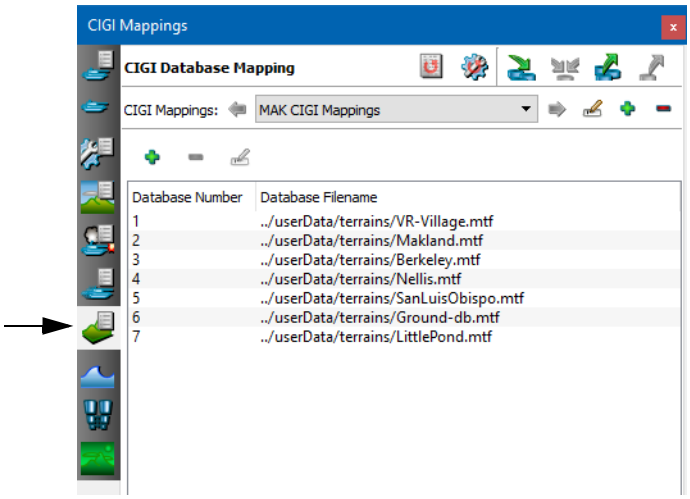


Figure 41-5. CIGI Database Mappings page

3. Click the Add button (+) above the list of databases. A file chooser dialog box opens.
4. Select the database that you want to map to.
5. Click Open. The Create Database Mapping dialog box opens (Figure 41-6).

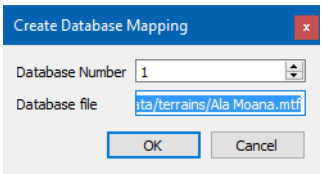


Figure 41-6. Create Database Mapping dialog box

6. Type a number for the database. If you choose a number that is in use, this database replaces the currently mapped database.
7. Click OK.

41.5. Mapping CIGI Views

VR-Vantage associates CIGI views with VR-Vantage channels using channel keywords. (For details about channel keywords, please see "Changing a Channel's Attributes," on page 6-12.) Any channel with keyword cigiView*X* is associated with CIGI view *X*. For example, keyword cigiView2 would map to CIGI view 2.

CIGI view control packets affect the observer that is controlling the specified view. VR-Vantage looks up the channel that has the keyword that matches the view ID specified in the view control packet. Then it finds the observer controlling that channel and applies the view control commands to that observer. If multiple channels have the same keyword, then the first channel found with a matching keyword is used.

Multiple channels with the same keyword are fine, as long as they are all controlled by the same observer. This removes the need for CIGI view groups, although VR-Vantage supports them.

41.5.1. Mapping a CIGI View to an Observer Mode

You can map an observer mode to the ViewType parameter in a CIGI view control message.

To map an observer mode to a CIGI view:

1. Choose **Settings** → **CIGI Mappings**. The CIGI Mappings dialog box opens (Figure 41-1).
2. Select the CIGI Observer Mode Mapping page (Figure 41-7).

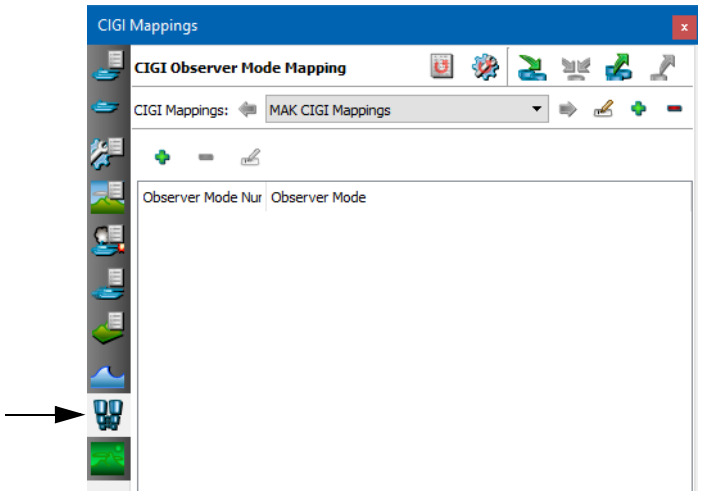


Figure 41-7. CIGI Observer Mode Mapping page

3. Click the Add button (+) above the list of observer mode mappings. The Create New Observer Mode Mapping dialog box opens.
4. Type the CIGI ID or select an ID from the list.

5. Select an observer mode from the Observer Mode list.
6. Click OK.

41.6. Prototypical Host – IG Configurations

This section describes several prototypes for host simulation and image generator configurations. The VR-Vantage CIGI implementation supports the networked IG configuration.

In the classic IG configuration (Figure 41-8), the host simulation connects to the network and filters all the entities and interactions, providing only the relevant ones to the image generator (shown as the VR-Vantage Application) using a protocol like CIGI. The IG renders the scene and presents it to the user on display screens of some sort.

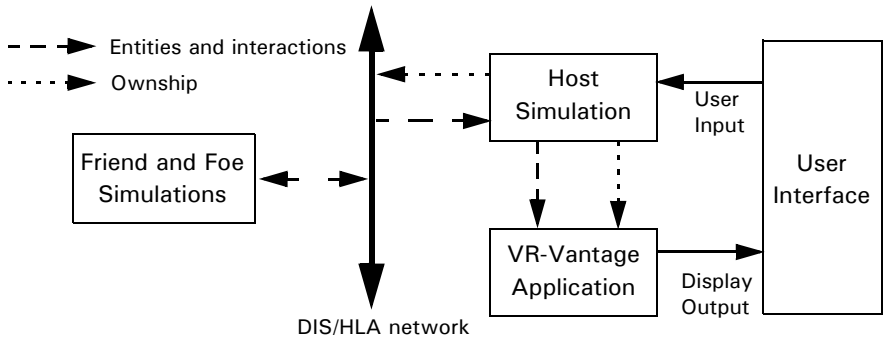


Figure 41-8. Classic IG Configuration

In the networked IG configuration (Figure 41-9), the host simulation communicates only the "ownship" (the one entity the host is simulating) to the image generator, using a protocol like CIGI. The image generator presents all the other entities and interactions in the distributed simulation.

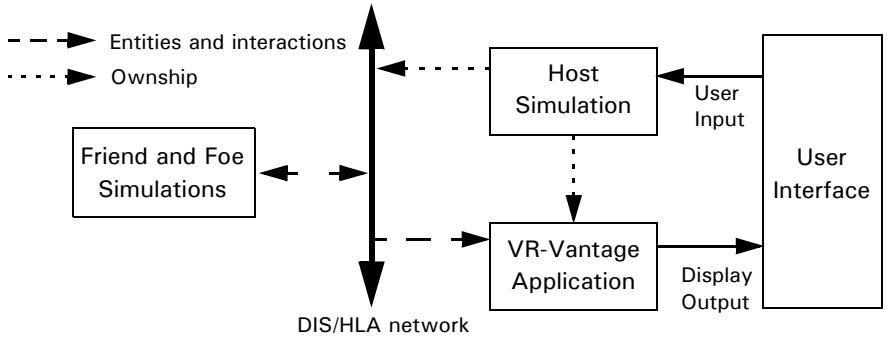


Figure 41-9. Networked IG Configuration

In the distributed visual/simulation configuration (Figure 41-10), the host simulation communicates with the visual application over the DIS/HLA network. (This is the way VR-Vantage connects to any distributed simulation entity.)

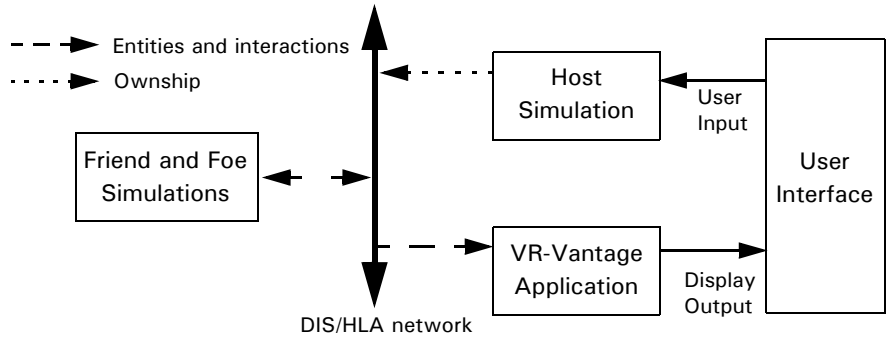


Figure 41-10. Distributed Visual/Simulation Application IG Configuration

In the integrated visual/simulation application (Figure 41-11), the simulation code is embedded in the visual application. This is the way many games are built. You can use the VR-Vantage toolkit to embed simulation code in VR-Vantage or embed VR-Vantage in a simulation application.

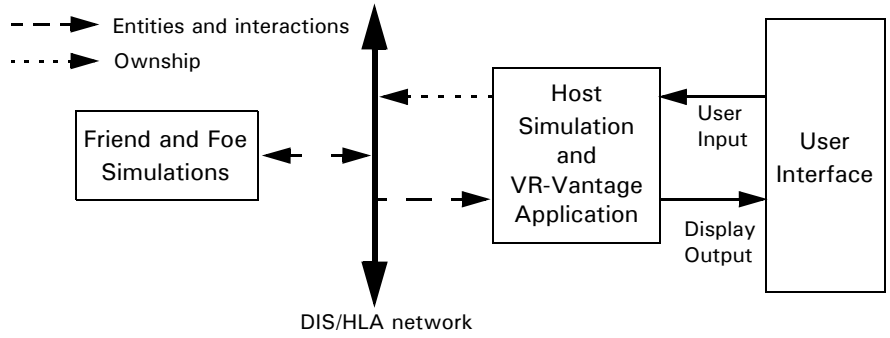


Figure 41-11. Integrated Visual/Simulation Application Configuration



42. CIGI Host Emulator Tutorial

VR-Vantage includes a CIGI host emulator (Windows only). This tutorial explains how to run it and interact with VR-Vantage.

Using the CIGI Host Emulator (Windows Only) 42-2
Start the Host Emulator 42-3
Connect VR-Vantage to the CIGI Host Emulator 42-4
Create an Ownship Entity on the Host Emulator 42-5
Create a Simulated Entity on the Host Emulator 42-6
Change an Entity's Model State 42-6
Add an Effect 42-8
Change the Weather 42-9

42.1. Using the CIGI Host Emulator (Windows Only)

The Windows version of VR-Vantage includes a CIGI host emulator and a set of mappings that you can use to test your CIGI setup. This section provides a basic introduction to starting the host emulator, connecting VR-Vantage to it, and sending basic control messages to VR-Vantage.

Figure 42-1 illustrates connection settings that must be the same on the host emulator ("Start the Host Emulator," on page 42-3) and the CIGI connection in VR-Vantage ("Connect VR-Vantage to the CIGI Host Emulator," on page 42-4).

- For best results, shut down VR-Vantage before you start this tutorial. Start it when instructed to do so.
- To see a set of brief videos that illustrate this tutorial, on the Start menu, choose **VR-Vantage 2.3 → Documentation → VR-Vantage Video Tutorials**, open *./doc/vrvantage_tutorialvideos.htm*, or click the camera icon next to the procedure that you want to view.

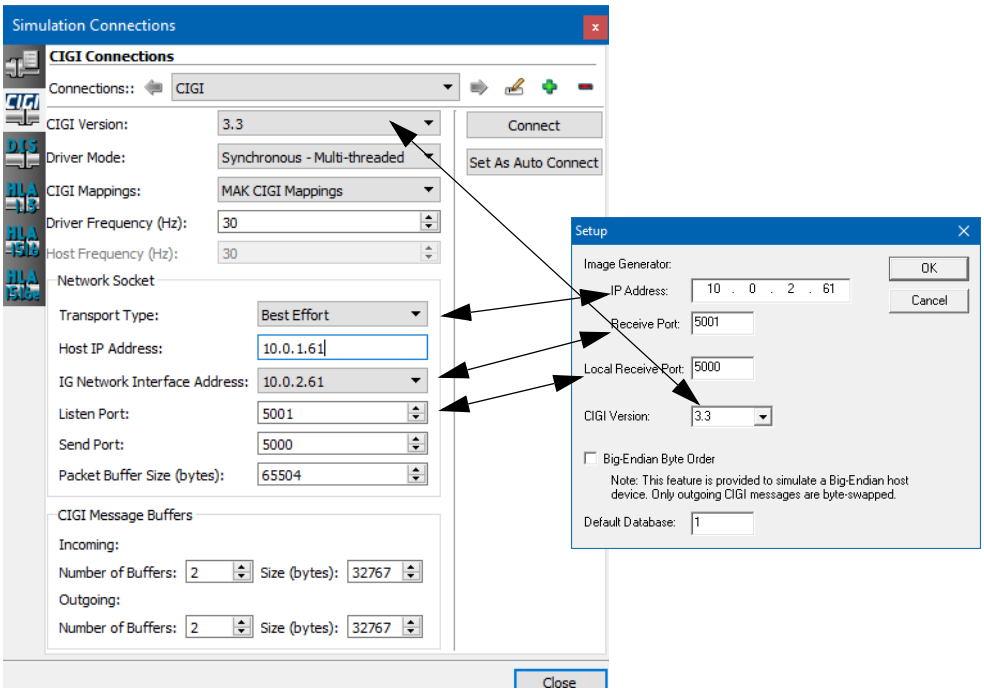


Figure 42-1. Connection settings and host emulator setup

42.1.1. Start the Host Emulator

Running the host emulator on the same computer as VR-Vantage could affect VR-Vantage's performance.

To start the host emulator:



1. On the Start menu, choose **VR-Vantage 2.3** → **Tools** → **CIGI Host Emulator**. The host emulator starts up (Figure 42-2). The first time you run it, the Setup dialog box opens automatically (Figure 42-3). (Thereafter, to change the setup, in the host emulator, choose **File** → **Setup**.)

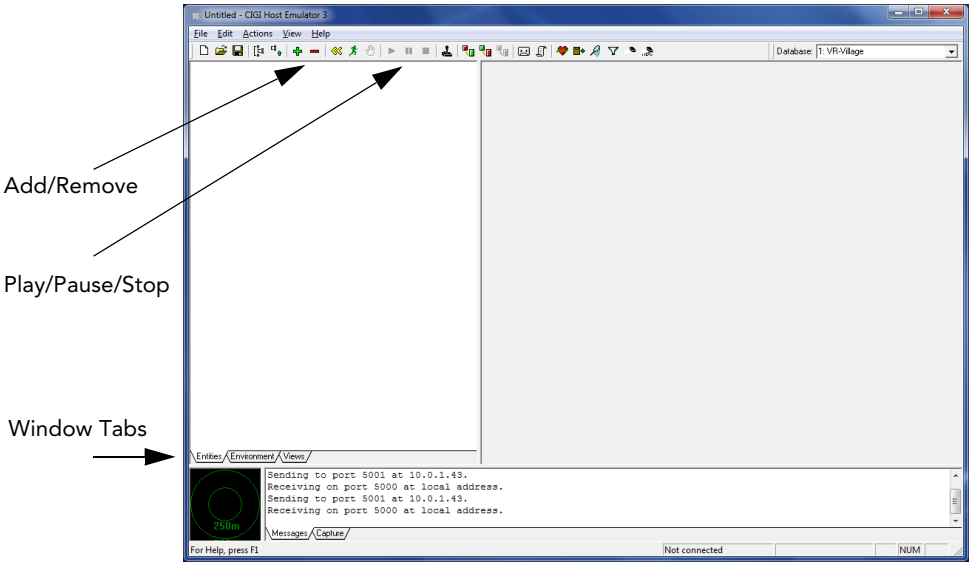


Figure 42-2. CIGI host emulator

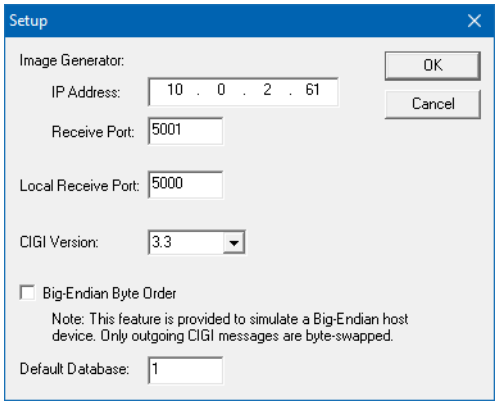


Figure 42-3. Setup dialog box

2. In the IP Address box, type the IP address for the computer on which you are running VR-Vantage.

3. In the Receive Port box, type the port that is configured as the Listen Port in the CIGI Connection that you are using in VR-Vantage. (In other words, this is the port to which the host emulator should send its messages.)
4. In the Local Receive Port, type the port on which the host emulator should receive messages. This must match the Send Port value on the CIGI Connection that you are using in VR-Vantage.
5. Specify the CIGI version that you want to use.
6. Specify the default database that you want to use. This value should match a database mapping in the CIGI Database Mapping page of the CIGI Mapping Settings in VR-Vantage. (Database 1 is VR-Village.)
7. Click OK.

42.1.2. Connect VR-Vantage to the CIGI Host Emulator

To connect VR-Vantage to the CIGI Host Emulator:

1. Start VR-Vantage IG.
2. Choose **Settings** → **Connections**. The Simulation Connections dialog box opens.
3. Select the CIGI Connections page (Figure 42-1).
4. Select a connection or create a new one.
5. Set the CIGI Version to the same CIGI version you specified in the host emulator setup.
6. Set CIGI Mappings to MAK CIGI Mappings.
7. Set the Host IP Address to the IP address of the computer on which the host emulator is running.
8. Set the Network Device Address to the IP address of the computer running VR-Vantage IG. The Image Generator IP Address in the host emulator setup must match this value.
9. Set the Listen Port to the same value as the Image Generator Receive Port value in the host emulator setup.
10. Set the Send Port to the same value as the Local Receive Port value in the host emulator setup.
11. Change any other connection parameters as desired.
12. Click Connect. After a few seconds, VR-Vantage displays the database configured in the host emulator.

42.1.3. Create an Ownship Entity on the Host Emulator

The ownship entity controls the observer in VR-Vantage.

To create the ownship entity:

1. In the CIGI host emulator, select the Entities tab (Figure 42-2). The window is empty because there are no entities.
2. On the toolbar (Figure 42-2), click the Add button (). The New Entity dialog box opens (Figure 42-4).

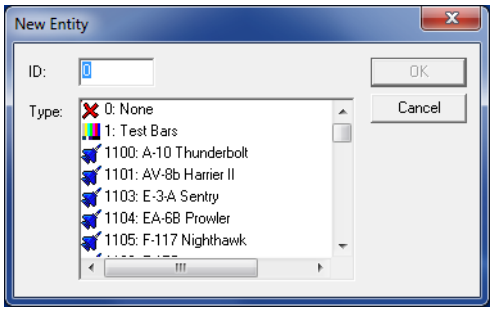


Figure 42-4. New Entity dialog box

3. Select 0: None.
4. Click OK. The Ownship entity is added to the entity list. Its properties are listed in the panel to the right of the entity list (Figure 42-5).

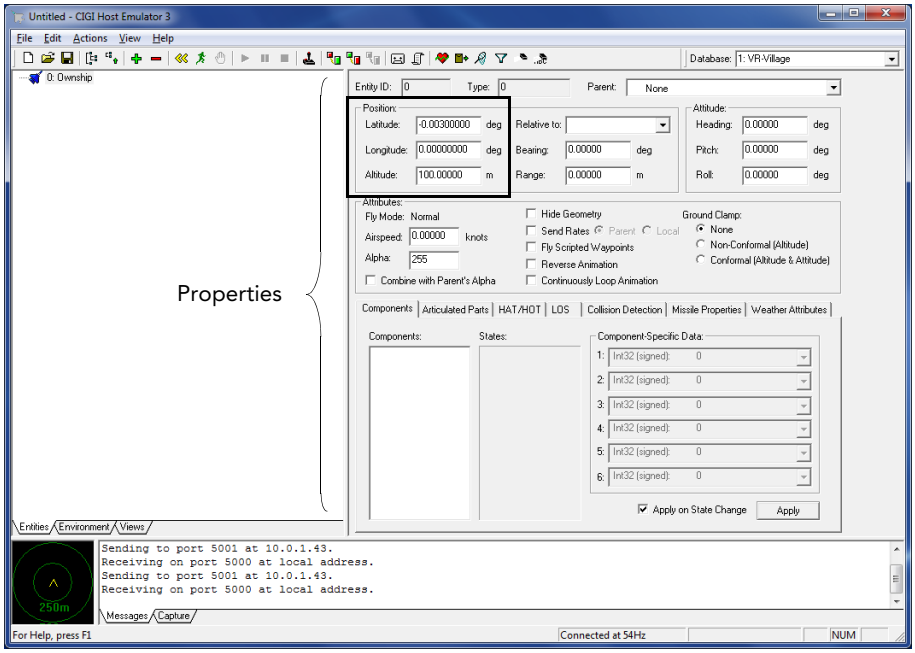


Figure 42-5. Entity properties for ownship entity

5. In VR-Vantage, an entity called CIGI Ownship is added to the entity list.

6. Attach to the entity by sending a view control message from the host emulator:
   a. Choose **Actions** → **Send Packet** → **View Control**. The Send View Control Packet dialog box opens.
   b. Edit the view control attributes as desired.
   c. Click Send. The CIGI host emulator sends a view control with the Position data in the main window. (If you are using database 1 (VR-Village), the view changes to a view of the village.)
7. In the Position group box on the host emulator, change the Altitude attribute (Figure 42-5). The view in VR-Vantage changes.

42.1.4. Create a Simulated Entity on the Host Emulator

Creating a simulated entity will let you apply effects and experiment with setting model states and other attributes.

To create a simulated entity:



1. In the CIGI host emulator, select the Entities tab (Figure 42-2).
2. On the toolbar (Figure 42-2), click the Add button (✚). The New Entity dialog box opens (Figure 42-4).
3. Select M1A2.
4. Click OK. The entity is added to the entity list.
5. In VR-Vantage, an entity called CIGI Entity 1 is added to the entity list.

42.1.5. Change an Entity's Model State

To change an entity's model state:

1. In VR-Vantage, attach to the M1A2.
2. Rotate the view so that you can see the rear of the tank.
3. In the host emulator, select the Entities tab.
4. Select the M1A2 entity (Figure 42-6).

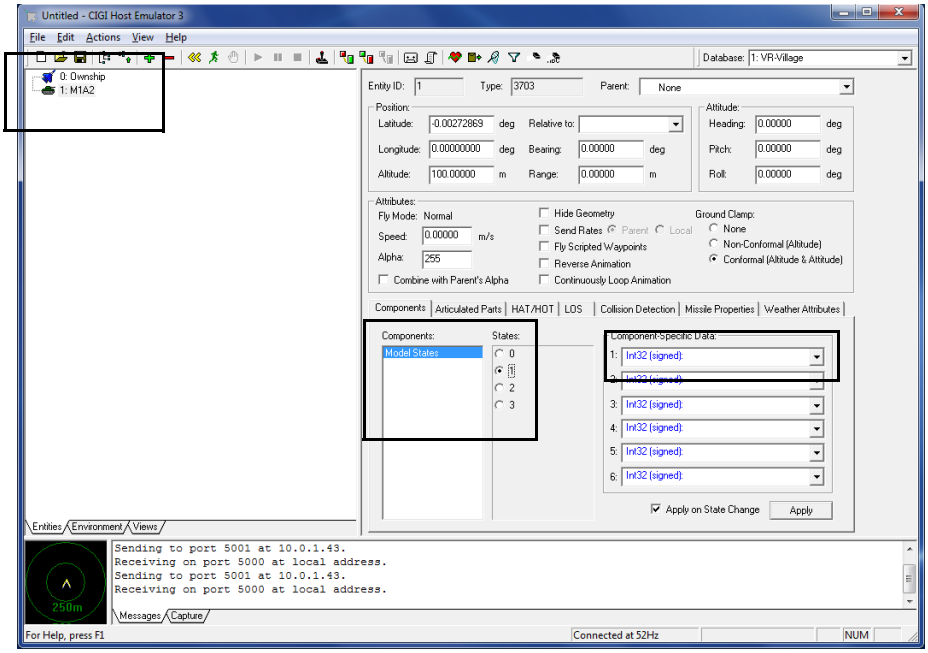


Figure 42-6. Entity attributes

5. In the properties panel, select the Components tab (Figure 42-6).
6. Select Model States.
7. In the States list, select option 1.
8. In the Component Specific Data list, put the cursor in the first box.
9. Type 16.
10. Click Apply. The brake lights on the tank turn on (Figure 42-7).



Figure 42-7. Brake lights

42.1.6. Add an Effect

To add an effect in the MAK implementation of CIGI, you add the effect and attach it to an entity.

To add an effect:



1. In the CIGI host emulator, select the Entities tab (Figure 42-2).
2. On the toolbar (Figure 42-2), click the Add button (). The New Entity dialog box opens (Figure 42-4).
3. Select Detonation.
4. Click OK. The effect is added to the entity list.
5. On the properties panel, in the Parent list, select M1A2. In the entity list, the Detonation is now subordinate to the M1A2 entity (Figure 42-6).

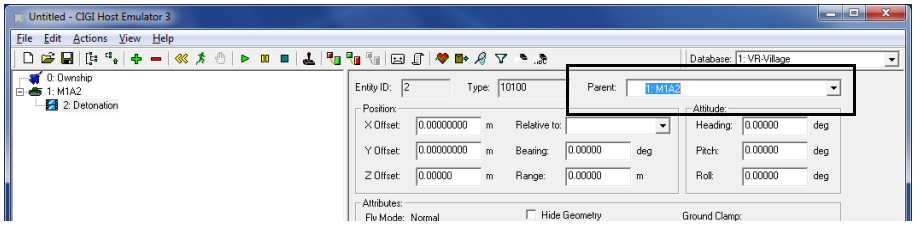


Figure 42-8. Entity attributes

6. To display the detonation, click Play (Figure 42-2).

42.1.7. Change the Weather

You can send control messages to enable and disable precipitation. VR-Vantage supports layer 4 (rain) and 5 (snow).

To change the weather:



1. In the host emulator, choose **Actions** → **Send Packet** → **Weather Control**. The Send Weather Control Packet dialog box opens (Figure 42-9).

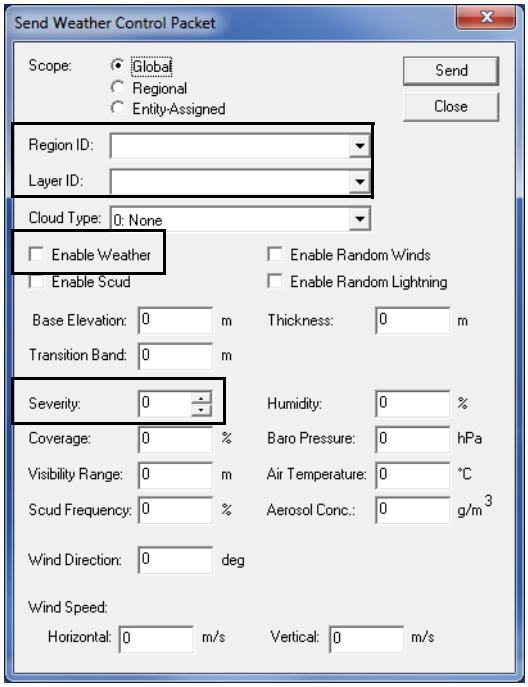


Figure 42-9. Send Weather Control Packet dialog box

2. In the Region ID box, type 0.
3. In the Layer ID box, type 4 (for rain) or 5 (for snow).
4. Select the Enable Weather check box.
5. In the Severity box, select an option greater than zero.
6. Click Send. VR-Vantage displays the selected form of precipitation.



43. CIGI Utilities

This chapter describes the utilities provided for testing CIGI connections.

Using asynchHostTest.py 43-2
Using logger.py 43-3
Using snooper.py 43-3
The pyCigi Module 43-5

43.1. Using asynchHostTest.py

The *./tools/CigiUtilities/asynchHostTest.py* script has two functions. It acts as a simple CIGI host that sends test packets to a socket for test purposes. It also serves as an example of how you can write a script to test your CIGI network.

This script opens a socket and repeatedly sends IGControl and EntityControl messages to the socket until you press **`Ctrl+c`**. The script instructs the IG to load the Ground-DB terrain database and 700 FixedWingA-10Thunderbolt entities from VR-Vantage. It then cycles the roll of the entities at a given frame rate.

To run *asyncHostTest.py* with default values (port 5001, IP 127:0:0:1), use:

```
asyncHostTest.py
```

The script creates 700 entities and changes their roll value. To see the entities in VR-Vantage:

1. Start VR-Vantage.
2. Choose **Settings** → **Connection**. The Simulation Connections dialog box opens.
3. Select the CIGI Connections page. The Listen Port (5001) and Host IP Address match the default values for the host created by the script.
4. Change the IG Network Interface Address to 127.0.0.1.
5. Click Connect. VR-Vantage loads the ground-db terrain and displays the entities (Figure 43-1).



Figure 43-1. Entities generated by *asyncHostTest.py*

43.2. Using logger.py

The *./tools/CigiUtilities/logger.py* script listens to a UDP socket (unicast or multicast) and records the packets it receives. It can then play the packets back.

To record:

```
logger.py -r
```

The default log file is *cigiSession.log*.

To play:

```
logger.py
```

The logger records data in a binary format. However, you can convert it to a readable format. To convert the data, run:

```
logger.py -c
```

The default output file is *cigiSession.txt*. The following is an example of converted log entries:

```
<Time(0.000000)>
<IGControl({'lastFrame': 0, 'timeStamp': 0, 'dbNum': 6, 'frameNum': 1})>
<EntityControl({'bits2': 0, 'alt': 140.0, 'parentID': 0, 'lon': -122.988,
  'pitch': 0.0, 'entType': 1100, 'roll': 1.0, 'alpha': 255, 'yaw': 0.0,
  'lat': 36.00827037, 'entID': 1, 'bits1': 1})>
<EntityControl({'bits2': 0, 'alt': 140.0, 'parentID': 0, 'lon': -
  122.9883, 'pitch': 0.0, 'entType': 1100, 'roll': 1.0, 'alpha': 255,
  'yaw': 0.0, 'lat': 36.00827037, 'entID': 2, 'bits1': 1})>
<EntityControl({'bits2': 0, 'alt': 140.0, 'parentID': 0, 'lon': -
  122.98859999999999, 'pitch': 0.0, 'entType': 1100, 'roll': 1.0,
  'alpha': 255, 'yaw': 0.0, 'lat': 36.00827037, 'entID': 3, 'bits1': 1})>
<EntityControl({'bits2': 0, 'alt': 140.0, 'parentID': 0, 'lon': -
  122.98889999999999, 'pitch': 0.0, 'entType': 1100, 'roll': 1.0,
  'alpha': 255, 'yaw': 0.0, 'lat': 36.00827037, 'entID': 4, 'bits1': 1})>
<EntityControl({'bits2': 0, 'alt': 140.0, 'parentID': 0, 'lon': -
  122.98919999999998, 'pitch': 0.0, 'entType': 1100, 'roll': 1.0,
  'alpha': 255, 'yaw': 0.0, 'lat': 36.00827037, 'entID': 5, 'bits1': 1})>
```



When the logger records packets, it does not compress them. The log file can quickly grow quite large.

43.3. Using snooper.py

The *./tools/CigiUtilities/snooper.py* script can connect to the host and the IG at the same time and can record traffic and then play it back. Figure 43-2 illustrates the snooper as follows:

- Host/IG interaction without the snooper.
- Host/IG interaction with the snooper mediating. The snooper receives the packets from the host and IG and passes them on. It also records them to a file.
- Snooper playback. The snooper sends the logged packets to the proper destination.

When the snooper records packets, it does not compress them. The log file can quickly grow quite large.

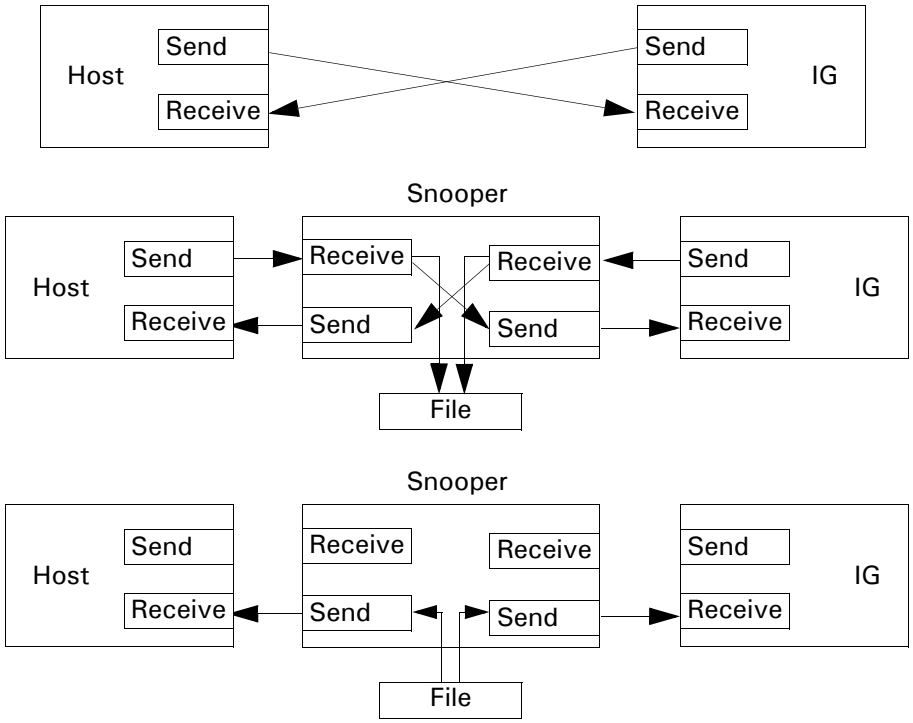


Figure 43-2. Using the snooper

To run the snooper you must specify send and receive ports for both the host and IG, for example:

```
snooper.py --host_send_port 5000 --host_recv_port 5001
  --ig_recv_port 5002 --ig_send_port 5003
```

Since this is a long command line, you can save the command line to a configuration file, as follows:

```
snooper.py --host_send_port 5000 --host_recv_port 5001
  --ig_recv_port 5002 --ig_send_port 5003 --save [-f
  myfile]
```

If you do not specify a file with `-f`, the command is saved to *snooper.cfg*.

To load the saved configuration, run:

```
snooper.py --load [-f myfile]
```

43.4. The pyCigi Module

The pyCigi module (*./tools/CigiUtilities/byCigi*) is a package of modules to handle the reading, writing, packing and unpacking of CIGI messages. pyCigi has modules to handle each of the CIGI message types that are supported by VR-Vantage. For a list of supported packets, please see Chapter 44, *CIGI Packets Supported*.

Each module within pyCigi is self documented and contains a unit test. The documentation for a module can be seen from inside a Python shell:

```
python
>>> import pyCigiStartOfFrame
>>> help(pyCigiStartOfFrame.StartOfFrame)
```

To run the unit test for a module, run the module as a script from a shell window:

```
pyCigiStartOfFrame.py
```

The script *pyCigiMappings.py* gets information about mappings from *./appData/settings/stealth/MAK CIGI Mappings.mcsx*. Do not move this file or the script will not work.

For complete documentation of the pyCigi module, please see *Readme.txt*.



44. CIGI Packets Supported

This Simulation Object Type MappingsSimulation Object Type Mappings lists the various CIGI packets and the fields that VR-Vantage supports.

CIGI Packet Support 44-2
Support for CIGI Weather Control Packet 44-9
User Defined CIGI Packet 44-9

44.1. CIGI Packet Support

The tables in this section describe VR-Vantage support for the following CIGI packet types:

- IG Control (Table 44-1).
- Entity Control (Table 44-2).
- Component Control (Table 44-3).
- Short Component Control (Table 44-4).
- Articulated Part Control (Table 44-5).
- Short Articulated Part Control (Table 44-6).
- Rate Control (Table 44-7).
- Celestial Sphere Control (Table 44-8).
- Atmosphere Control (Table 44-9).
- Weather Control (Table 44-10).
- View Control (Table 44-11).
- View Definition (Table 44-12).
- HAT/HOT Request (Table 44-13).
- Line-of-sight Segment Request (Table 44-14).
- Line-of-sight Vector Request (Table 44-15).

Table 44-1: IG control packets

Field	Supported	Unsupported
Database Number	X	
IG Mode	X	
Timestamp Valid	X	
Extrapolation/Interpolation Enable	X	
Host Frame Number	X	
Timestamp	X	
Last IG Frame Number	X	

Table 44-2: Entity control packets

Field	Supported	Unsupported
Entity ID	X	
Entity State	X	
Attach State	X	
Collision Detection Enable		X
Inherit Alpha	X	
Ground/Ocean Clamp	X (Conformal Only)	
Animation Direction	X	
Animation Loop Mode	X	
Animation State	X	
Linear Extrapolation/Interpolation Enable	X	
Alpha	X	
Entity Type	X	
Parent ID	X	
Roll/Pitch/Yaw	X	
Latitude/Longitude/Altitude or X/Y/Z offset	X	

Table 44-3: Component control packets

Field	Supported	Unsupported
Component ID	X	
Instance ID	X	
Component Class	X	
Component State	X	
Component Data	X	

Table 44-4: Short component control packets

Field	Supported	Unsupported
Component ID	X	
Instance ID	X	
Component Class	X	
Component State	X	
Component Data	X	

Table 44-5: Articulated part control packets

Field	Supported	Unsupported
Entity ID	X	
Articulated Part ID	X	
Articulated Part Enable	X	
X/Y/Z Offset Enable	X	
Roll/Pitch/Yaw Enable	X	
X/Y/Z Offset	X	
Roll/Pitch/Yaw	X	

Table 44-6: Short articulated part control packets

Field	Supported	Unsupported
Entity ID	X	
Articulated Part ID 1	X	
Articulated Part ID 2	X	
DOF Select 1	X	
DOF Select 2	X	
Articulated Part Enable 1	X	
Articulated Part Enable 2	X	
DOF 1	X	
DOF 2	X	

Table 44-7: Rate control packets

Field	Supported	Unsupported
Entity ID	X	
Articulated Part ID	X	
Apply To Articulated Part	X	
Coordinate System	X	
X/Y/Z Linear Rate	X	
Roll/Pitch/Yaw Angular Rate	X	

Table 44-8: Celestial sphere control packets

Field	Supported	Unsupported
Hour	X	
Minute	X	
Ephemeris Model Enable	X	
Sun Enable		X
Moon Enable		X
Star Field Enable		X
Date/Time Valid	X	
Date	X	
Star Field Intensity		X

Table 44-9: Atmosphere control packets

Field	Supported	Unsupported
Global Air Temperature		
Global Visibility Range	X	
Global Horizontal Wind Speed	X	
Global Wind Direction	X	
Global Barometric Pressure		X

Table 44-10: Weather control packets

Field	Supported	Unsupported
Entity/Region ID		X
Layer ID	X (Only Rain/Snow)	
Humidity		X
Weather Enable	X	
Random Winds Enable		X
Random Lightning Enable		X
Cloud Type		X
Scope		X
Severity	X	
Air Temperature		X
Visibility Range*	X	
Scud Frequency		X
Coverage		X
Base Elevation		X
Thickness		X
Transition Band		X
Horizontal Wind Speed*	X	
Wind Direction*	X	
Barometric Pressure		X
Aerosol Concentration		X
For more information, please see "Support for CIGI Weather Control Packet," on page 44-9		

Table 44-11: View control packets

Field	Supported	Unsupported
View ID	X	
Group ID	X	
X/Y/Z Offset Enable	X	
Roll/Pitch/Yaw Offset Enable	X	
Entity ID	X	
X/Y/Z Offset	X	
Roll/Pitch/Yaw	X	

Table 44-12: View definition packets

Field	Supported	Unsupported
View ID	X	
Group ID	X	
Near/Far/Left/Right/Top/Bottom Enable	X	
Mirror Mode		X
Pixel Replication Mode		X
Projection Type		X
Reorder		X
View Type		X
Near/Far/Left/Right/Top/Bottom	X	

Table 44-13: HAT/HOT request packets

Field	Supported	Unsupported
HAT/HOT ID	X	
Request Type	X (Extended not supported)	
Coordinate System	X	
Update Period	X	
Entity ID	X	
Latitude/Longitude/Altitude or X/Y/Z offset	X	

Table 44-14: Line-of-sight segment request packets

Field	Supported	Unsupported
LOS ID	X	
Request Type		X (Extended not supported)
Source Point Coordinate System	X	
Destination Point Coordinate System	X	
Response Coordinate System		X
Destination Entity ID Valid	X	
Alpha Threshold		X
Source Entity ID	X	
Source Latitude/Longitude/Altitude or X/Y/Z Offset	X	
Destination Latitude/Longitude/Altitude or X/Y/Z Offset	X	
Material Mask		X
Update Period	X	
Destination Entity ID	X	

Table 44-15: Line-of-sight vector request packets

Field	Supported	Unsupported
LOS ID	X	
Request Type		X (Extended Not Supported)
Source Point Coordinate System	X	
Response Coordinate System		X
Alpha Threshold		X
Entity ID	X	
Azimuth	X	
Elevation	X	
Minimum Range	X	
Maximum Range	X	
Source Latitude/Longitude/Altitude or X/Y/Z Offset	X	
Material Mask		X
Update Period	X	

44.1.1. Support for CIGI Weather Control Packet

VR-Vantage supports the CigiWeatherCtrlV3 packet's wind direction, horizontal wind direction, and visibility range fields (WindDir, HorizWindSp, and VisibilityRng respectively).

Prior to VR-Vantage 2.1, this packet was used just for precipitation. To provide backward compatibility, the RegionID field (which we normally do not support) is used as a flag set to indicate which weather control items are supported and which are ignored.

The flags are

```
enum WeatherControlState
{
  WindDirSet = 0           // bit 0
  , WindSpeedSet = 1         // bit 1
  , VisibilitySet = 2        // bit 2
  , PrecipitationIgnore = 3  // bit 3
};
```

As before, WeatherEn enables or disables precipitation, with LayerID indicating precipitation type (4 == rain, 5 == snow). Therefore, existing CIGI Weather Control packets used with VR-Vantage work as they did before.

If you want to specify wind direction, horizontal wind speed, visibility, or some combination of them, the appropriate flags must be set into the RegionID field. Precipitation values can be ignored by setting bit 3 of the RegionID field.

44.2. User Defined CIGI Packet

VR-Vantage supports a user-defined packet for CIGI. For details, please see *DtCigiDiGuyCtrl.h*.



XI. Configuring the Graphical User Interface

The chapters in this section describe the graphical user interface (GUI) and how to configure it.



45. The VR-Vantage User Interface

The VR-Vantage graphical user interface has features typical for modern windowing systems and follows standard conventions for using menus and selecting objects. This chapter describes some features that might not be immediately obvious.

Dockable Panels and Toolbars 45-2
- Docking and Undocking a Panel 45-2
- Docking and Undocking a Toolbar 45-3
- Displaying and Hiding Panels and Toolbars 45-3
- Restoring the Layout of Toolbars and Panels 45-4

Viewing the Window in Full Screen Mode 45-4

Using Context-Sensitive Menus 45-5

Common VR-Vantage Buttons 45-5

Specifying Display Units 45-6

45.1. Dockable Panels and Toolbars

VR-Vantage has floating, dockable panels for many of its important functions (Figure 45-1). It also has toolbars that let you quickly access the commands available on the menus.

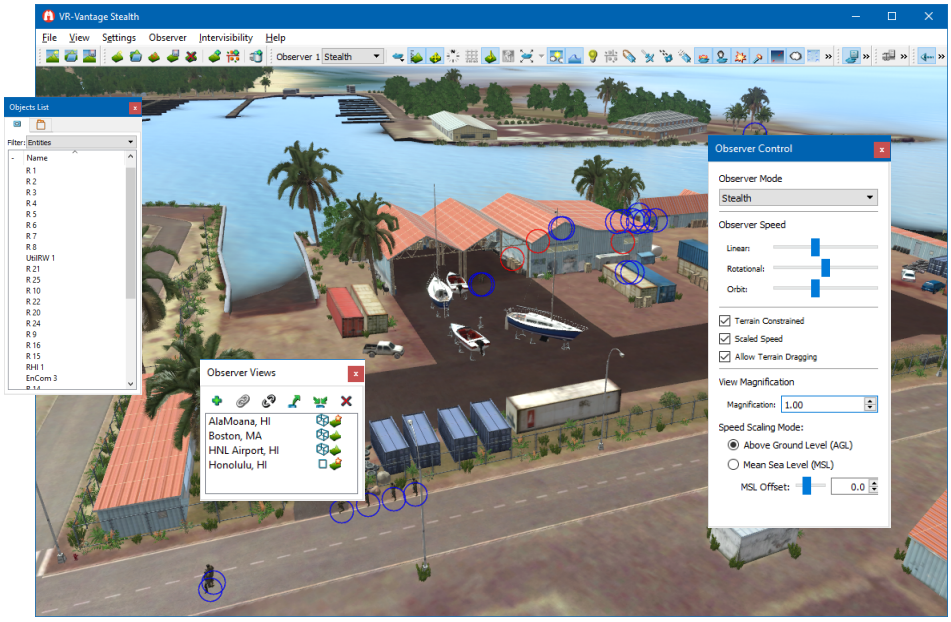


Figure 45-1. VR-Vantage window and undocked panels

45.1.1. Docking and Undocking a Panel

To undock a panel:

1. Place the cursor over the panel's title.
2. Press and hold the left mouse button.
3. Drag the panel away from the window.

➤ To dock a floating panel, double-click its title bar.

45.1.2. Docking and Undocking a Toolbar

To undock a toolbar:

1. Place the cursor over the handle.



Figure 45-2. Toolbar handle

2. Press and hold the left mouse button.
3. Drag the toolbar away from the window.

➤ To dock a toolbar, drag it back to the toolbar strip. The other toolbars will move to create a space to drop in the toolbar (Figure 45-3).

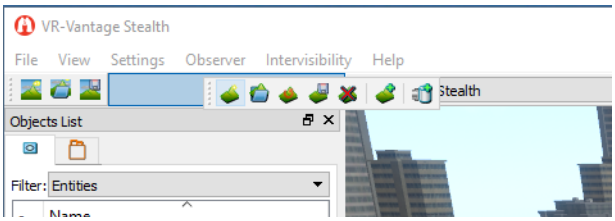


Figure 45-3. Docking a toolbar

45.1.3. Displaying and Hiding Panels and Toolbars

You can hide any of the panels or toolbars.

➤ To hide or view a panel or toolbar, select it on the View menu.

45.1.4. Restoring the Layout of Toolbars and Panels

VR-Vantage saves the layout of toolbars and panels when you shut down. (It does not save the size and location of the main window.) You can restore the factory layout or the last saved layout.

To restore the layout of toolbars and panels:

1. Choose **Settings** → **Application**. The Application Settings dialog box opens.
2. Select the GUI Layout page (Figure 45-4).

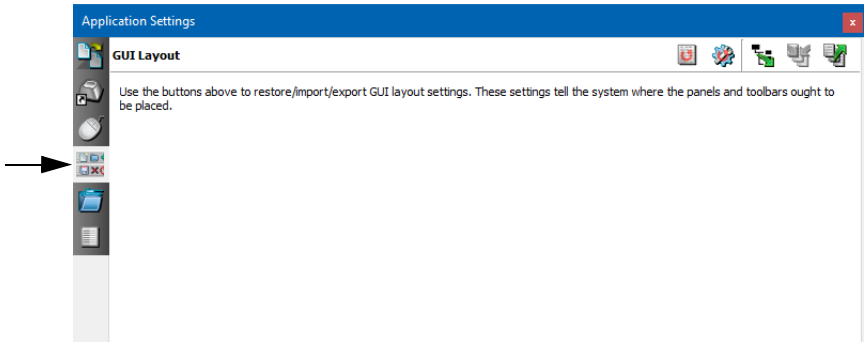


Figure 45-4. GUI Layout page

3. Click the settings button for the operation you want to perform. (For a summary of common settings buttons, please see "Managing VR-Vantage Settings," on page 3-15.)

45.2. Viewing the Window in Full Screen Mode

You can display VR-Vantage so that it covers the entire screen and does not show the title bar or border. This mode is intended for use in demonstrations. When menus and panels are hidden, you can still use all keyboard controls, and you can still click simulation objects in the window to attach the observer to them.

You can configure VR-Vantage to start in full screen mode with the `--fullScreen` command-line option. If you are viewing a simulation that takes place primarily over the ocean, you may get better performance using the `--pseudoFullScreen` command-line option.

➤ To view VR-Vantage in full screen mode, choose **View** → **Show Full Screen**, or press **`Ctrl+Enter`**.

➤ To exit full screen mode, press **`Ctrl+Enter`**.

45.3. Using Context-Sensitive Menus

VR-Vantage has context sensitive menus (also called popup menus or right-click menus) that you can display by clicking the right mouse button.

If you right-click a simulation object, tactical graphic, or other scene object, the menu displays the actions that you can take.

If you right-click the menu bar, you can select the commands available on the View menu.

45.4. Common VR-Vantage Buttons

VR-Vantage dialog boxes have a common set of buttons, as follows:

-  Add an item (definition, model, and so on).
- Remove the selected item.
- Copy the selected item and add the copied version to the list.
- Select a filter of some type.
- Rename the selected item.
- Add from a list.
- Move up in the list.
- Move down in the list.
- Select the previous item.
- Select the next item.
- Add an attribute.
- Remove local attribute definition and use the parent's definition.
- Search.

45.5. Specifying Display Units

You can specify the units used to display coordinate system, distance, altitude, and other measurements. Display units are configured by measurement type. This means that you can use different measurement units for different types of data. You can also specify the number of decimal places for measurement values. When you change a measurement display unit, the changes take effect immediately. They are saved and applied the next time you run VR-Vantage.



The following coordinate system and unit display information is independent of (and therefore not affected by) the settings described in this section:

- The coordinate system information for the observer location in the Observer Information Panel.
- Grid line coordinates, units, and precision.

The types of data and the display objects that they affect are as follows:

- Coordinate system
- Distance
- Altitude
- Velocity (speed)
- Acceleration
- Rotational velocity
- Orientation (heading).

Table 45-1 describes the coordinate systems that you can specify.

Table 45-1: Coordinate systems supported

Coordinate system	Description
Latitude/Longitude (degrees:minutes:seconds)	Location is displayed as two position fields and one altitude field. Displays coordinates in the latitude and longitude using the geodetic WGS84 coordinate system. Each angle is displayed in degrees:minutes:seconds with seconds displaying base 10 fractional seconds.
Latitude/Longitude (decimal radians)	Location is displayed as two position fields and one altitude field. Position coordinates in the latitude and longitude using the geodetic WGS84 coordinate system. Each angle is displayed in decimal radians.
Geocentric	Location is displayed as three position fields. Position coordinates are displayed in the geocentric coordinate system. The position fields are always in meters.

Table 45-1: Coordinate systems supported

Coordinate system	Description
UTM	Location is displayed as two position fields and one altitude field. Position coordinates are displayed in the Universal Transverse Mercator system. The first position field displays the zone and the x location in meters. The second position field displays the y position in meters.
MGRS	Location is displayed as two position fields and one altitude field. Position coordinates are displayed in the Military Grid Reference System. The first position field displays the zone. The second position field displays the grid location. The precision controls the number of digits used in the grid display.
Database	Location is displayed as three position fields. Location is displayed using VR-Vantage's current internal cartesian database system. The position fields are displayed using the current distance units.

To specify measurement units:

1. Choose **Settings** → **Display**. The Display Settings dialog box opens.
2. Select the Display Units Settings page (Figure 45-5).

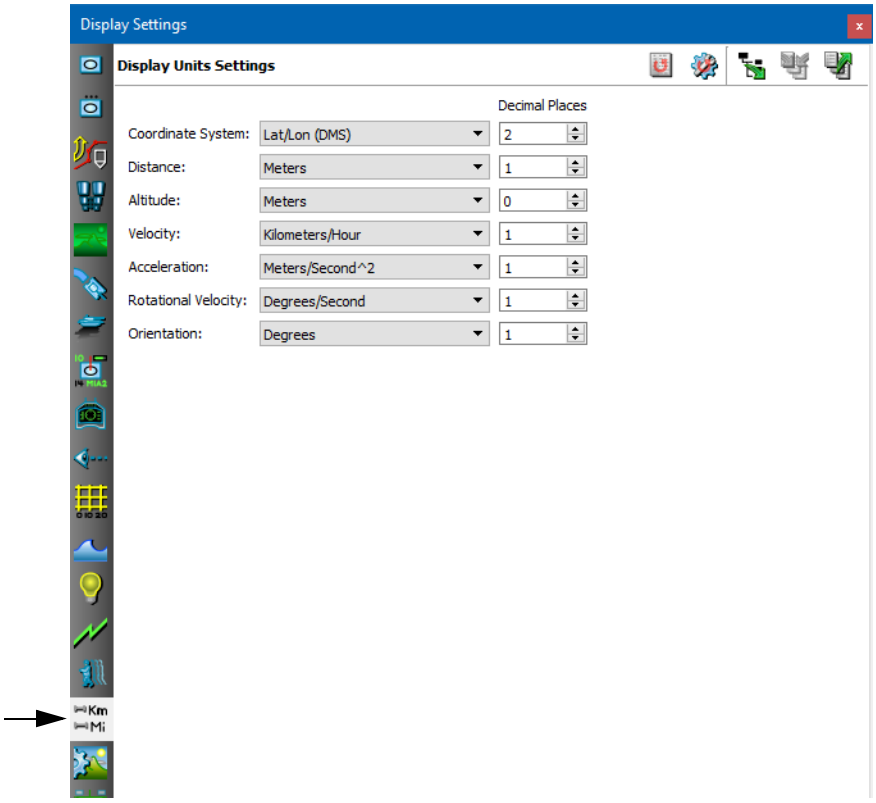


Figure 45-5. Display Units Settings page

3. Select the measurement unit you want for each option from its list.
4. Type a precision for each measurement unit in the corresponding box in the Decimal Places column.



46. Creating and Editing Key Mappings

This chapter explains how to edit key mappings and create new key mappings.

Introduction 46-2
The Key Mapping Editor 46-2
Binary Key Mappings 46-3
Editing a Key Map 46-3
Adding a Key Mapping 46-3
Changing a Key Mapping 46-5
Deleting a Key Mapping 46-5
Using Combined Key Mappings 46-6
Filtering the Function List 46-7
Creating a Key Map 46-7
Deleting a Key Map 46-7

46.1. Introduction

When you move the observer, you are actually executing functions that control the observer's movement. For example, the Move Observer Forward function moves the observer forward in observer coordinates. The Move Level Observer Forward function moves the observer forward in level observer coordinates.

The various movement functions are mapped to the keys on the keyboard and to mouse buttons and the mouse wheel.

VR-Vantage comes with key mappings that we think will meet the needs of most VR-Vantage users. However, you can edit the default keyboard mappings and create new mappings of your own with the Key Mapping Editor.

46.2. The Key Mapping Editor

The Key Mapping Editor (Figure 46-1) lists all of the functions that you can control from the keyboard and the keys to which they are mapped. Some functions are mapped to more than one key or key combination. A key can be mapped to more than one function (combined mappings). You can:

- Edit a function's key mappings.
- Add new mappings.
- Delete mappings.

Changes are saved automatically.

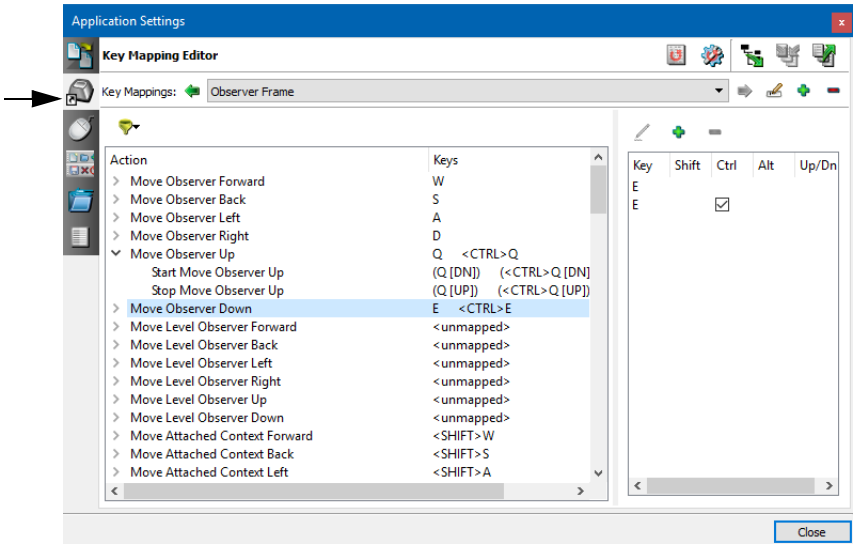


Figure 46-1. Key Mapping Editor

46.2.1. Binary Key Mappings

Many navigation functions require you to press and hold a key. For example, you press and hold **w** to move the observer forward. (By contrast some keyboard actions are toggles or unary functions, such as pressing period to attach to the next simulation object.) For these actions, VR-Vantage needs to keep track of when the key is pressed down (start the action) and when it is released (stop the action). To a human user, a function like Move Observer Forward is experienced as one action. But the computer experiences it as two actions. Therefore, key mappings must take this into account. The Key Mapping Editor shows the key mappings for these functions as paired functions that are subordinate (children) to the primary function (the parent), as shown in the Move Observer Up entry in Figure 46-1.

46.3. Editing a Key Map

The Key Mapping Editor lists all of the VR-Vantage functions that you can control from the keyboard.

To edit a key map:

1. Choose **Settings** → **Application**. The Application Settings dialog box opens.
2. Select the Key Mapping Editor page (Figure 46-1).
3. Select the function whose key mappings you want to edit.
4. Edit the key mappings as described in:
   - "Adding a Key Mapping," on page 46-3.
   - "Changing a Key Mapping," on page 46-5.
   - "Deleting a Key Mapping," on page 46-5.
5. Optionally, export the key map to a file.
6. Click OK.

46.3.1. Adding a Key Mapping

If a function is not mapped to a key, it is marked `<unmapped>`. You can add a mapping to unmapped functions and you can add additional key mappings to functions that are already mapped. If you add a key mapping to a binary function, the Key Mapping Editor automatically applies it to the child (UP and DN) functions. You can add key mappings to the child functions of binary mappings. When you add a key mapping to a child function, it is not applied to its paired function or the parent function.



In most cases, an unmapped function is unmapped because it is not meaningful in the observer frame for the key mapping. For example, the level observer movement functions are not meaningful in the observer frame key mappings.

To add a key mapping:

1. Optionally, filter the function list. For details, please see "Filtering the Function List," on page 46-7.
2. Select the function to which you want to add a key mapping.
3. Click the Add button (✚) above the Key Map window (Figure 46-2). The Key Entry dialog box opens (Figure 46-3).

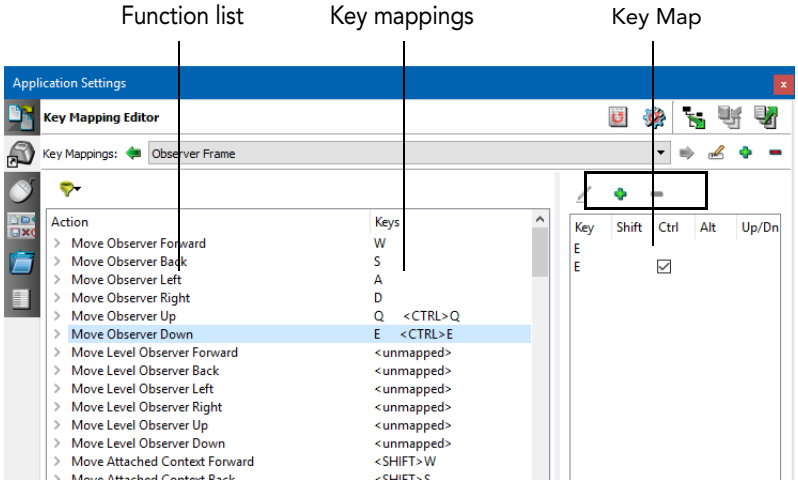


Figure 46-2. Key Mapping Editor

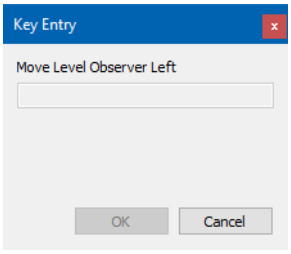


Figure 46-3. Key Entry dialog box

4. On the keyboard, press the key or key combination (such as **`Ctrl`**+key or **`Shift`**+key) that you want to map to this function.
5. Click OK. (If you choose a key combination that is already in use, VR-Vantage notifies you and asks how to proceed. For details about duplicate key mappings, please see "Using Combined Key Mappings," on page 46-6.) The mapping is added to the function and is listed in the Keys column of the function list and in the Key Map window.

46.3.2. Changing a Key Mapping

If a key mapping is assigned to a binary function, the key is automatically assigned to the child functions. You cannot edit the key mappings for the child functions. You can only change the parent's key mapping. However, if you add a key mapping to a child function, you can edit it.

To change a key mapping:

1. Select the function whose key mapping you want to change. Its key mappings get listed in the Key Map window (Figure 46-2).
2. In the Key Map window, select the key mapping that you want to change.
3. Click the Edit Key Mapping button (). The Key Entry dialog box opens (Figure 46-4). It shows the current key mapping.

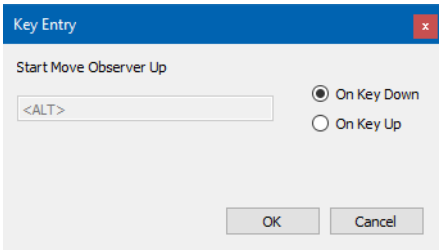


Figure 46-4. Key Entry dialog box

4. On the keyboard, press the key or key combination (such as **`Ctrl`**+key or **`Shift`**+key) that you want to map to this function.
5. Click OK. The mapping is changed.

46.3.3. Deleting a Key Mapping

To delete a key mapping:

1. Select the function whose key mapping you want to delete. Its key mappings get listed in the Key Map window (Figure 46-2).
2. In the Key Map window, select the key mapping that you want to delete.
3. Click the Delete button () above the Key Map window. The mapping is deleted. The function list is updated.

46.4. Using Combined Key Mappings

You can assign a key combination to more than one function. This allows you to combine the behaviors of two or more functions into one key combination. For example, the **Z** key detaches the observer from the attached simulation object. The **Space Bar** resets the observer view to the default view. Suppose that whenever you detach from a simulation object, you want the observer to return to the default view. You could map the Reset function to **Z**. **Z** is now mapped to two functions. When you press **Z**, VR-Vantage detaches from the attached simulation object and moves the observer to the default view.

To manage combined key mappings:

1. Add the appropriate key mapping to a function.
2. Click OK. The Key Already Mapped dialog box opens (Figure 46-5). It lists the functions that already use this key mapping.

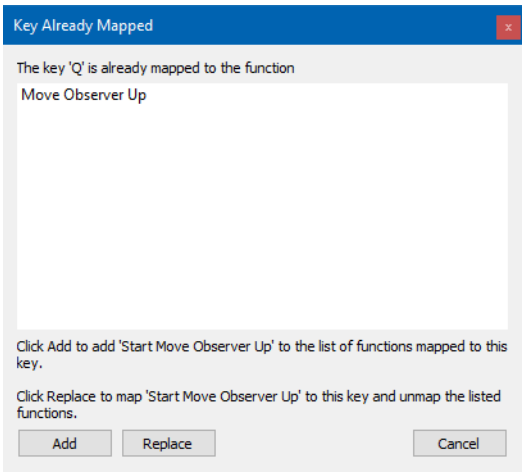


Figure 46-5. Key Already Mapped dialog box

3. If the functions listed are the ones with which you want to share this key mapping, click Add. The key mapping is added to the function.
4. If you do not want to share this key mapping with other functions and want to use it for the function that you are editing, click Replace. The key mapping is applied to the function you are editing and removed from the functions listed in the dialog box.
5. If you do not want to remove the key mapping from existing functions and do not want to share it with the other functions, click Cancel. Presumably you wanted a unique key mapping for the function you are editing and did not realize you chose one that was already in use. Choose a different key mapping for this function.

46.5. Filtering the Function List

You can filter the function list in the Key Mapping Editor. Functions are grouped thematically.

➤ To filter the function list, click the Filter button ( ) and select an option on the list.

46.6. Creating a Key Map

To create a key map:

1. Choose **Settings** → **Application**.The Application Settings dialog box opens.
2. Select the Key Mapping Editor page (Figure 46-1).
3. Click the Add button ( ). The Create New Key Map dialog box opens.
4. Type a name for the new key map.
5. Click OK.
6. Add key mappings for the movement actions as described in "Adding a Key Mapping," on page 46-3.

46.7. Deleting a Key Map

To delete a key map:

1. Choose **Settings** → **Application**.The Application Settings dialog box opens.
2. Select the Key Mapping Editor page (Figure 46-1).
3. In the key mapping list, select the key mapping that you want to delete.
4. Click the Delete button ( ).
5. Confirm that you want to delete the selected key map.



MAK Products Glossary

This glossary defines terms used in MAK product documentation.

absolute dead reckoning

A method of dead-reckoning in which an application uses the time stamps contained within state updates if the sender is using absolute time stamping. Contrast with relative dead reckoning.

aggregate

The combination of individual entities, aggregates, or both into a single object. For example, an organizational group such as a platoon.

API

Application Programming Interface.

articulated part

A part of an entity that is capable of movement relative to the entity, such as a turret or gun.

attached part

An object that is attached to another object, such as a rocket attached to a jet.

body coordinate frame

A coordinate framework centered on an entity. To represent the orientation of an entity, an application uses Euler angles or a rotation matrix that rotates from a reference frame to the body coordinate frame.

channel

A channel renders the view of an observer in a 3D visualization application.

clipping

A feature that prevents the display of terrain and objects or parts of objects, that are closer than or farther than specific distances from the observer.

clipping planes

The range of distances from the observer (near clipping and far clipping) in which an application clips objects.

culling

The process of discarding database objects which are not within view, and therefore need not be rendered and displayed.

Cartesian coordinates

A system for indicating location by means of three planes intersecting at right angles to one another at the origin.

Data Distribution Management

The set of HLA services used to specify the routing of data between publishers and subscribers. DDM adds greater control over the Declaration Management (DM) services that only specify filtering based on the class of the data (e.g., ground vehicle versus aircraft). DDM uses the data content to express regions of interest and regions of affect whose overlap establish a communication channel. An example is geographic filtering where a publisher indicates a region around the entity or effect and a subscriber indicates a region expressing its sensor range.

dead-reckoning

A process by which an application calculates the expected location of an entity during periods between state updates, based on velocity, acceleration, and rotational velocity.

DDM

Data Distribution Management.

Defense Modeling and Simulation Office

The former name of the executive secretariat for the Executive Council on Modeling and Simulation (EXCIMS), which provided a full-time focal point for information concerning Department of Defense modeling and simulation (M&S) activities. Renamed Modeling and Simulation Coordination Office. Currently the MSCO promulgates M&S policy, initiatives, and guidance to promote cooperation among DoD components to maximize efficiency and effectiveness.

DIS

Distributed Interactive Simulation.

DIS data packets

See Protocol Data Unit.

display engine

An object in an application that can render 3D data.

Distributed Interactive Simulation

The DIS protocol is a set of standards that govern how participating applications share information about the virtual world. The protocol specifies a set of packets, called protocol data units (PDUs), that communicate this information. Each PDU identifies the sender and contains other information depending on the PDU type. The DIS protocol also specifies when and how frequently PDUs are sent.

The DIS protocol has been superseded by the High-Level Architecture for use by the Department of Defense.

DMSO

See Defense Modeling and Simulation Office.

element definition

In VR-Forces and VR-Vantage, an element definition specifies all of the visualizers that control the various aspects of visualizing a scene element. For example, an element definition includes the main model or military symbology, trailing effects, cockpit displays, and so on.

emitter

A simulated device on an entity that emits electromagnetic radiation, for example, radar.

entity

An element in a simulation, such as a vehicle or a person, that is represented in the simulation through the issuance of state messages.

entity maintenance

A process by which the MAK Data Logger compensates for discontinuities following a time jump. The Logger sends out interim state messages for entities that were present before the time jump so that they do not time out before the next update message arrives.

entity-type

A list of seven components as specified by the DIS protocol and the HLA RPR FOM. If none of the seven components contains a wildcard (-1) value, then the entity type refers to a specific, narrowly-defined entity. If some components contain a wildcard, then the entity type refers to a class of entities. A larger number of wildcards indicates a broader, more general class.

The components of an entity-type are:

- Entity kind
- Domain
- Country
- Category
- Subcategory
- Specific
- Extra.

environmental process

According to the IEEE 1278.1a specification (DIS), environmental process PDUs communicate simple environment variables, small scale environmental updates, and embedded processes.

Euler angles

A set of three angles used to describe the orientation of an entity as a set of three successive rotations about three different orthogonal axes (x, y, and z). These angles specify successive rotations needed to transform from a reference coordinate system to the entity's body coordinate system.

event

An interaction between objects or between an object and the terrain, such as firing of a munition, or a collision of entities.

exercise

The DIS term for a one or more interacting simulation applications. Compare to federation execution in HLA.

exercise connection

The object (*DtExerciseConn*) through which a VR-Link application connects to the network. State messages are sent through the exercise connection and information about remote entities is received through the exercise connection. (All MAK products are VR-Link applications.)

exercise ID

A numeric identification for a DIS simulation exercise.

FED file

Federation Execution Data file. The Federation Execution Data is the subset of FOM data needed and read by the RTI. VR-Link applications also read the FED file.

federate

A connection to the RTI. Typically a single simulation application can be thought of as a federate.

federation

A group of HLA federates capable of playing in the same federation execution.

Federation Object Model

Defines the data content of a federation execution.

federation execution

The federation execution represents the actual operation, over time, of a subset of the federates and the RTI initialization data taken from a particular federation. A federation execution is the logical equivalent of a DIS simulation exercise.

field of view

Controls the perspective of the observer.

A wide field of view creates an effect like that of a wide-angle camera lens. Objects appear smaller and farther away from the observer, since the observer coverage spans a wider area. Depth become exaggerated.

A narrow field of view creates an effect like that of a telephoto lens. Objects appear larger and closer to the observer, and the overall scene depth appears flattened. The distances between objects appears compressed.

filter range

A setting that prevents distant entities from being processed while allowing distant terrain to appear normally.

FOM

See Federation Object Model.

FOM Module

In HLA Evolved, defines additional modular data content for a federation execution, usually extensions to an existing FOM. (Also supported by the HLA 1.3 and HLA 1516 versions of the MAK RTI. Per the HLA Evolved interface specification:

"A partial FOM (containing some or all OMT tables) that specifies a modular component of a FOM. A FOM module may contain classes not inherent to it but upon which the FOM module depends, i.e., superclasses to the modular components. These superclasses will be included in the FOM module either as complete or scaffolding definitions."

frame rate

The rate at which the application displays updated images.

ground clamping

A process by which the a 3D visualization application keeps an entity anchored to the surface of its terrain database, regardless of the altitude data contained in its entity state message.

geocentric coordinates

A coordinate system calculated with respect to the earth's center. The origin of the geocentric coordinate system is the center of the earth. The positive X-axis passes through the prime meridian at the equator; the positive Y-axis passes through 90 degrees east longitude at the equator; and the positive Z-axis passes though the north pole.

geodetic coordinates

A coordinate system in which position is determined relative to a reference ellipsoid, such as the surface of the earth at sea level. In MAK applications, geodetic coordinates consist of latitude and longitude in radians, and altitude in meters above the reference ellipsoid.

gridded data

Data that has been processed in a rectangular array of points, in X, Y or latitude/longitude, at which single data values define a two dimensional function. According to the IEEE 1278.1a specification, gridded data transmits information about large-scale or high-fidelity spatially and temporally varying ambient fields and about environmental processes and features.

guise

An alternative entity type used to display an object depending on the force ID. For example, a tank could look like an M1A1 to friendly forces and a T72 to hostile forces.

head-up display

A set of indicators and readouts superimposed onto a graphics display. Also called an overlay.

heartbeat

In DIS, the frequency with which current PDUs are sent to the network regardless of whether or not the entity's state has changed.

High-Level Architecture

The High Level Architecture (HLA) for simulations is a U. S. Department of Defense (DOD)-wide initiative to provide an architecture to support interoperability and reuse of simulations. The HLA supersedes DIS for the DOD.

HLA

See High-Level Architecture.

interaction

A message describing a simulation event. An interaction describes an event, it does not update an object's state.

Logger control PDUs

A set of DIS PDUs or HLA interactions that let you control the MAK Logger remotely.

logical range

A suite of TENA Resources, sharing a common object model, that work together for a given range event. Similar in concept to the HLA Federation.

Local RTI Component

The libraries on a computer that implement an RTI. A federate communicates with the HLA network through the LRC.

LRC

See Local RTI Component.

MAK Technologies Lisp

An adaptation of the Lisp language used in configuration files for MAK products.

message

A general term used to refer to DIS PDUs and HLA interactions and state updates. In TENA, a message is similar to an HLA interaction.

MIM

The MOM & Initialization Module (MIM) allows for extensions to be made to the HLA standard MOM. It is a subset of the FOM that contains OMT tables that describe the HLA MOM. The MIM also contains additional predefined HLA constructs such as object and interaction roots, data types, transportation types and dimensions. HLA specifies a standard MIM that is incorporated into all FDDs automatically by the RTI. The standard MIM can be replaced with a user-supplied MIM containing the standard MIM plus extensions.

model definition

In VR-Forces and VR-Vantage, a model definition specifies the 3D model and other basic attributes of a model. It is referenced by an element definition.

Modeling and Simulation Coordination Office

The executive secretariat for the Executive Council on Modeling and Simulation (EXCIMS), which provided a full-time focal point for information concerning Department of Defense modeling and simulation (M&S) activities. The MSCO promulgates M&S policy, initiatives, and guidance to promote cooperation among DoD components to maximize efficiency and effectiveness. (Formerly DMSO)

MSCO

Modeling and Simulation Coordination Office. (Formerly DMSO.)

MTL

See MAK Technologies Lisp.

orientation clamping

Adjusts the pitch and roll of an entity so that it appears properly seated when the terrain is inclined. For example, if a tank is moving horizontally across the face of a hill, orientation clamping prevents the tank from appearing level, and therefore, partially embedded into the hillside. Used with ground clamping.

object

An element in a simulation that has persistence, as opposed to an interaction, which is a transient element.

object handle

An integer that an application uses to identify a particular object in RTI service calls. An object handle is meaningful only to a particular federate. The same object can be known to different federates by different object handles.

object name

A character string that can be used to identify an object. In HLA, the object name is known to the RTI, and the RTI provides functions to find out an object's name, given its handle, and vice versa. Object names can be chosen by applications that register the objects with the RTI, however if you do not want to choose names for objects, the RTI will assign names for you.

observer

In MAK 3D applications, the point of view into the scene. (Called the eyepoint in MAK Stealth 6.x and earlier.)

PDU

See Protocol Data Unit.

Protocol Data Unit

A unit of data message (packet) that is passed on a network between DIS simulation applications.

radio transmitter

A simulated device on an entity, capable of transmitting radio communications.

radio receiver

A simulated device on an entity, capable of receiving radio communications.

Realtime Platform Reference FOM

An HLA reference FOM based on the DIS protocol.

recording

A Data Logger file that stores a history of the interactions in a simulation for playback.

reflected entity list

A list of entities simulated by remote applications (federates in HLA) and about which the local simulation has received information over the network.

reference FOM

A FOM designed to be used as a whole, or with modification, by a wide variety of similar federation executions.

relative dead reckoning

A method of dead reckoning in which an application approximates the location of an object based on the local time of receipt of the last state update message.

remote display engine

A display engine running in an application that is separate from the master application, and which is controlled by the master application.

RID file

See RTI Initialization Data.

RTI Initialization Data

The initialization data required by the RTI for operation.

Data required by an RTI during initialization, independent of the FOM being used. RID data is usually dependent on a specific implementation of the RTI.

RPR FOM

See Realtime Platform Reference FOM.

RTI

See Run-Time Infrastructure.

Run-Time Infrastructure

A library and other supporting software that implements the HLA interface specification. All federates communicate with one another in an HLA environment through RTI functions.

scene element

In VR-Vantage and VR-Forces, any object that is rendered in the scene, such as an entity model, interaction, or tactical graphic.

Simulation Interoperability Standards Organization

A group that seeks to promote modeling and simulation interoperability and reuse for the benefit of diverse M&S communities, including developers, procurers, and users, world-wide.

simulation time

In VR-Forces, simulation time is used in dead-reckoning of remote entities and thresholding of local entities. Typically, simulation time is set once during each iteration of the application's main simulation loop so that all entities are dead-reckoned based on the same value of current time.

SISO

See Simulation Interoperability Standards Organization.

smooth period

The period of time over which trajectory smoothing takes place.

smoothing

A method of ensuring that transitions from an entity's dead-reckoned position to its actual position are not so abrupt as to be visually disconcerting.

state

The current status of an object, including location, direction of movement, extent of damage, and so on.

tape

An alternative term for a Logger recording. Not a physical magnetic tape.

terrain following

In a 3D visualization application, causes the observer (eyepoint) to maintain a constant distance above the terrain surface. The observer's height changes in tandem with the peaks and valleys as it passes over the geography.

timeout

The period of time in which an application continues to display an entity after that entity's update messages have stopped appearing on the network. Time-outs are usually not used in HLA because there is no set frequency (no heart-beat) for transmitting messages.

timescale

A factor by which time is accelerated or slowed during playback of a Data Logger recording.

topographic coordinates

A right-handed Cartesian coordinate system whose X-Y plane is tangent to the earth's surface at the origin, with the positive X axis pointing north, the positive Y axis pointing east, and the positive Z axis pointing down. There are an infinite number of topographic coordinate systems – one for each point on the earth's surface.

topographic coordinate frame

A coordinate frame in the context of the terrain.

trajectory smoothing

A method used by to smooth positional discontinuities that could occur when new state updates arrive.

UDP port

A network channel through which an application sends and receives data for DIS exercises.

Universal Transverse Mercator

In general, a non-cartesian coordinate system in which the X, Y, and Z correspond to easting (nearly east), northing (nearly north), and height above an ellipsoid that approximates the surface of the earth.

UTM

See Universal Transverse Mercator.

view control messages

A set of programmatic messages that let you control the view in VR-Vantage applications remotely.



Index

--enableVSync command-line option 5-8
--noAppDataEntityTypeMap command-line option 34-9
-- command-line option 3-5, 27-4
--alphaBits command-line option 3-5
--antiAliasing command-line option 3-5, 5-8
--appDataDir command-line option 3-5
--autoConnect command-line option 3-5
--autoReconnectPeriod command-line option 3-5
--bindObservers command-line option 3-5
--col command-line option 27-4
--dataDir command-line option 3-5
--depthBuffer command-line option 3-5
--disableDynamicTerrain command-line option 3-6
--disableKDTrees command-line option 3-6
--discoveredModelPreloadFileName command-line option 3-6, 5-7
--displayEnginePort command-line option 3-6
--dispSetting command-line option 3-6, 6-9, 7-8
--enableOculus command-line option 3-6
--enableVSync command-line option 3-6
--encryptImages command-line option 27-4
--entDispSetting command-line option 3-6
--entityDefsDir command-line option 3-6, 33-15
--entTypeMap command-line option 3-6
--envSetting command-line option 3-6
--extendLevelZero command-line option 27-4
--extraArgs command-line option 3-6
--factoryRootDir command-line option 3-6
--file command-line option 27-4
--forceTextShaping command-line option 3-6
--frameRate command-line option 3-6
--fullScreen command-line option 3-6, 45-4
--generateFiles command-line option 27-5
--help command-line option 3-6, 27-5
--highestPriority command-line option 3-6
--ignore_rest command-line option 3-5, 27-4
--ignoreNonBaseTex command-line option 35-28
--ignoreZoomForSpeedtreeLod command-line option 3-9
--level command-line option 27-5
--loadObservers command-line option 3-7
--locale command-line option 3-6
--logFileName command-line option 3-6
--mainScreenNum command-line option 3-7
--mergeEntityTypeMap command-line option 3-7
--mergeHierarchyEntityDef command-line option 3-7, 33-15
--mergeHierarchyTacticalGraphicsDef command-line option 3-7, 33-15
--mergeHierarchyUnitDef command-line option 3-7, 33-15
--mergeLeafEntityDef command-line option 3-7, 33-15

--mergeLeafTacticalGraphicsDef command-line option 3-7, 33-15
--mergeLeafUnitDef command-line option 3-7, 33-15
--mergeModelDefs command-line option 3-7, 32-12
--mergeTacticalGraphicsTypeMap command-line option 3-7
--mergeUnitTypeMap command-line option 3-7
--missing command-line option 27-5
--modelPreloadFileName command-line option 3-7, 5-7
--noAppDataEntityDefs command-line option 3-8, 33-16
--noAppDataEntityTypeMap command-line option 3-8
--noAppDataModelDefs command-line option 3-8, 32-12
--noAppDataTacticalGraphicsDefs command-line option 3-8, 33-16
--noAppDataTacticalGraphicsTypeMap command-line option 3-8
--noAppDataUnitDefs command-line option 3-8, 33-16
--noGui command-line option 3-8
--notifyLevel command-line option 3-8
--plugin command-line option 3-8
--preLoadClouds command-line option 3-8, 11-11
--processEmats command-line option 27-5
--pseudoFullScreen command-line option 3-8, 45-4
--publishMftOnly command-line option 27-5
--regenerateAll command-line option 27-4
--row command-line option 27-6
--searchDir command-line option 27-5
--showConsole command-line option 3-5, 8-8
--slave command-line option 3-8
--stencilBits command-line option 3-8
--subDir command-line option 27-5
--suppressHatIntersectOnAttach command-line option 3-8, 16-6
--synchronizeOcean command-line option 3-8
--tcpSendTimeout command-line option 3-8
--textureSize command-line option 27-6
--threads command-line option 27-5
--unbatchedRendering command-line option 3-6
--unplug command-line option 3-9
--userDataDir command-line option 3-9
--useReverseZBuffer command-line option 3-9
--version command-line option 3-9, 27-5
--vtFile command-line option 27-5
--withDisplayEngine command-line option 3-9, 6-9, 7-8
-b command-line option 35-28
-e command-line option 3-6
-K command-line option 3-6
-r command-line option 35-29
-Z command-line option 3-9
-z command-line option 3-9

Symbols

@dis comment 26-26
@dis navigation_lights 26-26

Numerics

2525B icon 35-2
2D
  icon 19-14
    changing size 19-12
    configuring 35-2
    editing 35-2
    label 19-4
    transparency 19-14
  label, shadow text 19-9
  observer, attachment behavior 16-14
  observer coordinate system 15-7
  projection, switching to 3D 14-2
2D icon, color 19-8
3D
  model, licensing 1-10
  observer coordinate systems 15-7
  projection, switching to 2D 14-2
3D Colorized Models model set
  color, changing 19-9
3DS 30-4
  file 25-3
900_ExtImagery.xml 28-18

A

-A command-line option 27-4
Absolute mode 16-6, 16-7
absolute time stamps 4-6, 4-7
acceleration
  displaying 19-4
  specifying measurement unit 45-6
active, stereo 6-28
add image attribute 25-10
Add Model Definition dialog box 32-5, 37-2
Add Props page 25-23
Add Terrain Patch dialog box 25-7, 30-9
Add Visualizer Definition Attribute window 33-11
adding
  attribute, visual definition 33-11
  channel 6-7
  cockpit model 35-10
  connection 4-3
  entity bitmap image 35-9
  environment condition 11-7
  feature layer 25-4, 25-14
  key mapping 46-3
  model mapping 34-4
  observer 14-6, 14-7
  ocean layer 25-16, 25-17
  parameter
    model definition 32-8
    schema 31-5
  prop 25-21
    from feature layer 25-23
  terrain patch to terrain 25-7
  terrain server 26-5
  visual definition 33-7
  window 6-5
address
  multicast 4-7
  target 4-6
adjusting, sound volume 23-3
aggregate 20-2
  See also, unit
  aggregated 20-2
  icon, displaying 20-2
agility, terrain 1-5
alpha 35-27
alpha bits 3-5
altitude 19-3
  entity, displaying 19-10
  intervisibility line 22-11
altitude 19-3 (continued)
  observer, filtering simulation objects by 19-17
  specifying measurement unit 45-6
  switching from 2D to 3D 14-2
  vertex, displaying 40-5
altitude point
  changing color 24-11
  deleting 24-10
  inserting 24-10
ambient
  color 13-3
  light 11-3
  occlusion map 1-6, 29-10
  value, material 13-3
anaglyphic stereo 6-28
  configuring 6-29
Angle of Attack 35-13
angleSegmentMax parameter 36-5
animation
  DI-Guy, choosing 34-10
  model 39-11
  observer view transition 15-29
animation control message 41-6, 41-7
anti-aliasing 3-5, 5-8
AOA updater 35-13
appending, saved views to view list 15-32
application
  data, specifying location of 3-5
  ID, DIS 4-6
  seeing observer 14-10
Application Number, HLA 4-5
Application Settings dialog box
  File Caching Settings page 5-7, 29-2, 29-5
  General Application Settings page 6-31
  GUI Layout page 45-4
  Key Mapping Editor page 46-3, 46-7
  Mouse Mappings page 15-14
applying updated model definitions 32-8
arc, segment 36-5
area
  feature, list of 24-4
areal, feature 26-15
articulated part 39-2, 39-7
  attaching to 16-4
  mapping in CIGI 41-5
Articulated Part Control 41-3
asynchHostTest.py 43-2

asynchronous
  CIGI operation 41-2
  IO, DIS 4-7
atlas, texture 35-21
attach 16-6
  observer, articulated part 16-4
attach mode 1-4, 15-3, 15-8, 16-6
  2D behavior 16-14
  Absolute 16-7
  Follow 16-7
  Mimic 16-8
  Mimic Location Track 16-9
  Mimic Track 16-9
  movement, attached context 15-8
  prop, affect of 16-5
  Space Follow 16-10
  Tether 16-11
  Tether Location Track 16-12
  Tether Track 16-12
  Track 16-13
attached, context 15-8
Attached Near Clip Altitude, attribute 6-15
attaching observer
  to object, with keyboard 16-3
  to simulation object or prop 16-2
  tutorial 16-16
attachment
  list, filtering 16-5
  order 16-3
  preservation of view changes 16-3
Attachment Panel 16-2
  filtering attachment list 16-5
attribute
  Attached Near Clip Altitude 6-15
  BOYSHP 26-21
  buoy 26-21
    building model definition 26-21
  cockpit updater 35-10
  COLOUR 26-21
  COLPAT 26-21
  DESIGNATOR 26-23
  destroyedsymbol 35-7
  Dynamic Near Clip Attached Policy 6-15
  Dynamic Water Visibility Enabled 6-21
  Dynamic Water Visibility Maximum 6-21
  Fixed Near Clip 6-15
  Fixed Near Clip When Attached 6-15
attribute (continued)
  image 25-10
  imagesymbolmap 35-6, 35-9
  light characteristic 26-29
  LITCHR 26-29
  marker-scale 35-26
  modeldefinitionschema 35-6
  Near Far Clip Policy 6-15
  OBJNAM 26-23
  parent 35-5
  Reduce Z Fighting Ratio 6-15
  sensor 6-12
  SIGGRP 26-29
  signal group 26-29
  signal period 26-29
  SIGPER 26-29
  SIGSEQ 26-29
  TOPSHP 26-28
  type 35-5
  visual definition 33-8
    adding 33-11
  WidgetTheme 35-6
  window 6-11
audio 23-2
Audio Settings dialog box 23-2, 23-3, 23-5, 23-6, 23-7, 23-9
Audio Settings page 23-2, 23-3
authentication, proxy 26-5
auto connection 3-13
Auto Generate 25-21
auto start, video stream 8-4

B

-B command-line option 3-5
background
  2D label 19-9
  intervisibility information 22-11
beacon 26-29
  model definition 26-21
  seasonal 26-27
  streaming 26-20, 26-25
  topmark 26-28
  wake 11-16
bead
  DOF 39-7
  switch 39-8
bearing, intervisibility line 22-11
best practice, model design 35-27
binary key mapping 46-3

bit
  depth 3-5
  depth and stencil 5-8
bitmap 25-3
  adding new 35-9
  displaying 25-9
  icon 35-3
blend image attribute 25-10
blur 12-2, 12-4
BMP 25-3
body coordinates 15-3
border, icon 19-11
Boundary Generation Tool 26-16
  running 26-17
  using 26-17
boundary.txt 26-17
bounding volume
  unit, color 20-10
BOYSHP, attribute 26-21
breaking, wave 11-17
brightness 12-4
broadcasting
  state update 14-10
  stealth position and orientation 19-16
browser, streaming video to 8-13
buffer, video stream 8-8
buffer size, netsocket 4-7
building
  extruded 26-18
  prop type 25-27
  terrain 25-7
bump map 1-6, 29-10
buoy
  attributes 26-21
  daymark 26-28
  designator 26-23
  model definition 26-20, 26-21
    using texture atlas 26-24
  seasonal 26-27
  signal sequence 26-29
  streaming 26-20
  topmark 26-28
  unique 26-23
  wake 11-16
buoyancy
  enabling or disabling 21-24
  LOD distance 21-24
button, common for model definitions 45-5

C

-C command-line option 3-5
-c command-line option 27-4
cache
  clearing 29-5
  clearing model instancing 5-7
  model instance 5-7
  osgEarth Cache tool 29-3
  streaming terrain 29-8
caching
  earth file, offline 29-4
  earth file data 29-3
  file 29-2
  terrain 29-3
  terrain server data 29-2
camera
  effects 12-2
  jitter 15-23
  moving 15-13
  position and orientation 6-26
  shaking 15-23
  ways to move 15-12
Camera Orientation Offset 6-12
Camera Position Offset 6-12, 6-26
CDB
  configuration files 28-17
  earth file 28-2
  lightmap 28-6
  lightpoint 28-9, 28-10
  osgEarth 28-2
CDB database 28-2
CDB raster driver 28-3
Celestial Sphere Control 41-3
changing
  channel attributes 6-12
  color
    altitude point 24-11
    force 19-9
  field of view 6-19
  gain 15-27
  icon size 19-12
  image display order 25-13
  intervisibility object 22-8
  key mapping 46-5
  model definition, prop 30-11
  model set 19-11
  observer mode 14-3

changing (continued)
prop
model definition 25-27
position or orientation 25-27
type 25-27
unit of measurement 45-6
viewport 6-20
weather, in host emulator 42-9
window attributes 6-11
channel 6-4
adding 6-7
attribute
editing 7-2
sensor 6-12
changing attributes 6-12
keyword 6-12
compass 15-5
keywords 6-22
multichannel configuration, loading 6-27
multiple 6-24
observer 14-6
projection resize policy 6-16
property 6-12
removing 6-8
sound mapping 23-8
Channel Name 6-12
ChannelKeyword parameter 35-12
check box, Color by Force Type 19-8
child, element definition 33-2
Choose a Terrain, window 3-2
chop, wave 11-12
choppiness, ocean 11-15
CIGI 1-3, 41-2
adding effect in host emulator 42-8
articulated parts, mapping 41-5
changing model state 42-6
channel keyword 6-22
Class Library 41-2
component control message, formats 41-6
configuring tidal stream speed and direction 35-16
connection types 41-2
creating simulated entity on host emulator 42-6
driver 41-2
Global Terrain Surface Control 41-10
CIGI 1-3, 41-2 (continued)
host emulator 41-2, 42-2
connecting to 42-4
starting 42-3
tutorial 42-2
mapping database 41-11
mappings 41-1
model mapping 41-4
model state 41-7
network configurations 41-13
packet
supported 41-3
user-defined 41-3
socket 41-2
synchronous and asynchronous 41-2
system component control messages 41-8
view, mapping to observer mode 41-12
view component control 41-9
view control packet 41-12
views 41-12
weather control 42-9
weather control packet 44-9
Common Image Generator Interface*, See also* CIGI
CIGI Connections page 42-4
CIGI control message, mapping 41-6
CIGI Database Mapping page 41-11, 42-4
CIGI Entity Models page 41-4, 41-5
CIGI Entity Type Mapping parameter 4-8
CIGI Global Terrain Surface Components page 41-10
CIGI Light Point Groups page 41-10
CIGI Mapping Settings dialog box 41-4, 41-5, 41-6, 41-11, 41-12
CIGI Messages parameter 4-8
CIGI Model Animations page 41-7
CIGI Model State mappings page 41-7
CIGI Observer Mode Mapping page 41-12
CIGI Version parameter 4-8
CIGI View Components page 41-9
cigiView1 keyword 6-22
clamp image attribute 25-11
clamp to border image attribute 25-11
clamp to edge image attribute 25-11
clamping, entities 21-2
class
DtLightSignalSequenceGenerator 29
DtLightSignalSequenceGenerator-Creator 29

clearing
file cache 29-5
model instancing cache 5-7
view list 15-30
clipping plane 35-24
cloud
cover 11-5, 11-11
disabling 11-22
preloading 11-11
skybox cube map 13-13
cockpit, enabling with keyword 6-22
cockpit display 17-6, 34-2, 35-10
adding new 35-10
attaching at startup 16-15
element definition 33-2
enabling or disabling 17-3
installing 35-11
model definition, creating 35-11
updater 35-10
attribute 35-10
Cockpit Display Settings page 17-5
cockpitOnly keyword 6-22
collapsing
unit 20-2, 20-4
to root 20-4
color
2D icon 19-8
2D label background 19-9
altitude point
changing 24-11
deleting 24-10
inserting 24-10
ambient 13-3
compass 15-5
diffuse 13-3
emitter volume, pulse rate 36-2
entity label 19-3
fog 11-5, 11-10
force, changing 19-9
intervisibility, configuring 22-9
light point group, CIGI 41-10
pattern 26-21
radio communication line 21-18
represents emitter volume frequency 36-2
specular 13-3
squawk indicator 21-18
terrain, configuring 24-10
unit bounding volume 20-10
Color by Force Type check box 19-8
coloring
draw call 5-11
terrain 24-9
COLOUR attribute 26-21
COLPAT attribute 26-21
combining, key mapping 46-6
command line
loading scene 3-9
loading terrain 3-9
options 3-5
starting VR-Vantage 3-4
command-line option
-- 3-5, 27-4
-A 27-4
--alphaBits 3-5
--antiAliasing 3-5, 5-8
--appDataDir 3-5
--autoConnect 3-5
--autoReconnectPeriod 3-5
-B 3-5
-b 35-28
--bindObservers 3-5
-C 3-5
-c 27-4
--col 27-4
--dataDir 3-5
--depthBuffer 3-5
--disableDynamicTerrain 3-6
--disableKDTrees 3-6
--discoveredModelPreloadFileName 3-6, 5-7
--displayEnginePort 3-6
--dispSetting 3-6, 6-9, 7-8
-E 3-6
-e 3-6, 27-4
--enableOculus 3-6
--enableVSync 5-8
--enableVSync 3-6
--encryptImages 27-4
--entDispSetting 3-6
--entityDefsDir 3-6, 33-15
--entTypeMap 3-6
--envSetting 3-6
--extendLevelZero 27-4
--extraArgs 3-6
-F 3-6
-f 27-4
--factoryRootDir 3-6
--file 27-4
--forceTextShaping 3-6

command-line option (continued)
--frameRate 3-6
--fullScreen 3-6, 45-4
-G 3-6
-g 27-5
--generateFiles 27-5
-h 3-6, 27-5
--help 3-6, 27-5
--highestPriority 3-6
-I 3-6
--ignore_rest 3-5, 27-4
--ignoreNonBaseTex 35-28
--ignoreZoomForSpeedtreeLod 3-9
-K 3-6
-L 3-7
-l 3-6, 27-5
--level 27-5
--loadObservers 3-7
--locale 3-6
--logFileName 3-6
-m 27-5
--mainScreenNum 3-7
--mergeEntityTypeMap 3-7
--mergeHierarchyEntityDef 3-7, 33-15
--mergeHierarchyTacticalGraphicsDef 3-7, 33-15
--mergeHierarchyUnitDef 3-7, 33-15
--mergeLeafEntityDef 3-7, 33-15
--mergeLeafTacticalGraphicsDef 3-7, 33-15
--mergeLeafUnitDef 3-7, 33-15
--mergeModelDefs 3-7, 32-12
--mergeTacticalGraphicsTypeMap 3-7
--mergeUnitTypeMap 3-7
--missing 27-5
--modelPreloadFileName 3-7, 5-7
-n 3-8
--noAppDataEntityDefs 3-8, 33-16
--noAppDataEntityTypeMap 34-9
--noAppDataEntityTypeMap 3-8
--noAppDataModelDefs 3-8, 32-12
--noAppDataTacticalGraphicsDefs 3-8, 33-16
--noAppDataTacticalGraphics-TypeMap 3-8
--noAppDataUnitDefs 3-8, 33-16
--noGui 3-8
--notifyLevel 3-8
-O 3-8
-p 27-5
command-line option (continued)
--plugin 3-8
--preLoadClouds 3-8, 11-11
--processEmats 27-5
--pseudoFullScreen 3-8, 45-4
--publishMftOnly 27-5
-r 27-5, 35-29
--regenerateAll 27-4
--row 27-6
-S 3-8, 16-6
-s 3-8, 27-5
--searchDir 27-5
--showConsole 3-5, 8-8
--slave 3-8
--stencilBits 3-8
--subDir 27-5
--suppressHatIntersectOnAttach 3-8, 16-6
--synchronizeOcean 3-8
-t 27-5
--tcpSendTimeout 3-8
--textureSize 27-6
--threads 27-5
-u 27-5
--unbatchedRendering 3-6
--unplug 3-9
--userDataDir 3-9
--useReverseZBuffer 3-9
-v 3-9, 27-5
--version 3-9, 27-5
--vtFile 27-5
-W 3-9, 6-9, 7-8
-w 27-6
--withDisplayEngine 3-9, 6-9, 7-8
-x 27-6
-Z 3-9
-z 3-9
comment, @dis 26-26
Common Image Generator Interface 1-3, 41-2
communication
line 21-15
line-of-sight 21-17
compass
color 15-5
configuring 15-5
keyword 6-22
location 15-5
model definition 15-5
observer 15-4

compiling, incremental 5-11
Component Control 41-3
component control message 41-6
  formats for 41-6
Component Visualization 29-13
composing, terrain 25-2
composite effects maps 29-12
compressing, model and image files 35-28
compression, texture 29-5
cone, gimbaled sensor 17-9
configName parameter 35-21, 35-23
configuration
  CIGI 41-13
  display engine 6-5
    loading 6-9
    saving 6-8
  entity display 3-6
  environment 3-6
  file, RTI 2-9
  image generator 41-13
  multichannel
    loading 6-27
    saving 6-27
configuring
  2D icon 35-2
  compass 15-5
  contour lines 24-12
  depth of field 21-22
  display 3-6
  display engines 6-4
  earth file 26-2
  emitter volume 36-2
    segments 36-4
  GDB soil type 24-8
  grid lines 24-7
  ground clamping 21-3
  HLA connection 4-4
  implicit DDM 4-6
  instrument panel 17-7
  interest management 19-17
  intervisibility objects 22-9
  LOD 29-8
  model definition 32-2
  multichannel display 6-24
  ocean planar reflection 13-8
  orientation clamping 21-3
  projection resize policy 6-16
  props 25-4
  pulse rate 36-2
  schema 31-2
configuring (continued)
  sensor effects 12-4
  shadows 13-18
  ship wakes 35-14
  stereo
    anaglyphic 6-29
    polarized 6-30
  system for MAK RTI 2-9
  terrain color settings 24-10
  track history 21-9
  video stream 8-4
  virtual reality 9-6
  weather 11-9
Connect To Display Engine dialog box 6-23
connecting
  automatically 3-13
  display engine to 6-23
  emulator, CIGI host 42-4
  simulation network to 3-11, 4-2
  to network automatically 3-5
  to simulation network to 3-14
connection 4-2
  adding 4-3
  automatic 3-13
  deleting 4-4
  DIS, device address 4-6
  HLA, configuring 4-4
  response to 6-4
  simulation 3-14
  stopping 3-14
Connection Settings dialog box 19-17
Connection Status icon 3-14
Connection Status page 3-11, 3-13, 3-14, 4-2
constraint 16-6
  view 15-25
constructing, terrain 25-7
context-sensitive menu 45-5
contour line
  displaying 24-11
  spacing 24-12
  thickness 24-12
Contour Lines page 24-12
contrail 21-6
contrast 12-4
control, remote 1-6
control object, count of 19-15
controlling, remote 15-32
conventions, manual xxix

coordinate
observer 15-3
UV 25-11
coordinate system
3D 15-7
changing unit of 45-6
copying
model definition 32-7
schema 31-4
count
object 19-15
simulation object 19-15
Create Database Mapping dialog box 41-11
Create New Art Part Mapping dialog box 41-5
Create New Component Mapping dialog box 41-6
Create New Entity Type Model Mapping dialog box 34-4
creating
CIGI ownship entity 42-5
element definition 33-5, 37-6
inset view 17-2
key map 46-7
MEDF and MEIF files 35-28
model, guidelines for 35-27
model definition 32-2, 32-5, 37-2, 38-2
cockpit display 35-11
observer mode 14-4
scene 11-2
schema 31-3
terrain 25-5
crepuscular ray 13-11
CTDB file 25-3
cube map, recalculating 13-13
culling, face front 21-25
curvature of the earth
display, mode 21-17
cut-in site 26-16

D

data
caching terrain 29-2
directory, specifying location of 3-5
elevation 25-3
replicating 6-3
streamed, processing 26-10
synchronizing 3-17
data distribution management 19-16, 19-17
See also DDM
database
CDB 28-2
mapping for CIGI 41-11
terrain 1-5
provided 23-2
dataset
CDB
geospecific, CDB 28-6
elevation, CDB 28-3
extended imagery, CDB 28-15
geotypical, CDB 28-13
imagery, CDB 28-5
virtual and tiled 27-4
Datasets.xml 28-17
date
setting 11-3
synchronizing with VR-Forces 11-22
time scale 11-4
Date and Time page 11-3, 11-4
day, time 11-3
daylight, controlling light 13-6
daymark, buoy 26-28
daymark_attr_map.csv 26-28
daymark_layer_list.csv 26-28
daymark.dtxc 26-28
DDM 19-16, 19-17
region size 19-17
DDS texture 29-5, 29-6
flipping 25-8, 35-23
dead-reckoning 21-4
debug overlay 5-13
debugging
earth file 29-9
shaders 29-13
terrain 29-15
decal effects 21-6, 34-2
enabling and disabling 21-7
models 21-8
decal image attribute 25-10
decreasing speed of navigation 15-25
default
environment condition 11-7
settings, restoring 3-15
default_ModelPreload.mpsx, file 5-6
DefaultEllipsoidArc, model definition 36-2
definition
element 33-2

definition (continued)
model, buoy 26-20, 26-21
degrees of freedom 39-7
deleting
altitude point 24-10
channel 6-8
element definition 33-13
environment condition 11-8
intervisibility object 22-12
key map 46-7
key mapping 46-5
model definition 32-6
model mapping 34-7
observer 14-9
mode 14-5
parameter
model definition 32-11
schema 31-6
schema 31-4
simulation, connection 4-4
terrain, cached 29-5
view 15-30
window 6-8
demonstration, mode 45-4
depth
bits 3-5
buffer 5-8
focal 21-20
depth of field 21-20
configuring 21-22
enabling and disabling 21-20
DESIGNATOR, attribute 26-23
designator, buoy 26-23
destination address, DIS 4-6
destroyedsymbol attribute 35-7
detaching, observer from object 16-5
detonation
adding in host emulator 42-8
effects 21-11, 34-2
event
mapping sound effects to 23-4
sound effects 23-2
line 21-12
result, sound mapping 23-8
sound mapping 23-8
Detonation Sound Editor page 23-9
Developer Debug Overlay 28-12
developer debug overlay 5-13, 28-9
Device Address 4-6
dialog box
Add Model Definition 32-5, 37-2
Add Terrain Patch 25-7, 30-9
Application Settings
File Caching Settings page 5-7, 29-2, 29-5
General Application Settings page 6-31
GUI Layout page 45-4
Key Mapping Editor page 46-3, 46-7
Mouse Mappings page 15-14
Audio Settings 23-2, 23-3, 23-5, 23-6, 23-7, 23-9
CIGI Mapping Settings 41-4, 41-5, 41-6, 41-11, 41-12
Connect To Display Engine 6-23
Connection Settings 19-17
Create Database Mapping 41-11
Create New Art Part Mapping 41-5
Create New Component Mapping 41-6
Create New Entity Type Model Mapping 34-4
Display Settings 3-15
Cockpit Display Settings page 17-5
Display Units Settings page 45-8
Entity Display Settings page 19-5, 19-6, 19-9, 19-10, 19-12, 19-13, 19-16, 21-2, 21-3, 21-5, 21-14, 21-23, 21-25
Grid Lines Settings page 24-7
Interest Management Settings page 19-18
Intervisibility Settings page 22-6, 22-7, 22-9, 22-10, 22-11
Lighting Render Settings page 13-3, 13-4
Loader Settings page 29-7
Observer Settings page 11-13, 11-19, 13-6, 14-3, 14-4, 14-5, 14-6, 14-9, 15-5, 15-15, 15-23, 15-24, 15-27, 17-9, 18-4, 19-15, 21-7, 21-8, 21-11, 21-13, 21-21, 24-2, 24-6
Ocean Settings page 11-16, 11-17, 11-20, 11-21, 13-8, 13-13, 25-19, 25-20
Radio Communications Settings page 21-16, 21-17, 21-18

dialog box (continued)
Display Settings 3-15
Render Settings page 5-4, 5-5, 5-10, 5-11, 6-17, 13-12, 18-6, 18-7, 18-8, 19-14, 21-22, 24-14, 28-9, 28-12, 29-14
Sensor Settings page 12-5, 12-6
Shadow Settings page 13-17, 13-18
SpeedTree Settings page 35-24
Symbol Decoration Settings page 19-7, 19-8
Tactical Graphics Display Settings page 40-4, 40-5, 40-6
Track History Settings page 21-10
Unit Display Settings page 20-6, 20-9, 20-10
Duplicate Model Definition 32-7
Experimental Features 11-17, 15-21, 15-22
License Setup 2-6
osgEarth Profile 29-9
Scene Settings 11-6, 11-7, 11-8, 11-9, 11-10, 11-11
Simulation Connections 3-12, 4-2
Simulation Object Type Mappings 34-2
Terrain Settings
Add Props page 25-23
Edit Existing Props page 25-25, 25-26, 25-27, 30-11, 38-10
Extract Ocean page 25-16
Extract Props page 25-21, 30-9
Raster Maps page 25-9, 25-13
SpeedTree Randomization page 35-26
Terrain Contents page 25-14, 25-17, 25-18, 26-3, 29-9
Visible Surfaces page 24-8
Video Streams 8-3, 8-4, 8-5, 8-13
Visual Model Editors 32-2
diffuse, color 13-3
DI-Guy
choosing animation 34-10
instancing 5-12
license 1-7
model definition 32-5
performance 5-12
DI-Guy Settings page 5-12
direction
tidal current 11-12, 11-15
wind 11-9
DirectX 29-6
DIS
appearance state 41-7
application ID 4-6
asynchronous IO 4-7
connecting to 3-11
connection 4-3
settings 4-6
destination address 4-6
Enable Timeout Processing 4-7
encoding information in OpenFlight format 39-2
enumerations document 30-5
exercise ID 4-6
Listen for View Controls 4-7
multicast address 4-7
multicast TTL 4-7
Override Global View Control Setting 4-7
port 4-6
Publish Observers 4-7
receive buffer size 4-7
send buffer size 4-7
Show Self Published Objects 4-7
site number 4-6
Timeout Interval 4-7
timeout interval 4-7
Use Absolute Time Stamps 4-7
versions supported 1-3
WRM specification 39-1
DIS Connections page 14-11
DIS version 4-6
disabling
buoyancy 21-24
cockpit displays 17-3
decal effects 21-7
entity labels 19-6
environment 11-22
HLA advisories 4-6
intervisibility 22-10
radio communication lines 21-16
shadows 13-16
speed scaling 15-25, 15-26
squawk indicators 21-16
tactical smoke 21-13
terrain dragging 15-15
terrain scaling 18-6
texture compression 29-5
track histories for all simulation objects 21-8
trailing effects 21-7

disabling (continued)
wireframe mode 24-13
disaggregated aggregate 20-2
disconnecting
display engine 6-24
simulation network from 3-14
display
dragging 15-16
field of view 6-19
Linux 6-32
loading configuration 3-6
navigating 15-3
perspective 6-19
unit, changing 45-6
display engine 6-3, 6-4
configuration 6-5
loading 6-9, 7-2
loading multiple 6-9, 7-8
saving 6-8, 7-2
connecting to 6-23
connection states 6-4
data, synchronizing 6-3
disconnecting from 6-24
remote, connecting to 7-2
starting 6-22
tutorial 7-2
window, adding 6-5
Display Engine Configuration Editor Panel 6-6, 6-7, 6-8, 6-9, 6-11, 6-13, 6-23, 6-24, 6-27, 14-6
display mode, curvature of the earth 21-17
Display Settings dialog box 3-15
Cockpit Display Settings page 17-5
Display Units Settings page 45-8
Entity Display Settings page 19-5, 19-6, 19-9, 19-10, 19-12, 19-13, 19-16, 21-2, 21-3, 21-5, 21-14, 21-23, 21-25
Grid Lines Settings page 24-7
High Dynamic Range Settings page 13-14
Interest Management Settings page 19-18
Intervisibility Settings page 22-6, 22-7, 22-9, 22-10, 22-11
Lighting Render Settings page 13-3, 13-4
Loader Settings page 29-7
Display Settings dialog box 3-15 (continued)
Observer Settings page 11-13, 11-19, 13-6, 14-3, 14-4, 14-5, 14-6, 14-9, 15-5, 15-15, 15-23, 15-24, 15-27, 17-9, 18-4, 19-15, 21-7, 21-8, 21-11, 21-13, 21-21, 24-2, 24-6
Ocean Settings page 11-16, 11-17, 11-20, 11-21, 13-8, 13-13, 25-19, 25-20
Radio Communications Settings page 21-16, 21-17, 21-18
Render Settings page 5-4, 5-5, 5-10, 5-11, 6-17, 13-12, 18-6, 18-7, 18-8, 19-14, 21-22, 24-14, 28-9, 28-12, 29-14
Sensor Settings page 12-5, 12-6
Shadow Settings page 13-17, 13-18
SpeedTree Settings page 35-24
Symbol Decoration Settings page 19-7, 19-8
Tactical Graphics Display Settings page 40-4, 40-5, 40-6
Track History Settings page 21-10
Unit Display Settings page 20-6, 20-9, 20-10
Display Settings Toolbar 19-8
Display Units Settings page 19-10, 40-5, 45-8
displaying
aggregate icon 20-2
aggregates 20-2
contour lines 24-11
entity information 19-3
feature data 24-2
full screen 45-4
grid line 24-6
height-above-terrain lines 19-10, 40-5
intervisibility
lines 22-10
object 22-7
label
2D 19-7
entity 19-4
laser designator 21-26
objects 40-2, 40-4
other MAK 3D GUI viewers 14-11
panel 45-3
performance statistics 5-3

displaying (continued)
raster maps 25-9
segment outlines 36-6
stealth models 19-16
tactical graphics 40-4
toolbars 45-3
track histories 21-8
trailing effects 21-6
transient intervisibility 22-6
units 20-2
vertex labels 40-5
view 15-29
distance, specifying unit 45-6
Distance To Attached, policy 6-15
Distributed Interactive Simulation 1-3
See also DIS
DLL
cockpit display 35-10
installing cockpit display, installing 35-11
dockable panel 45-2
docking, panel 45-2
DOF bead 39-7
Douglas sea state 11-12, 11-14
dragging
terrain 15-14
view 15-16
draw call, coloring 5-11
driver
CIGI 41-2
OGR 26-12
osgEarth 26-9
Driver Frequency parameter 4-8
Driver Mode parameter 4-8
DtCigiDiGuyCtrl.h header file 3, 9
DTED
file 25-3
loading 25-6
loading multiple 26-9
support for 25-5
DtLightSignalSequenceGenerator class 29
DtLightSignalSequenceGeneratorCreator class 29
DtLightSignalSequenceGenerator.h header file 29
dtxc, file 35-23
duplicate, key mapping 46-6
Duplicate Model Definition dialog box 32-7
dust cloud 21-6
dynamic lighting 13-5
Dynamic Near Clip Attached Policy, attribute 6-15
dynamic ocean 1-6, 11-12
enabling and disabling 11-13
layer 25-15
wake and spray, disabling 21-23
wakes 35-14
dynamic terrain 3-6
Dynamic Water Visibility Enabled, attribute 6-21
Dynamic Water Visibility Maximum, attribute 6-21

E

-E command-line option 3-6
-e command-line option 27-4
earth file 26-2
adding mask 26-16
areal features 26-15
buoys and beacons 26-20
caching 29-3
offline 29-4
CDB 28-2
debugging 29-9
including files in 26-8
loading DTED 26-9
template 29-4
template, earth file 26-8
echelon level, ghosted icons by 20-6
ECW 25-3
Edit Existing Props page 25-22, 25-25, 25-26, 25-27, 30-11, 38-10
editing
channel attributes 6-12, 7-2
element definition 33-7
environment condition 11-8
frustum 6-19
icon, 2D 35-2
intervisibility object 22-8
key maps 46-2, 46-3
model definition 32-2, 32-8
model mapping 34-5
observer mode 14-5
parameter
model definition 32-10
schema 31-6
visual definition 33-8
parameter 33-10
window attributes 6-11

effect
adding in host emulator 42-8
fire and detonation 21-11
lighting 1-6, 29-10
mapping 34-2
model 30-4
ocean 11-12
trailing 21-6
effect map, composite 29-12
electromagnetic emissions 21-14
segment angle 36-5
elem file 33-16
element
mask 26-16
model 26-11, 35-26
option 26-19
parent 33-8
element definition 33-2
adding visual definition 33-7
common buttons 45-5
creating 33-5, 37-6
deleting 33-13
editing 33-7
exporting 33-16
exporting hierarchy and leaves 33-17
exporting to elem file 33-16
filtering list in Simulation Object Type Mappings dialog box 34-7
importing 33-14, 33-15
saving 33-13
simulation object 33-2
elevation
coloring terrain by 24-9
data 25-3
intervisibility point, configuring 22-9
Elevation Color page 24-9, 24-10, 24-11
elevation dataset, CDB 28-3
elevation sub-dataset
CDB
CDB, elevation sub-dataset 28-4
embarked
entities 19-15
simulation object 19-15
embedded window 6-4
EM-Emission update message 36-3
emitter, systems 21-14
Emitter Power 36-3
emitter volume
color 36-2
configuring 36-2
emitter volume (continued)
outlines 36-6
pulse rate 36-2
radius 36-3
segment angle 36-5
segment size 36-4
emulator
CIGI host 41-2, 42-2
starting 42-3
Enable Timeout Processing, DIS 4-7
enabling
buoyancy 21-24
cockpit displays 17-3
decal effects 21-7
entity labels 19-6
HLA advisories 4-6
HMD 9-3
intervisibility 22-10
marine effects 11-13
radio communication lines 21-16
remote weather effects 11-22
shadows 13-16
sound effects 23-2
speed scaling 15-25, 15-26
squawk indicator 21-16
tactical smoke 21-13
terrain scaling 18-6
texture compression 29-5
track histories for all simulation objects 21-8
trailing effects 21-7
wireframe mode 24-13
entity
aggregate 20-2
altitude, displaying 19-10
animation control in CIGI 41-7
appearance 39-2
articulated parts, mapping for CIGI 41-5
attaching observer to 16-2, 16-6
body frame coordinate system 15-7
clamping to ground 21-2
count 19-15
display settings 3-6
element definition 33-2
embarked 19-15
filtering 19-16, 19-17
attachment list 16-5
ghosted 20-5
ground clamping 21-3

entity (continued)
icon
2D 19-14
adding new image 35-9
changing size 19-12
inset view 17-2
instrument panel, configuring 17-7
intervisibility 22-4
label 19-7
2D 19-4, 19-7
3D 19-3
color 19-3
enabling and disabling 19-6
pinning to display 19-6
level body frame coordinate system 15-7
mapping sound effects to 23-4
model 30-4
model state 41-7
changing in host emulator 42-6
name 19-3
ownship, creating 42-5
primary 16-9, 16-12
secondary 16-9, 16-12
See also simulation object and unit
selecting 19-2
shadows 13-16
sound effects 23-2
surface
buoyancy 21-24
configuring wake 35-14
trailing effects 21-6
type 30-5
enumeration 19-3, 34-4
wake, enabling and disabling 21-23
entity component control 41-7
message 41-6
Entity Components Mappings page 41-7, 41-8
Entity Control 41-3
Entity Definition Editor page 35-2, 35-3, 35-14, 36-7, 38-5
Entity Display Settings page 19-5, 19-6, 19-9, 19-10, 19-12, 19-13, 19-16, 21-2, 21-3, 21-5, 21-14, 21-23, 21-25
Entity Image Symbol, visual definition 35-2, 35-3, 35-9
Entity Mapping Settings page 34-4, 34-5, 34-7, 38-8
Entity Sound Editor page 23-5, 23-6, 23-7
enumeration 34-4
document 30-5
environment
disabling 11-22
loading configuration 3-6
environment condition 11-5
adding 11-7
default 11-7
deleting 11-8
editing 11-8
marine, setting 11-14
Environment Condition Settings page 11-14
Environment Conditions Settings page 6-21, 11-6, 11-7, 11-8, 11-9, 11-10, 11-11, 35-16
Environment Settings page 11-22
environment variable
GDAL_CACHEMAX 29-8
MAK_GEOID_GRID 25-6
MAKLMGRD_LICENSE_FILE 2-7
OpenSceneGraph, anaglyphic stereo 6-29
OSG_CURL_PROXY 26-5
OSG_CURL_PROXYPORT 26-5
OSG_ICO 5-11
OSG_MAX_PAGEDLOD 29-8
OSGEARTH_CACHE_PATH 29-2
OSGEARTH_CURL_PROXYAUTH 26-5
RTI_CONFIG 2-9
exaggerating, terrain scale 18-5
exercise ID, DIS 4-6
exiting
full screen window 45-4
VR-Vantage 3-9
expanded label 19-3
pinning to display 19-6
expanding
unit 20-2, 20-4
all 20-4
Experimental Features dialog box 11-17, 15-21, 15-22
explosion 21-12
exporting
element definition 33-16
element definition to elem file 33-16
hierarchy 33-16
leaf file 33-16
model definition 32-13

exporting (continued)
settings 3-15
extended imagery dataset, CDB 28-15
extended imagery sub-dataset, CDB 28-16
external reference 35-27
Extract Ocean page 25-16
Extract Props page 25-21, 30-9
extracting
prop
from terrain 25-4
from terrain patch 25-21
water texture 25-16
extruded, building 26-18
eyepoint. *See* observer

F

-F command-line option 3-6
-f command-line option 27-4
face front culling 21-25
factory settings, restoring 3-15
fan, intervisibility 22-4
far clipping plane 6-14
FDD file 2-9
feature
areal 26-15
data
buoys and beacons 26-20
displaying 24-2
processing 26-10
extracting as prop 25-21
layer, adding 25-4
SwitchStateMapper 35-22
UvAtlasMapper 35-22
vector, list of 24-4
VR-Vantage 1-2
feature layer 26-20
adding 25-14
adding prop from 25-23
Feature Settings page 24-5
Feature_Data_Dictionary.xml 28-17
featureDataBroker 26-10
FED file 2-9
FED File Name, HLA 4-5
Federate Name, HLA 4-5
Federate Type, HLA 4-5
federation execution
name, specifying 4-5
Federation Name, HLA 4-5
Feet Above Ground Level 35-13
Feet Above Mean Sea Level 35-13
FeetAGL updater 35-13
FeetMSL updater 35-13
FeetPerMinute updater 35-13
field of view 6-19, 6-20, 15-19, 15-20, 36-3
changing 6-19
file
cache, clearing 29-5
caching 29-2
compressed 35-28
default_ModelPreload.mpsx 5-6
dtxc 35-23
FED 2-9
geometry 35-21
RID 2-9
File Caching Settings page 5-7, 5-8, 29-2, 29-5
Filename parameter 35-12
filename_shp_map.txt configuration file 25-14
filtering
element definition list 34-7
function list 46-7
model definition list 32-8
object attachment list 16-5
simulation objects
by observer altitude 19-17
in scene 19-16
fire
effects 21-11
event
mapping sound effects to 23-4
sound effects 23-2
line 21-12
Fire Sound Editor page 23-9
firing, effects 34-2
Fixed Near Clip attribute 6-15
Fixed Near Clip When Attached, attribute 6-15
flag, movement in wind 35-20
flaming effects 21-11, 34-2
flat earth, coordinate system 15-7
FLEXlm 2-4, 2-5
flipping, DDS textures 25-8, 35-23
FLIR 12-2
floating, panel 45-2
FLT file 25-3, 30-4
mapping to models 30-5
focal depth 21-20
focal range 21-20

focus, scene 21-20
fog 11-5
  denseness of 11-10
  height and color 11-10
Follow mode 15-8, 16-6, 16-7
FOM 1-3, 2-8
FOM Mapper
  specifying
    initialization data 4-5
    library 4-5
FOM Modules, HLA 4-5
footprint 21-6
force 19-3
  color, 2D icon 19-8
format
  component control message 41-6
  OpenFlight 39-2
frame rate 5-2, 5-8
  setting 3-6
  virtual reality 9-5
framebuffer 8-8
free-fly mode 16-7
frequency
  color of 36-2
  radio communication line 21-18
frustum 6-12, 6-26
  editing 6-19
Full Color Mode 12-5
full screen 3-8
  mode 3-6
  window 6-4, 45-4
function
  list, filtering 46-7
  movement 46-2
  profiler 5-5

G

-G command-line option 3-6
-g command-line option 27-5
gain 12-2
  changing 15-27
  sensor effects 12-4
GDAL, configuring cache 29-8
GDAL_CACHEMAX, environment variable 29-8
GDB terrain 25-3
  configuring soil type 24-8
  shapefile 25-4, 25-14
General Application Settings page 6-31
geocentric
  coordinate system 15-7
  terrain 26-2
geographic DDM 19-17
geoid grid file 25-6
geometry 35-27
  file 35-21
  sharing 5-7
Geometry Grid Dataset 27-3, 27-7
geospecific airport lightpoint sub-dataset, CDB 28-9
geospecific dataset 28-6
geospecific environmental lightpoint sub-dataset, CDB 28-10
geospecific sub-dataset, CDB 28-8
GeoTIFF 25-3
  image 25-11
geotypical dataset, CDB 28-13
geotypical sub-dataset, CDB 28-15
GGDS 27-7
ghosted
  icon, by echelon level 20-6
  unit 20-5
ghosted units 20-5
GIF 25-3
gimbaled sensor, field of view 17-9
GL Studio 1-8, 17-3, 35-10
  instrument panel 17-6, 17-7
  license 1-9
GLImagesink Test 8-9
global setting 3-17
Global Terrain Surface Control, CIGI 41-10
Global Terrain Surface Control message 41-10
God ray 13-11
graphic
  buffer, depth and stencil 5-8
  depth 3-5
  fire and detonation 21-12
  mode, wireframe 24-13
  object, showing and hiding 40-2, 40-4
grid line
  configuring 24-7
  displaying 24-6
  keyword 6-22
Grid Lines Settings page 24-7
GridLines keyword 6-22
ground clamping 21-2
  configuring 21-3

Ground Clamping Cutoff Distance Scale 21-3
GStreamer 8-2, 8-9
  configuring for VLC Out 8-7
GUI, localizing 2-10
GUI Layout page 45-4
guidelines 35-27
  creating models 35-27

H

-h command-line option 3-6, 27-5
H.264 8-2
  configuring 8-7
HAT, *See* height-above-terrain line
HAT/HOT Request 41-3
HAT/HOT Response 41-3
HDR 13-14
head mounted display
  HMD, virtual reality 9-2
header file
  DtCigiDiGuyCtrl.h 3, 9
  DtLightSignalSequenceGenerator.h 29
heading 15-3, 19-3
  indicator 19-4
  intervisibility line 22-11
  observer 15-4
  rotating icon to 19-13
  specifying measurement unit 45-6
headlight 13-5
Headlight updater 35-13
height
  fog 11-10
  ocean 25-17
height map
  ocean
    regenerating 25-20
    resolution 25-18, 25-19
    texture size 25-18, 25-19
height-above-terrain line 15-25, 19-10, 40-2
  displaying 40-5
helicopter
  rotor wash, high quality 11-21
hiding
  3D models 19-15
  intervisibility lines 22-10
  objects 40-2, 40-4
  occluded icon 19-14
  panel 45-3
hiding (continued)
  symbol decorations 19-8
  toolbar 45-3
  vertex labels 40-5
hierarchy
  element definition 33-2
  exporting 33-16, 33-17
  importing 33-14
hierarchy file, importing 33-14
high dynamic range 13-14
High Dynamic Range Settings page 13-10, 13-14
HLA 19-16, 19-17
  advisory message, enabling and disabling 4-6
  Application Number 4-5
  connecting to 3-11
  connection 4-3
    configuring 4-4
  FED File Name 4-5
  Federate Name 4-5
  Federate Type 4-5
  Federation Name 4-5
  FOM Modules 4-5
  Listen for View Controls 4-6
  Override Global View Control Setting 4-6
  Publish Observers 4-6
  Site Number 4-5
  versions supported 1-3
HMD
  enabling 9-3
  enabling and disabling 9-4
horizontal navigation 46-2
host
  CIGI, configurations 41-13
host emulator 41-2
  CIGI 42-2
    connecting to 42-4
Host IP Address parameter 4-8
hostname, license server 2-6
HTC Vive 9-2
HUD. *See* cockpit display

I

-I command-line option 3-6
icon
  2525B 35-2

icon (continued)
2D 35-3
bitmap 35-3
displaying label 19-7
editing 35-2
occluded 19-14
aggregate, displaying 20-2
bitmap, adding new 35-9
borders 19-11
common 45-5
configuring 2D 35-2
heading, rotating to 19-13
image 35-3
adding new 35-9
level of nesting 20-5
settings 3-15
size, changing 19-12
IG Control 41-3
image
adding 35-9
attributes 25-10
compressing 35-28
display order 25-13
displaying 25-9
format, supported 25-3
GeoTIFF 25-11
icon 35-3
texture coordinate system 25-11
imagery dataset, CDB 28-5
imagery lightmap sub-dataset, CDB 28-6
imagery sub-dataset, CDB 28-5
imagesymbolmap attribute 35-6, 35-9
ImGui'overlay, debug 5-13
implicit DDM, configuring 4-6
Implicit Geographic DDM 19-17
importing
element definition 33-14, 33-15
hierarchy 33-14
hierarchy file 33-14
leaf file 33-15
leaf files 33-14
model definition 32-12
settings 3-15
including XML files in earth files 26-8
increasing speed of navigation 15-25
incremental compiling 5-11
independent setting, global and observer 3-17
indicator 19-6
heading 19-4
indicator 19-6 (continued)
squawk 21-15
information, entity label 19-3
infrared 12-2
inheritance, element definition 33-2
initialization data, FOM Mapper 4-5
inserting, altitude point 24-10
inset view 17-2
installing, RTI 2-9
instancing 3-6
DI-Guy 5-12
instrument panel 17-6
channel keyword 6-22
configuring 17-7
GL Studio 17-6, 17-7
instrumentPanelCategory parameter 17-7
intensity, rain and snow 11-11
interaction, signal 21-15
interest management 19-16, 19-17
configuring 19-17
Interest Management Settings page 19-18
intermediate contour line 24-11
intersection, KD trees 3-6
intersector line, terrain 29-15
Intersector Settings page 29-15
interval, contour line 24-12
intervisibility 22-2
enabling and disabling 22-10
entity 22-3, 22-4
fan 22-3, 22-4
moving 22-8
grid 22-3
line
hiding 22-10
information 22-11
object
configuring 22-9
deleting 22-12
displaying 22-7
editing 22-8
transient and permanent 22-5
pinning object to simulation object 22-10
point-to-point 22-3
types 22-3
Intervisibility Settings page 22-6, 22-7, 22-9, 22-10, 22-11
invalid model definition 32-11
IP address
for CIGI connections 42-3

IP address (continued)
for outgoing data 4-6
IVE 29-8

J

jitter, camera 15-23
JPEG 25-3
JPG 25-3
JRM Technologies 12-2
jumping, observer 15-17

K

KD tree 3-6
key map 46-2
creating 46-7
deleting 46-7
editing 46-3
key mapping 15-11
adding 46-3
binary 46-3
changing 46-5
combining 46-6
deleting 46-5
Key Mapping Editor 46-2
adding mapping 46-3
binary mapping 46-3
changing mapping 46-5
deleting mapping 46-5
filtering function list 46-7
Key Mapping Editor page 46-3, 46-7
keyboard 46-2
attaching to objects from 16-3
changing speed 15-27
control 45-4
inset views 17-2
mappings 15-6, 15-10, 46-2
movement functions 15-6
moving observer 15-10
navigation 46-2
keyword
channel 6-12, 6-22
cigiView1 6-22
cockpitOnly 6-22
GridLines 6-22
largeCompass 6-22, 15-5
NoExternalInput 6-22
showCockpits 6-22
showRemoteGraphics 6-22
keyword (continued)
smallCompass 6-22, 15-5
KilometersPerHour updater 35-13
KnotsPerHour updater 35-13

L

-L command-line option 3-7
-l command-line option 3-6, 27-5
label
2D
background 19-9
displaying 19-7
showing and hiding 19-8
entity 19-7
2D 19-4
3D 19-3
enabling and disabling 19-6
object 19-7, 40-5
pinning to display 19-6
simulation object 19-7
tactical graphics 40-5
Lambert Conformal Conic, coordinate system 15-7
lamppost, prop type 25-27
language, specifying 3-6
largeCompass, keyword 15-5
largeCompass keyword 6-22
laser
designator, displaying 21-26
latency
video stream, reducing 8-10
latitude 19-3
layer
earth file, debugging 29-9
feature 26-20
adding 25-4, 25-14
adding prop from 25-23
name 26-12
shapefile 26-12
ocean 25-15
adding 25-17
LCC, coordinate system 15-7
leaf, exporting 33-17
leaf file
exporting 33-16
importing 33-15
leaf files, importing 33-14
length
intervisibility object 22-8

length (continued)
track history 21-9
LettersNumbers.ctxc 26-24
level observer 15-7
level-of-detail 15-19, 15-20, 35-24
See also LOD
library
CIGI class 41-2
specifying for FOM Mapper 4-5
license
3D models 1-10
DI-Guy 1-7
GL Studio 1-9
OpenSceneGraph 1-11
osgEarth 1-11
SilverLining 1-8
SpeedTree 1-10
License Manager 2-4, 2-5
license server, hostname 2-6
License Setup dialog box 2-6
lifetime
radio communication line 21-17
squawk indicator and communication line 21-17
light
by time of day 11-3
daylight control 13-6
navigation 26-26
point 13-6
source 13-6
light characteristic attribute 26-29
light point
CDB
properties'CDB
lightpoint, properties 28-11
light point group
color, CIGI 41-10
Light_DefaultValues_3_0.xml 28-18
Light_DefaultValues_3_2.xml 28-17
light_layer_list.csv 26-26
lighting
disabling 11-22
dynamic 13-5
effects 1-6, 29-10
high dynamic range 13-14
Lighting Render Settings page 13-2, 13-3, 13-4
lightmap, CDB 28-6
lightpoint, CDB 28-9, 28-10
Lights_xxx.xml 28-18
limitation, terrain 25-5
line
fire 21-12
height-above-terrain 19-10, 40-2
intervisibility, information 22-11
radio communication 21-15
Line of Sight Response 41-3
Line of Sight Segment Request 41-3
Line of Sight Vector Request 41-3
linear feature, list of 24-4
line-of-sight
See also intervisibility
communication line 21-17
Linux
display 6-32
starting VR-Vantage 3-4
list
props 25-25
vector feature 24-4
Listen for View Controls
DIS 4-7
HLA 4-6
Listen Port parameter 4-8
LITCHR attribute 26-29
Little Pond terrain 23-2
Loader Settings page 29-7
loading
display engine configuration 3-6, 6-9, 7-8
DTED 25-6
entity display settings 3-6
environment configuration 3-6
models 29-2
multichannel configuration 6-27
multiple DTED files 26-9
object type mapping 3-6
observer views 16-17
plug-ins 3-8
preventing 3-9
saved views 15-32
scene from command line 3-9
settings 3-15
terrain 3-2, 3-10, 29-2
from command line 3-9
local, terrain 3-10
Local Settings Designator 4-5
locale, specifying 3-6
localization, specifying 3-6
localizing, GUI 2-10

location
  compass 15-5
  jumping to 15-17
  observer 15-3
  window 6-11
location image attribute 25-11
LOD 15-19, 15-20, 35-24, 35-27
  configuring 29-8
  ocean elevation 25-18
  paged, scale factor 15-21
  scale, setting 13-8
  terrain, zooming in 15-22
LOD distance, buoyancy 21-24
log file, specifying 3-6
logger.py 43-3
longitude 19-3
LWO file 25-3

M

-m command-line option 27-5
magnification, performance 15-22
MAK CIGI Mappings 41-1
MAK Encrypted Data Format 27-7
MAK RTI, configuring 2-9
MAK_GEOID_GRID, environment variable 25-6
Makland terrain 23-3
MAKLMGRD_LICENSE_FILE, environment variable 2-7
manual, conventions xxix
map
  cube, recalculating 13-13
  texture 1-6, 29-10
mapping
  CIGI control message 41-6
  database, CIGI 41-11
  keyboard 15-11
  keys 46-2
  model
    CIGI 41-4
    to object type 30-5, 34-4
    to remote observer 14-11
  movement to keyboard 15-10
  object type 34-4
  shapefiles to features 25-14
  sound effects 23-4
  sound file to entity 23-5
marine
  conditions, setting 11-14
marine (continued)
  effects, enabling and disabling 11-13
marker-scale attribute 35-26
marking text 19-3
mask
  adding to earth file 26-16
  element 26-16
material, ambient value 13-3
MEDF 27-7
  creating 35-28
  file 25-3
medfTool 27-7
MEIF, creating 35-28
memory 5-2
  improving use ofc 5-7
menu
  context-sensitive 45-5
  Settings 3-15
merging, model definition 32-12
message
  Global Terrain Surface Control 41-10
  notification level 3-8
  outgoing 4-6
  state update 14-10
  view control 15-32
metafile, texture skinning 35-21, 35-22
metaFileName parameter 35-21, 35-22, 35-23
MetaFlight
  building efficient terrains 27-2
  processing 27-3
mftTool 27-4, 27-6
middle-click menu 45-5
MilesPerHour updater 35-13
Military Symbol Icon, visual definition 35-2
MIL-STD 2525B color 19-9
  unit 20-10
Mimic Location Track mode 16-9
Mimic mode 15-8, 16-6, 16-8
Mimic Track mode 16-9
mirror image attribute 25-11
MISB 8-2
Mixed Reality Headset 9-2
mode
  Absolute 16-7
  Follow 16-7
  Mimic 16-8
  Mimic Track 16-9
  observer 14-2
  Space Follow 16-10

mode (continued)
speed scaling 15-26
Tether 16-11
Tether Track 16-12
Track 16-13
view 15-8, 16-6
model
3D 30-4
hiding 19-15
scaling 18-4
adding, tutorial 38-2
animation 39-11
compressing 35-28
creating 35-27
decal effects 21-8
definition, buoy 26-21
DI-Guy, choosing 34-10
editing model mapping 34-5
element 26-10, 26-11, 35-26
file caching 29-2
flipping DDS textures 35-23
instance, caching 5-7
instancing 3-6
clearing cache 5-7
mapping
adding 34-4
CIGI 41-4
common buttons 45-5
deleting 34-7
editing 34-5
to FLT files 30-5
to remote observer 14-11
name 19-3
preloading 5-6
schema 32-2
set 33-7
changing 19-11
sharing resources 5-7
state
changing in host emulator 42-6
CIGI 41-7
stealth 19-16
texture 35-21
texture atlas 35-21
trailing effects 21-8
types 30-4
vehicle 1-4
model definition 30-5
applying updated 32-8
model definition 30-5 (continued)
buoy 26-20, 26-21
building from attributes 26-21
texture atlas 26-24
compass 15-5
copying 32-7
creating 32-2, 32-5, 37-2, 38-2
cockpit display 35-11
DefaultEllipsoidArc 36-2
deleting 32-6
DI-Guy character 32-5
editing 32-8
exporting 32-13
filtering list of 32-8
importing 32-12
importing from command line 32-12
invalid 32-11
merging 32-12
parameter
adding 32-8
deleting 32-11
editing 32-10
prop, changing 25-27, 30-11
saving 32-13
schema 31-2
SpeedTree 32-5
texture atlasing 35-21
wake 11-16
Model Definition Editor page 32-2, 32-5, 32-6, 32-7, 32-8, 32-10, 32-11, 32-12, 32-13, 35-9, 35-21, 35-23, 36-5, 36-6, 36-7, 37-2, 38-2
Model Preload Configuration page 5-6
Model Specification for DIS Interoperability 39-2
model state component control message 41-7
model state control message 41-6
Model_PagingDistance.xml 28-17
modeldefinitionschema, attribute 35-6
module, pyCigi 43-5
monitor, multiple 6-24
monitoring traffic 4-7
moonlight 11-3
Motion Imagery Standards Board 8-2
mouse
changing speed 15-27
drag terrain 15-14
drag view 15-16
moving observer 15-14, 15-16, 15-19

mouse (continued)
orbiting terrain 15-16, 15-19
teleporting observer 15-17
Mouse Mappings page 15-14, 15-19
movement, speed scaling 15-25
moving
camera 15-13
intervisibility fan 22-8
observer 15-3, 15-12
mouse with 15-14, 15-16, 15-19
options 15-12
MPEG 8-9
MPEG-2 8-2, 8-9
MTD file 25-3
MTL file 25-3
MTP file 25-3
multicast
address, DIS 4-7
groups 4-7
multicast TTL, DIS 4-7
multichannel
configuration, saving 6-27
configuring 6-24
MultiGen-Paradigm, OpenFlight format 39-2
multiple, parameter values 33-10
munition, display of path and detonation 21-12
muzzle flashes 21-11

N

-n command-line option 3-8
name
displaying 19-4
entity 19-3
layer 26-12
simulation object 19-3
navigating, view 15-3
navigation
attached context 15-8
function, editing key map 46-3
key map 46-2
keyboard 46-2
light 26-26
streaming 26-26
options to 15-12
speed scaling 15-25
navigational aid light 26-29
near clipping plane 6-14
Near Far Clip Policy, attribute 6-15
netsocket 4-7
network
connecting to 3-11, 3-13, 3-14
automatically 3-5
disconnecting from 3-14
network card 4-6
Network Device Address parameter 4-8
new, terrain 25-5
night vision 12-2
node, switch 35-20, 35-22
NoExternalInput keyword 6-22
noise 12-4
normal map 1-6, 29-10
north, orienting Plan View 15-24
notification level, specifying 3-8
NVG 12-2

O

-O command-line option 3-8
oberver, camera shake 15-23
OBJ 30-4
file 25-3
object
attaching observer to 16-2
count 5-2, 19-15
detaching observer from 16-5
intervisibility 22-5
pinning to simulation object 22-10
label 19-7, 40-5
model 30-4
orbiting 15-13
showing and hiding 40-4
object type
mapping 34-4
loading 3-6
mapping to element definition 30-5
Objects List Panel 16-2, 19-2
aggregates, listing 20-2
filtering 20-2
props, selecting 25-25
OBJNAM, attribute 26-23
observer 15-3
adding 14-6, 14-7
altitude, filtering simulation objects by 19-17

observer 15-3 (continued)
attaching
articulated part 16-4
to entity at startup 16-15
to simulation object or prop 16-2
to simulation objects 16-6
tutorial 16-16
attachment
in 2D 16-14
order 16-3
binding to view 16-17
camera 15-13
changing 6-26
CIGI views 41-12
compass 15-4
coordinate system
2D 15-7
3D 15-7
detaching from object 16-5
dragging
terrain 15-14
view 15-16
heading 15-4
inset view, in 17-2
location and orientation 15-3
LOD, ocean 25-18
mode 14-2
changing 14-3
creating 14-4
deleting 14-5
editing 14-5
mapping to CIGI view 41-12
Plan View 1-5
renaming 14-5
setting 3-17
switching projection 14-2
movement 16-6
functions 15-6, 46-2
speed 15-27
moving
keyboard with 15-10
mouse with 15-14, 15-19
multiple, loading 16-17
orbiting 15-13
orientation 15-16, 15-24
publishing 4-6, 4-7
rain and wave effects 11-18
range for sound effects 23-3
remote 14-11
removing 14-9
observer 15-3 (continued)
saving view 15-28
teleporting 15-17
view
attachment change 16-3
saving 15-28
transitioning between 15-29
visibility 11-5
to other applications 14-10
ways to move 15-12
zooming 15-19
2D 15-19
3D 15-20
Observer Control Panel 15-25
Observer Information Panel 15-3
Observer Name 6-12
Observer Settings page 11-13, 11-19, 12-4, 13-6, 14-3, 14-4, 14-5, 14-6, 14-9, 15-5, 15-15, 15-23, 15-24, 15-26, 15-27, 17-9, 18-4, 19-15, 21-7, 21-8, 21-11, 21-13, 21-21, 24-2, 24-6
Observer Settings Toolbar 18-4, 19-11
Observer Views Panel 15-28
loading saved views 15-32
saving views 15-31
occluded icon, hiding 19-14
ocean 1-9
choppiness 11-15
dynamic, enabling and disabling 11-13
effect 11-12
height 25-17
height map
regenerating 25-20
resolution 25-18
texture size 25-18, 25-19
layer 25-15
adding 25-16, 25-17
planar reflection 13-7
configuring 13-8
performance 13-8
SpeedTrees 13-8
texture height 13-8
texture width 13-8
quality 11-17
swell 11-14
synchronized 3-8
Ocean Height Map Regeneration Threshold 25-20
ocean height map resolution 25-19
Ocean LOD Elevation 6-21, 25-18

Ocean Settings page 11-16, 11-17, 11-20, 11-21, 13-8, 13-13, 25-19, 25-20
Oculus Rift 9-2
Oculus Rift HMD--enableOculus 3-6
offset, UV 35-23
OffsetX parameter 35-12
OffsetY parameter 35-12
OffsetZ parameter 35-12
OGR, driver 26-12
opacity
  2D label background 19-9
  prop 25-26
    setting 25-26
OpenAL 23-3
OpenFlight 30-4
  file 25-3
  format 1-4, 39-2
OpenGL 8-9, 29-6
opening
  inset view 17-2
  scene 11-2
  terrain 3-2, 3-10
OpenSceneGraph 1-2, 1-8
  environment variables for anaglyphic stereo 6-29
  license 1-11
optimizing, performance 5-2
option
  command-line 3-5
  element 26-19
  navigation 15-12
orbit, observer 15-13
orbiting
  terrain, with mouse 15-16, 15-19
orientation 15-3
  camera 6-26
  clamping 21-3
  displaying 19-4
  icon 19-13
  observer 15-3, 15-14, 15-16, 15-19, 16-6
  Plan View, north 15-24
  prop, changing 25-27
  specifying measurement unit 45-6
OSG file 25-3
OSG_CURL_PROXY, environment variable 26-5
OSG_CURL_PROXYPORT, environment variable 26-5
OSG_ICO, environment variable 5-11
OSG_MAX_PAGEDLOD, environment variable 29-8
osgEarth 1-8, 26-2, 29-9
  adding terrain servers 26-5
  CDB 28-2
  driver 26-9
  license 1-11
osgEarth Boundary Generation Tool 26-16, 26-17
osgEarth Cache Tool 29-3
osgEarth Profile dialog box 29-9
osgearth_cache 29-4
osgearth_cache tool 29-3
OSGEARTH_CACHE_PATH, environment variable 29-2
OSGEARTH_CURL_PROXYAUTH, environment variable 26-5
outgoing PDU 4-6
outline
  emitter volume segment 36-6
  segment 36-6
Out-the-Window, observer mode 3-17
out-the-window view 6-24
overlay
  developer debug 28-9
  hiding graphics on 40-4
Overlays tab 40-4
Override Global View Control Setting
  DIS 4-7
  HLA 4-6
ownship 41-13
  entity, creating 42-5

P

-p command-line option 27-5
packet
  CIGI 41-3
    user-defined 41-3
Packet Buffer Size parameter 4-8
page
  Audio Settings 23-2, 23-3
  CIGI Global Terrain Surface Components 41-10
  CIGI Light Point Groups 41-10
  CIGI Observer Mode Mapping 41-12
  Connection Status 3-14
  Detonation Sound Editor 23-9
  DI-Guy Settings 5-12
  DIS Connections 14-11

page (continued)
Display Units Settings 19-10, 40-5
Edit Existing Props 25-22
Elevation Color 24-9, 24-10, 24-11
Entity Definition Editor 35-2, 35-3, 35-14, 36-7, 38-5
Entity Mapping Settings 38-8
Entity Sound Editor 23-5, 23-6, 23-7
Environment Conditions Settings 6-21, 11-6, 11-7, 11-8, 11-9, 11-10, 11-11, 35-16
Feature Settings 24-5
File Caching Settings 5-8
Fire Sound Editor 23-9
General Application Settings 6-31
High Dynamic Range Settings 13-10, 13-14
Intersector Settings 29-15
Lighting Render Settings 13-2
Model Definition Editor 32-12, 35-9, 35-23, 36-5, 36-6, 36-7
Model Preload Configuration 5-6
Mouse Mappings 15-19
Observer Settings 12-4, 14-6, 15-26
Raster Maps 29-11
Render Settings 11-21, 21-22
Schema Editor 32-5
Sensor Settings 12-4, 12-5
Shadow Settings 13-17, 13-18
Symbol Decoration Settings 19-9
Terrain Contents 25-14, 25-17, 29-3
Video Streams 8-3, 8-4
Video Streams Configuration 8-5, 8-13
Virtual Reality Settings 9-4
Paged LOD Magnification LOD Angle 15-22
Paged LOD Magnification Scale Factor 15-21
paged terrain
flipping DDS textures 29-7
preprocessing 29-8
panel
displaying and hiding 45-3
dockable 45-2
docking 45-2
floating 45-2
undocking 45-2
parameter
adding, to model definition 32-8
angleSegmentMax 36-5
parameter (continued)
ChannelKeyword 35-12
CIGI Entity Type Mapping 4-8
CIGI Messages 4-8
CIGI Version 4-8
configName 35-21, 35-23
deleting, from model definition 32-11
Driver Frequency 4-8
Driver Mode 4-8
Filename 35-12
Host IP Address 4-8
instrumentPanelCategory 17-7
Listen Port 4-8
metaFileName 35-21, 35-22, 35-23
model definition, editing 32-8, 32-10
Network Device Address 4-8
OffsetX 35-12
OffsetY 35-12
OffsetZ 35-12
Packet Buffer Size 4-8
Perspective 35-12
PerspectiveHeight 35-12
PerspectiveWidth 35-12
RotDir 35-20
RotOffsetDeg 35-20
RSOName 35-12
schema
adding 31-5
deleting 31-6
editing 31-6
Send Port 4-8
showLines 36-6
Transport Type 4-8
visual definition, editing 33-10
wakeLength 11-16
WindSpeed 35-20
WindState 35-20
parent
element 33-8
element definition 33-2
parent, attribute 35-5
particle effect, sea spray 11-20
patch, terrain 25-3
path, specifying for RTI 2-9
pattern, color 26-21
PDU
sending 14-10
view control 15-32

performance 5-8, 15-20, 19-16, 19-17, 25-18
  anti-aliasing 3-5
  attached observer 16-6
  coloring draw calls 5-11
  compressing texture 29-5
  cube map calculation 13-13
  DI-Guy 5-12
  function profiler 5-5
  graphics, depth 3-5
  HAT lines 19-10, 40-5
  improving 5-7
  instancing 3-6
  interest management 19-16
  model, best practices 35-27
  ocean planar reflections 13-8
  optimizing 5-2
  preloading clouds 11-11
  preloading models 5-6
  setting application priority 3-6
  shadows 13-18
  statistics
    displaying 5-3
    enabling in CIGI 41-9
  stencil bits 3-8
  streamed data 5-11
  terrain 25-5
  virtual reality 9-5
  visual 5-8
  zooming in 15-22
permanent intervisibility object 22-5
perspective, in display 6-19
Perspective parameter 35-12
PerspectiveHeight parameter 35-12
PerspectiveWidth parameter 35-12
pipeline, video stream 8-4
pitch 15-3
Pitch updater 35-13
pixel 35-27
Plan View
  observer mode 1-5
  orient north 15-24
planar reflection 13-7
  SpeedTrees 13-8
plug-in
  loading 3-8
  prevent loading 3-9
  Scalable Display Technologies 10-2, 10-10
PNG 25-3
point
  feature, list of 24-4
  orbiting around 15-13
point-to-point intervisibility 22-3
polarized stereo 6-28
  configuring 6-30
policy
  Distance To Attached 6-15
  Reduce Near Clip 6-15
  Reduce Z Fighting 6-15
polygon 35-27
  count 25-5
  not going through 15-25
popup menu 45-5
port
  CIGI connection 42-4
  DIS 4-6
  proxy 26-5
position 15-3
  camera 6-26
  observer 15-14, 15-19, 16-6
  prop, changing 25-27
  window attribute 6-11
precipitation 11-5
  intensity 11-11
  remote 11-22
  type 11-11
preloading
  cloud 11-11
  model 5-6
preprocessing
  terrain, paged 29-8
primary color, unit 20-10
primary simulation object 16-9, 16-12
processing
  MetaFlight 27-3
  streamed data 26-10
product
  features 1-3
  technical support xxviii
  upgrades xxviii
profiler, function 5-5
projected terrain database 15-7
projection, switching 2D to 3D 14-2
Projection Resize Policy 6-12
  configuring 6-16
Projection Units 6-12
projector, multiple 10-2

prop
  adding
    from feature layer 25-23
    to terrain 25-21
  attaching observer to 16-2, 16-5
  configuring 25-4
  extracting 25-21
    from terrain 25-4
  filtering attachment list 16-5
  model definition, changing 25-27, 30-11
  opacity 25-26
  opaque 25-26
  selecting 25-25
  transparent 25-26
  type 25-27
  viewing list of 25-25
property
  channel 6-12
  window 6-11
proxy
  connecting through 26-5
  port 26-5
Publish Observers
  DIS 4-7
  HLA 4-6
publishing, observer 4-6, 4-7
pulse repetition rate 36-2
pyCigi module 43-5

Q

Qt Linguist 2-10
quality, ocean 11-17

R

-r command-line option 27-5
radio
  communication line 21-15
    color 21-18
    enabling and disabling 21-16
    lifetime 21-17
Radio Communications Settings page 21-16, 21-17, 21-18
radius, emitter volume 36-3
rain 11-5
  intensity 11-11
  remote 11-22
  splash effect 11-18
range
  focal 21-20
  ground clamping 21-3
  intervisibility line 22-11
raster driver, CDB 28-3
raster map
  display order 25-13
  displaying 25-9
Raster Maps page 25-9, 25-13, 29-11
Rate Control 41-3
ray
  crepuscular 13-11
  intervisibility 22-4
  sun 13-11
Real DB 1-10
recalling
  view 15-29
  view state 15-28
receive buffer size 4-7
  DIS 4-7
Reduce Near Clip, policy 6-15
Reduce Z Fighting, policy 6-15
Reduce Z Fighting Ratio, attribute 6-15
reflection
  map 1-6, 29-10
  planar 13-7
refresh rate 5-8
regenerating, ocean height map 25-20
region
  DDM 19-17
  update threshold 19-17
registrant_feature_list.csv 26-10
reloading
  scene 6-4
  terrain 3-11
remote
  control 1-6
    of the observer 15-32
  display engine, loading configurations 6-9, 7-8
  observers 14-11
  stealth, displaying model 19-16
  weather 11-22
remote graphics, keyword 6-22
removing
  channel 6-8
  observer 14-9
  view 15-30
  window 6-8

renaming
observer mode 14-5
simulation connection 4-4
Render Settings page 5-4, 5-5, 5-10, 5-11, 6-17, 11-21, 13-12, 18-6, 18-7, 18-8, 19-14, 21-22, 24-14, 28-9, 28-12, 29-14
repeat image attribute 25-11
replace image attribute 25-10
replacing, saved view list 15-32
replicating, settings and data 3-17, 6-3
resizeable window 6-4
resizing, windows 6-16
resolution
ocean height map 25-19
tree 35-24
restoring, view 15-29
result
detonation, sound mapping 23-8
reusable software object 35-10
revolving, observer 15-13
RGB 25-3
RID file 2-9
rid.mtl file 2-9
right-click menu 45-5
roll 15-3
Roll updater 35-13
rotation path, smoothing 21-4
RotDir parameter 35-20
RotOffsetDeg parameter 35-20
rotor wash 11-18
high quality 11-21
route
altitude, displaying 40-5
RPR FOM 1-3, 30-5
version, specifying 4-5
RSO 35-10
RSOName parameter 35-12
RTI 2-8
definition of 2-8
RTI_CONFIG environment variable 2-9
running, Boundary Generation Tool 26-17

S

-S command-line option 3-8, 16-6
-s command-line option 3-8, 27-5
S-57
buoys and beacons 26-20
signal sequence 26-29
s57_attr_map.csv 26-22, 26-25
s57_defaults.csv 26-25
s57_layer_list.csv 26-20, 26-25
s57_modeldef_by_name_map.csv 26-23
sampling rate
intervisibility, configuring 22-9
San Luis Obispo terrain 23-3
saving
display engine configuration 6-8
element definition 33-13
hierarchy and leaves 33-17
model definition 32-13
multichannel configuration 6-27
observer view, individual 15-31
scene 11-2
schemas 31-2
screen capture 6-31
settings 3-15
terrain 25-6
view 15-28, 15-29
to file 15-31
Scalable Display Manager 10-2
Scalable Display Technologies plug-in 10-2, 10-10
scale
factor, material ambient and diffuse value 13-3
SpeedTree, random 35-25
Scale Factor slider 18-5
Scaled Speed check box 15-26
scaling
model, 3D 18-4
terrain 18-5, 18-6, 18-7
scene
creating 11-2
definition 11-2
loading from command line 3-9
opening 11-2
reloading 6-4
saving 11-2
Scene Settings dialog box 11-6, 11-7, 11-8, 11-9, 11-10, 11-11
schema 31-2
copying 31-4
creating 31-3
deleting 31-4
model definition 31-2

schema 31-2 (continued)
parameter
adding 31-5
deleting 31-6
editing 31-6
saving 31-2
visual definition 31-2
Schema Editor page 31-3, 31-4, 31-5, 31-6, 32-5
screen capture
saving 6-31
setting save location 6-31
screen number, setting 3-7
Screen Number attribute 6-11
sea
choppiness 11-5
spray 11-18
enabling and disabling 11-20
state 11-5, 11-14
Douglas 11-12
swell 11-5, 11-14
sea spray, size 11-21
season 11-3
seasonal buoys and beacons 26-27
secondary, simulation object 16-9, 16-12
segment
angle, setting 36-5
arc size 36-5
edge 36-6
emitter volume 36-4
outline, displaying 36-6
selecting
observer 14-6
props 25-25
simulation objects 19-2
send buffer size 4-7
DIS 4-7
Send Port parameter 4-8
Send Weather Control Packet dialog box 42-9
SenSim 12-2
sensor
channel attribute 6-12
gimbaled, field of view 17-9
mode, full color 12-5
tracking, resetting 9-2
sensor effects 12-2, 12-3
blur, noise, and gain 12-4
configuring 12-4
Sensor Settings page 12-4, 12-5, 12-6
server
terrain
caching 29-2
cutting in site 26-16
setting
color, unit bounding volume 20-10
date 11-3
emitter volume radius 36-3
factory and default 3-15
frame rate 3-6
icons 3-15
loading 3-15
location of saved 3-17
LOD scale 13-8
precipitation
intensity 11-11
type 11-11
prop opacity 25-26
replicating 6-3
saving 3-15
segment arc size 36-5
sound volume 23-3
time scale 11-4
visibility 11-10
settings, synchronizing 3-17
Settings menu 3-15
shader 1-6, 29-10
debugging 29-13
effects map 29-12
shadow
configuring 13-18
enabling and disabling 13-16
CIGI in 41-8
quality, setting in CIGI 41-8
text, 2D label 19-9
Shadow Settings page 13-17, 13-18
shake, camera 15-23
shaking camera 15-23
shape terrain data 25-3
shapefile
buoys and beacons 26-20
layer name 26-12
mapping to feature 25-14
sharing, resources 5-7
ship
wake, configuring 35-14
Short Articulated Part Control 41-3
Short Component Control 41-3
short component control message 41-6
Show Function Profiler Overlay 5-5

Show Performance Statistics Overlay check box 5-4
Show Self Published Objects, DIS 4-7
showCockpits keyword 6-22
showHuds 6-12
showing
  objects 40-2, 40-4
  symbol decoration 19-8
  vertex labels 40-5
showLines parameter 36-6
showRemoteGraphics keyword 6-22
SHP, file 25-3
shutdown 3-9
SIGGRP, attribute 26-29
signal
  interaction 21-15
  sequence 26-29
signal group attribute 26-29
signal period attribute 26-29
SignificantSize_SwitchDistance.xml 28-17
SIGPER, attribute 26-29
SIGSEQ attribute 26-29
SilverLining 1-8
  license 1-8
Simthetiq 1-10
simulated world, navigating 15-3
simulation
  connecting to 4-2
  connection
    adding 4-3
    deleting 4-4
    renaming 4-4
  network
    connecting to 3-11
    disconnecting from 3-14
  objects 5-2
Simulation Connections dialog box 3-12, 4-2
Simulation Connections Toolbar 3-14
simulation object
  attaching observer to 16-6
  count 19-15
  element definition 33-2
  embarked 19-15
  filtering attachment list 16-5
  filtering using interest management 19-16
  icon, image 35-9
  inset view 17-2
simulation object (continued)
  label 19-7
  name 19-3
  pinning intervisibility object to 22-10
  See also entity and unit
  track history 21-8
  trailing effect 21-6
Simulation Object Type Mappings dialog box 34-2
  filtering element definition list 34-7
site, cutting in 26-16
Site Number, HLA 4-5
site number, DIS 4-6
size
  region 19-17
  sea spray 11-21
  SpeedTree, randomizing 35-25
  splash effect 11-21
  water droplet 11-21
  window 6-11
skybox cube map 13-13
  omitting clouds 13-13
Skybox Cube Map Regeneration Threshold 13-13
smallCompass, keyword 15-5
smallCompass keyword 6-22
smoke
  plume 21-11
  trail 21-6
smooth period 21-4
smoothing 21-4
snooper.py 43-3
snow 11-5
  intensity 11-11
  remote 11-22
socket
  send and receive buffer size 4-7
  web 8-2
software supported 1-3
soil type, configuring for GDB 24-8
sound effects 23-2
  adjusting volume 23-3
  enabling 23-2
  mapping, weapon fire and detonation 23-8
  mapping to entities 23-4
  range 23-3
sound file, mapping to entity 23-5
sound mapping
  channel 23-8

sound mapping (continued)
detonation result 23-8
Space Follow mode 16-10
spacing, contour line 24-12
specifying
federation execution name 4-5
language 3-6
log file 3-6
notification level 3-8
RPR FOM, version 4-5
segment angle 36-5
specular
color 13-3
map 1-6, 29-10
speed 19-3
gain 15-27
scaling 15-25
enabling and disabling 15-26
specifying measurement unit 45-6
tidal current 11-12, 11-15
wind 11-9
speed scaling, mode 15-26
SpeedTree 1-8, 26-10
configuring 35-24
displaying in planar reflection 35-26
license 1-10
LOD, zoom 3-9
model definition 32-5
scale, random 35-25
SpeedTree Randomization page 35-26
SpeedTree Settings page 35-24
speedTreeManager 26-10
SpeedTrees, planar reflection 13-8
splash effect 11-18
size 11-21
spotlight 13-5
spray, enabling and disabling 21-23
squawk indicator 21-15
color 21-18
enabling and disabling 21-16
lifetime 21-17
starting
display engine 6-22
host emulator, CIGI 42-3
Linux on 3-4
video stream 8-3
VR-Vantage applications 3-2
Windows on 3-2
startup
attaching observer 16-15
startup (continued)
connecting to network 3-13
state
sea 11-14
update, sending 14-10
statistics, displaying 5-3
Stealth, observer mode 3-17
SteamVR 9-2
stencil bits 3-8
stencil buffer 5-8
stereo display, types of 6-28
streamed data
performance 5-11
processing 26-10
streaming
beacon 26-25
buoys and beacons 26-20
navigation lights 26-26
terrain, configuring cache 29-8
video 8-2
streaming video, web browser 8-13
street light 13-5
sub-dataset
elevation, CDB 28-4
extended imagery, CDB 28-16
geospecific, CDB 28-8
geospecific airport lightpoint, CDB 28-9
geospecific environmental lightpoint, CDB 28-10
geotypical, CDB 28-15
imagery, CDB 28-5
imagery lightmap, CDB 28-6
subscribing to multicast groups 4-7
sun, ray 13-11
sunbeam 13-11
sunlight 11-3
support, technical xxviii
supported software 1-3
surface
clamping to 21-2
transparency 6-21, 11-12, 11-15
surface entity
buoyancy 21-24
wake 21-6
configuring 35-14
surge depth 11-5, 11-12, 11-15
swell 1-6, 11-5, 11-12
sea 11-14
switch bead 39-8
switch node 35-20, 35-22

SwitchStateMapper, feature 35-22
symbol decoration, showing and hiding 19-8
Symbol Decoration Settings page 19-7, 19-8, 19-9
synchronization, vertical 5-8
synchronized, ocean 3-8
synchronizing
  application data 6-3
  data and settings 3-17
synchronous, CIGI operation 41-2
system component control message 41-6
System Components mappings page 41-8

T

-t command-line option 27-5
tactical graphic 34-2
  element definition 33-2
  HAT lines 40-5
  hiding from Overlays tab 40-4
  label 40-5
  showing and hiding 40-2, 40-4
Tactical Graphics Display Settings page 40-4, 40-5, 40-6
tactical smoke, enabling or disabling 21-13
taillight 13-5
target address 4-6
TCP, CIGI 41-2
technical support xxviii
teleporting observer 15-17
template, earth file 29-4
terrain
  agility 1-5
  agility and composability 25-2
  altitude point
    changing color 24-11
    deleting 24-10
    inserting 24-10
  building 25-7
  caching 29-2, 29-3
  coloring by elevation 24-9
  creating 25-5
  database 1-5
  databases provided 23-2
  debugging 29-15
  deleting cached 29-5
  displaying raster image on 25-9
  drag, disabling 15-15
  dragging with mouse 15-14
terrain (continued)
  DTED support 25-5
  dynamic 3-6
  earth file, debugging 29-9
  extracting props from 25-4
  formats, supported 25-3
  GDB, soil type 24-8
  intersector line 29-15
  Little Pond 23-2
  loading 3-2
    from command line 3-9
  local 3-10
  Makland 23-3
  MetaFlight
    designing 27-2
    processing 27-3
  opening 3-10
  orbiting, with mouse 15-16, 15-19
  paged
    configuring LODs 29-8
    flipping DDS textures 29-7
    preprocessing 29-8
  patch 25-3
    adding to terrain 25-7
    extracting props from 25-21
  prop
    adding 25-21
    list 25-25
  reloading 3-11
  San Luis Obispo 23-3
  saving 25-6
  scale, exaggerating 18-5
  scaling 18-5, 18-7
    clearing track histories 18-8
    enabling and disabling 18-6
  server. See terrain server
  shape 25-3
  streaming, cache 29-8
  wireframe mode 24-13
Terrain Contents page 25-14, 25-17, 25-18, 26-3, 29-3, 29-9
Terrain Information Panel 24-4
Terrain Scale toolbar 18-7
terrain server 26-2
  adding 26-5
  caching data 29-2
  connecting through proxy 26-5
  connecting to 3-2
  cutting in site 26-16
  mask, adding 26-16

Terrain Settings dialog box
Add Props page 25-23
Edit Existing Props page 25-25, 25-26, 25-27, 30-11, 38-10
Extract Ocean page 25-16
Extract Props page 25-21, 30-9
Raster Maps page 25-9, 25-13
SpeedTree Randomization page 35-26
Terrain Contents page 25-14, 25-17, 25-18, 26-3, 29-9
Visible Surfaces page 24-8
TerraPage file 25-3
TerrraPage file 29-8
Tether Location Track mode 16-12
Tether mode 15-8, 16-6, 16-11
Tether Track mode 16-12
text, shadow 19-9
texture 35-27
atlas, buoy and beacon 26-24
compression, enabling and disabling 29-5
coordinate system for image 25-11
DDS 29-6
flipping, paged terrain 29-7
height, ocean planar reflection 13-8
map 1-6, 29-10
mapping to model 35-21
model 35-21
sharing 5-7
size, ocean height map 25-18, 25-19
skinning
metafile 35-22
model 35-21
tiled and virtual 27-4
water 25-15
width, ocean planar reflection 13-8
texture atlas 35-21
thickness, contour line 24-12
threading, CIGI 41-2
tidal current
direction 11-15
speed 11-15
speed and direction 11-12
tidal stream
speed and direction, configuring in CIGI 35-16
wake 11-16, 35-16
TIFF 25-3
tiled texture 27-4
time
scale 11-4
time scale 11-4
time of day 11-3
synchronizing with VR-Forces 11-22
time stamps, absolute 4-6, 4-7
Timeout Interval, DIS 4-7
timeout interval, DIS 4-7
tire track 21-6
To Echelon Level 20-6
tool, osgearth_cache 29-3
toolbar
Attachments 16-2
Display Settings 19-8
displaying and hiding 45-3
Observer Settings 18-4, 19-11
Simulation Connections 3-14
Terrain Scale 18-7
topmark, buoy and beacon 26-28
topmark_attr_map.csv 26-28
TOPSHP attribute 26-28
tracer 21-11
track history
configuring 21-9
displaying 21-8
enabling and disabling for all simulation objects 21-8
length 21-9
scaling 18-8
Track History Settings page 21-10
Track mode 15-8, 16-6, 16-13
tracking sensor, resetting 9-2
traffic, monitoring 4-7
trail, model 21-8
trailing effects 21-6, 34-2
enabling and disabling 21-7
model 21-8
simulation object 21-6
trajectory, smoothing 21-4
trajectory history. *See* track history
transient intervisibility
displaying 22-6
object 22-5
translating, GUI text 2-10
transparency 35-27
2D icon 19-14
prop, setting 25-26
surface 6-21, 11-5, 11-12, 11-15
Transport Type parameter 4-8
tread track 21-6

tree
  prop type 25-27
  SpeedTree 35-24
triangle, vertex label 40-5
Triton Ocean SDK 1-9
TrueType font, 2D icon 35-3
turret, attaching to 16-4
tutorial
  adding model 38-2
  attaching observer 16-16
  CIGI host emulator 42-2
  configuring display engine 7-2
TXP file 25-3
type, prop 25-27
type attribute 35-5

U

-u command-line option 27-5
UDP, CIGI 41-2
underwater, visibility 11-12, 11-15
undocking, panel 45-2
unique, buoy 26-23
unit
  bounding volume, color 20-10
  collapsing 20-4
  collapsing to root 20-4
  count of 19-15
  display 45-6
  displaying 20-2
  displaying ghosted icons 20-5
  expanding 20-4
  expanding all 20-4
  expanding and collapsing 20-2
  measurement 45-6
  See also simulation object and entity
Unit Display Settings page 20-6, 20-9, 20-10
update
  messages 14-10
  rate, intervisibility 22-9
updater 35-10
  AOA 35-13
  attribute 35-10
  FeetAGL 35-13
  FeetMSL 35-13
  FeetPerMinute 35-13
  Headlight 35-13
  KilometersPerHour 35-13
  KnotsPerHour 35-13
updater 35-10 (continued)
  MilesPerHour 35-13
  Pitch 35-13
  Roll 35-13
  Value 35-13
  Yaw 35-13
upgrades xxviii
Use Absolute Time Stamps, DIS 4-7
Use Extracted Geometry 25-21
Use Topo For Flat Earth Vector Conversion 4-6, 4-7
user data, specifying directory 3-9
user-defined CIGI packet 41-3
using
  Boundary Generation Tool 26-17
  video stream 8-9
UTM, coordinate system 15-7
UV, coordinates 25-11
UV offset 35-23
UvAtlasMapper, feature 35-22

V

-v command-line option 3-9, 27-5
Value updater 35-13
vapor trail 21-6
vector feature, list 24-4
vehicle, model 1-4
velocity
  displaying 19-4
  specifying measurement unit 45-6
version
  DIS 4-6
  displaying 3-9
vertex
  altitude, displaying 40-5
  showing and hiding 40-5
video
  encoding 8-2
  streaming 8-2
    web browser 8-13
video card, processor speed 5-2
video stream
  auto start 8-4
  capture buffer 8-8
  configuring 8-4
  pipeline 8-4
  reducing latency 8-10
  starting 8-3
  using 8-9

video stream (continued)
viewing with VLC 8-10
Video Streams Configuration page 8-5, 8-13
Video Streams dialog box 8-3, 8-4, 8-5, 8-13
Video Streams page 8-3, 8-4
view
binding to observer 16-17
CIGI 41-12
mapping to observer 41-12
clearing list 15-30
constraint 15-25
deleting 15-30
dragging with mouse 15-16
floor 15-25
inset 17-2
loading saved 15-32
navigating 15-3
observer, transitioning between 15-29
recalling 15-28
restoring 15-29
saving 15-28, 15-29, 15-31
individual 15-31
zooming 15-19
2D 15-19
3D 15-20
view component control 41-9
message 41-6
View Control 41-3
view control
overriding global setting 15-33
packet, CIGI 41-12
view control message 15-32
enabling processing of message 15-32
script 15-32
View Definition 41-3
view mode
Absolute 16-7
Follow 16-7
Mimic 16-8
Mimic Location Track 16-9
Mimic Track 16-9
Space Follow 16-10
Tether 16-11
Tether Location Track 16-12
Tether Track 16-12
Track 16-13
viewing
list of props 25-25
viewing (continued)
other MAK 3D GUI viewers 14-11
viewport 6-12
changing 6-20
virtual reality
configuring 9-6
enabling and disabling 9-4
frame rate 9-5
performance 9-5
starting VR-Vantage with 9-3
tracking sensor, resetting 9-2
Virtual Reality Settings page 9-4
virtual texture 27-4
Virtual Texture Datasets 27-3
visibility 11-10
observer 11-5
of observer to other applications 14-10
remote 11-22
underwater 11-5, 11-12, 11-15
water 6-21
visible surface, configuring 24-8
Visible Surfaces page 24-8
visual definition 31-2, 33-2, 33-7
adding 33-7
attribute 33-8
adding 33-11
editing 33-8, 33-10
Entity Image Symbol 35-2, 35-3, 35-9
Military Symbol Icon 35-2
schema 31-2
Visual Model Editors dialog box 32-2
visual quality 25-18
visual sensor, gimbaled 17-8
visualizer 33-2
VLC 8-9
reducing latency 8-10
viewing video stream 8-10
VLC Out, configuring GStreamer 8-7
volume
sound, adjusting 23-3
VR-Forces
aggregate entities 20-2
weather, enabling 11-22
VR-Vantage
features 1-2
main features 1-3
shutdown 3-9
viewing other 14-11
VR-Vantage Control Toolkit 15-32
VR-Vantage IG 1-1

VR-Vantage PVD 1-1
VR-Vantage Stealth 1-1
VR-Vantage Toolkit 1-1
VSync 3-6, 5-8
VTDS 27-3

W

-W command-line option 3-9, 6-9, 7-8
-w command-line option 27-6
wake 1-6, 1-9, 21-6
  enabling and disabling 21-23
  model definition 11-16
  ship, configuring 35-14
  tidal stream 11-16, 35-16
wakeLength parameter 11-16
wall, not walking through 15-25
water
  splash 11-18
  texture 25-15
    extracting 25-16
  visibility 6-21
water droplet, size 11-21
wave
  breaking 11-17
  choppiness 11-12
  effect 11-18
waypoint
  altitude, displaying 40-5
weapon
  fire, sound mapping 23-8
weather 11-5
  changing in host emulator 42-9
  configuring 11-9
  remote 11-22
  synchronizing with VR-Forces 11-22
Weather Control 41-3
weather control message 42-9
weather control packet, CIGI 44-9
web browser, streaming video to 8-13
web socket 8-2
WebSocket 8-9
websocket, streaming video 8-13
WidgetTheme, attribute 35-6
wind 1-6
  direction 11-5
  moving flags and windsocks 35-20
  speed 11-5
  speed and direction 11-9
WindDirPart 35-20
window 6-4
  Add Visualizer Definition Attribute 33-11
  adding 6-5
  attributes, editing 6-11
  channel, adding 6-7
  Choose a Terrain 3-2
  location, changing 6-11
  observer 14-6
  property 6-11
  removing 6-8
  size, changing 6-11
  specifying 3-6
  types 6-4
Window Name attribute 6-11
Window Type attribute 6-11
Windows
  installing on 2-2
  starting VR-Vantage 3-2
windsock, movement in wind 35-20
WindSpeed parameter 35-20
WindState parameter 35-20
WindSwitchPart 35-20
wireframe mode 24-13
  enabling in CIGI 41-9
world coordinates 15-3
WRM Entity Model Specification 39-1, 39-9

X-Y-Z

-x command-line option 27-6
X screen 6-32
XML 26-2
  include files 26-8
  RTI configuration file 2-9
XXX_SpeedTrees.xml 28-18
yaw 15-3
Yaw updater 35-13
Z Far 6-12
Z Near 6-12
Z-fighting 3-9, 6-15, 6-21, 16-6, 25-15
zoom
  scale factor, LOD 15-21
  SpeedTree LOD 3-9
  switching from 3D to 2D 14-2
zooming
  observer 15-19
    2D 15-19
    3D 15-20

zooming (continued)
performance 15-22



Link - Simulate - Visualize

VRV-2.3-1-180226

150 CAMBRIDGE PARK DRIVE, 3RD FLOOR CAMBRIDGE, MA 02140 617.876.8085 www.mak.com

VR-Vantage Quick Reference Card

Table 1: Command-Line Options

Option	Description
(-- | --ignore_rest)	Ignore all command-line options after this one.
--alphaBits *bits*	Sets the number of alpha bits (0, 8).
--antiAliasing *level*	Set the anti-aliasing level (0, 2, 4, 8, or 16). Default: 4.
--appDataDir *directory*	Specifies the location of application data.
--autoReconnectPeriod *period*	Specifies the amount of time, in seconds, to wait before trying to attach to disconnected remote displays.
-B | --bindObservers "*observer1, observern,saved_views_file*"	Assigns the views in the specified saved view file to the observers in the order that the observers are listed in the command. Must also specify terrain file. For multiple observers, must specify display engine configuration with --dispSetting.
(-c | --showConsole)	Display a console window.
(-C | --autoConnect) {*filename* | *connector_name*}	Automatically connect to the network using the specified connector.
--dataDir *directory*	Specifies the location of the data directory .
--depthBits *bits*	Sets the depth of graphics to 24 bits or 32 bits. Default: 24.
--disableDynamicTerrain	Disables the dynamic terrain engine.
--discoveredModelPreloadFileName *filename*	Specifies a file to which the list of models discovered during a session that were not preloaded is written.
--displayEnginePort *port*	Specifies the display engine port used by a remote display engine. Used only if --slave is specified.
--dispSetting *filename*	Specifies the display engine configuration to load.
--enableVSync	Turns on VSync.
--entDispSetting *filename*	Specifies the entity display setting to load.
--entityDefsDir *directory*	Loads the *.hier* and *.leaf* files in the specified directory.
--envSetting *filename*	Specifies the environment configuration file to load.
(-E | --entTypeMap) *filename*	Load the specified entity type mapping file. The file extension determines the type of mapping.
--factoryRootDir	Set the root directory used to restore settings to factory defaults.
--forceTextShaping	Force all text to use HarfBuzz shaping.
--forceVBO	Forces use of vertex buffer objects in OpenGL.
--fullScreen	Start in full screen mode.
(-F | --frameRate) *rate*	Specifies a fixed frame rate for rendering. If 0, it is variable.
(-G | --locale) *language*	Specifies the locale (for localization).
--gui *filename*	Specify GUI configuration file to load.
(-h | --help)	Displays command-line usage information.
--highestPriority	Changes application and the main render thread to the highest available priority.
(-I | --unbatchedRendering)	Disables instance rendering.
(-K | --disableKDTrees)	Disable creation of KD trees for intersections.
(-l | --logFileName) *filename*	Specifies the log file name.

Copyright © 2018 VT MAK. All rights reserved. Document ID: VRV-2.3-QRC-180226

Table 1: Command-Line Options

Option	Description
(-L | --loadObservers) *filename*	Loads the saved observers in the specified file. Can be used multiple times.
--mainScreenNum *num*	Specifies the display that the main window should use.
--mergeEntityTypeMap {*directory* | *filename*}	Merges all entity type mapping files (*.*metx*) in a directory or individual entity type mapping files. Can be used multiple times.
--mergeHierarchyEntityDef *filename*	Merges the specified *.hier* file into the default hierarchy. Merging is additive.
--mergeHierarchyTacticalGraphicsDef *filename*	Merges the specified *.hier* file into the default hierarchy. Merging is additive.
--mergeHierarchyUnitDef *filename*	Merges the specified *.hier* file into the default hierarchy. Merging is additive.
--mergeLeafEntityDef *filename*	Merges the specified leaf file into the default entity element definitions.
--mergeLeafTacticalGraphicsDef *filename*	Merges the specified leaf file into the default tactical graphics element definitions.
--mergeLeafUnitDef *filename*	Merges the specified leaf file into the default unit element definitions.
--mergeModelDefs {*directory* | *filename*}	Merges all model definition files (*.*ommx*) in the specified directory or individual model definition files. Can be used multiple times.
--mergeTacticalGraphicsTypeMap {*directory* | *filename*}	Merges all tactical graphics type mapping files (*.*metx*) in a directory or individual tactical graphics type mapping files. Can be used multiple times.
--mergeUnitTypeMap {*directory* | *filename*}	Merges all unit type mapping files (*.*metx*) in a directory or individual unit type mapping files. Can be used multiple times.
--modelPreloadFileName *filename*	Specifies a file that has a list of models to preload.
(-n | --notifyLevel) *notify_level*	Specifies the notification level for application messages. Range: 0-4.
--noAppDataEntityDefs	Do not load the entity element definitions in *./appData/definitions*.
--noAppDataTacticalGraphicsDefs	Do not load the tactical graphics element definitions in *./appData/definitions*.
--noAppDataUnitDefs	Do not load the unit element definitions in *./appData/definitions*.
--noAppDataEntityTypeMap	Do not load entity type mappings in *./appData/settings/<application>*.
--noAppDataModelDefs	Do not load model definitions in *./appData/definitions*.
--noAppDataTacticalGraphicsTypeMap	Do not load tactical graphics type mappings in *./appData/settings/<application>*.
--noAppDataEntityTypeMap	Do not load unit type mappings in *./appData/settings/<application>*.
--noGui	Start without the graphical user interface.
(-O | --hasNTPSync)	Enables synchronized ocean between different display engines. System clocks must be synchronized.
--plugin *filename*	Load the specified plug-in. Can be used multiple times.
--pseudoFullScreen	Use pseudo full screen for full screen mode.
(-s | --slave)	Run as a remote display engine.
(-S | --suppressHatIntersectOnAttach)	Suppress the periodic HAT intersection when attached.
--stencilBits *bits*	Sets the number of stencil bits (0 or 8). Default: 0.
--tcpSendTimeout	Sets the timeout (in seconds) for the master display engine to wait for a remote display engine before giving up on a send. Default: 30 seconds.
(*terrain_MTF_file* | *scene_MSF_file*)	Specifies a terrain (MTF) or scene (MSF) file to load.
--unplug *filename*	Do not load the specified plug-in. Can be used multiple times.

VR-Vantage Quick Reference Card

Table 1: Command-Line Options

Option	Description
`--userDataDir` *directory*	Specifies the user data directory.
`(-v | --version)`	Display version information and exit.
`-W | --withDisplayEngine` *port@netAddress;displayConfigFile name1 .... n* `-W | --withDisplayEngine` *port@netAddress:displayConfigFile name1 .... n*	Connect to a display engine running in subordinate mode on another computer. On Windows, use a semi-colon (;). On Linux, use a colon (:).
`(-z | --useReverseZBuffer)`	Specifies use of reverse Z-buffering, which can reduce Z-fighting.
`(-Z | --ignoreZoomForSpeedtreeLod`	Disable zoom for SpeedTree LOD.

Table 2: Mouse Button Mappings

Navigation Action	Mouse Action	Context
Attach to entity.	**Shift** + left-click	Entity.
Attach to prop.	**Ctrl** + **Shift** + left-click	Prop.
Deselect all.	Left-click	Terrain or sky.
Drag terrain.	Left-click + drag	Terrain or prop.
Object selected, others unselected.	Left-click	Selected entity or prop.
Select entity or prop.	**Ctrl** + left-click	Unselected entity or prop.
Select object and unselect all others.	Left-click	Unselected entity or prop.
Teleport to location.	**Shift** + left-click	Terrain.
Teleport to prop.	**Shift** + left-click	Prop.
Teleport to entity.	**Ctrl** + **Shift** + left-click	Entity.
Unselect.	**Ctrl** + left-click	Selected entity or prop.
Display context-sensitive menu.	Right-click	Entity or prop.
3D Only		
Drag the view.	Middle mouse button drag	All.
Drag heading and pitch (mouse look).	Right Drag	All.
Orbit drag.	**Ctrl** + left-click + drag	Terrain or prop.
Zoom in (magnification).	**Ctrl** + mouse wheel forward	All.
Zoom out (magnification).	**Ctrl** + mouse wheel backward	All.
Reset magnification.	Middle mouse button	All.
Move observer forward.	Mouse wheel forward	All.
Move observer back.	Mouse wheel back	All.
Move observer level observer forward.	**Shift** + mouse wheel forward.	All.
Move observer level observer backward.	**Shift** + mouse wheel backward.	All.
Move flashlight beam.	Right mouse button drag	All.
Expand flashlight beam.	**Alt** + mouse wheel forward.	All.

Copyright © 2018 VT MAK. All rights reserved. Document ID: VRV-2.3-QRC-180226

Table 2: Mouse Button Mappings

Navigation Action	Mouse Action	Context
Contract flashlight beam.	**Alt** + mouse wheel backward.	All.
2D Only		
Rotate terrain.	Right-click + drag	All. Orient North disabled.
2D Zoom in.	Mouse wheel forward; Middle button + drag	All.
2D Zoom out.	Mouse wheel backward; Middle button + drag	All.

K = numeric keypad (K8 means press Keypad 8). To use the numeric keypad for navigation, NumLock must be on.

Table 3: Observer Keyboard Mappings

Observer Movement	Observer	Level Observer	Laptop	2D Level Observer
Move forward.	**w**	unmapped	**w**	unmapped
Move back.	**s**	unmapped	**s**	unmapped
Move left.	**a**	unmapped	**a**	unmapped
Move right.	**d**	unmapped	**d**	unmapped
Move up.	**q**, **Ctrl** + **q**	unmapped	**q**, **Ctrl** + **q**	unmapped
Move down.	**e**, **Ctrl** + **e**	unmapped	**e**, **Ctrl** + **e**	unmapped
Move forward level observer.	unmapped	**w**	unmapped	**w**
Move back level observer.	unmapped	**s**	unmapped	**s**
Move left level observer.	unmapped	**a**	unmapped	**a**
Move right level observer.	unmapped	**d**	unmapped	**d**
Move up level observer.	unmapped	**q**, **Ctrl** + **q**	unmapped	unmapped
Move down level observer.	unmapped	**e**, **Ctrl** + **e**	unmapped	unmapped
Move attached context forward.	**Shift** + **w**	**Shift** + **w**	**Shift** + **w**	**Shift** + **w**
Move attached context back.	**Shift** + **s**	**Shift** + **s**	**Shift** + **s**	**Shift** + **s**
Move attached context left.	**Shift** + **a**	**Shift** + **a**	**Shift** + **a**	**Shift** + **a**
Move attached context right.	**Shift** + **d**	**Shift** + **d**	**Shift** + **d**	**Shift** + **d**
Move attached context up.	**Shift** + **q**	**Shift** + **q**	**Shift** + **q**	unmapped
Move attached context down.	**Shift** + **e**	**Shift** + **e**	**Shift** + **e**	unmapped
Level yaw left.	K **4**	K **4**	Left arrow	Left arrow
Level yaw right.	K **6**	K **6**	Right arrow	Right arrow
Level pitch up.	**Shift** + **j**	**Shift** + **j**	**Shift** + **j**	unmapped
Level pitch down.	**Shift** + **k**	**Shift** + **k**	**Shift** + **k**	unmapped
Level roll right.	**Shift** + **i**	**Shift** + **i**	**Shift** + **i**	unmapped
Level roll left.	**Shift** + **m**	**Shift** + **m**	**Shift** + **m**	unmapped
Yaw left.	**Shift** + **h**	**Shift** + **h**	**Shift** + **h**	**Shift** + **h**
Yaw right.	**Shift** + **l**	**Shift** + **l**	**Shift** + **l**	**Shift** + **l**

Table 3: Observer Keyboard Mappings

Observer Movement	Observer	Level Observer	Laptop	2D Level Observer
Pitch up.	K 8	K 8	↑	unmapped
Pitch down.	K 2	K 2	↓	unmapped
Roll right.	K 9	K 9	K 9	unmapped
Roll left.	K 3	K 3	K 3	unmapped
Orbit right.	Ctrl + d, →	Ctrl + d, →	Ctrl + d	unmapped
Orbit left.	Ctrl + a, ←	Ctrl + a, ←	Ctrl + a	unmapped
Orbit up.	Ctrl + w, ↑	Ctrl + w, ↑	Ctrl + w	unmapped
Orbit down.	Ctrl + s, ↓	Ctrl + s, ↓	Ctrl + s	unmapped
Look up.	Ctrl + K 8	Ctrl + K 8	Ctrl + K 8	Ctrl + K 8
Look down.	Ctrl + K 2	Ctrl + K 2	Ctrl + K 2	Ctrl + K 2
Look left.	Ctrl + K 4	Ctrl + K 4	Ctrl + K 4	Ctrl + K 4
Look right.	Ctrl + K 6	Ctrl + K 6	Ctrl + K 6	Ctrl + K 6
Look tilt left.	Ctrl + K 3	Ctrl + K 3	Ctrl + K 3	Ctrl + K 3
Look tilt right.	Ctrl + K 9	Ctrl + K 9	Ctrl + K 9	Ctrl + K 9
Zoom In	unmapped	unmapped	unmapped	Ctrl + e
Zoom Out	unmapped	unmapped	unmapped	Ctrl + q
Suppress constrain to terrain.	x	x	x	x
Water Impact	unmapped	unmapped	unmapped	unmapped
Reset observer:	Space	Space	Space	Space
In absolute mode, go to the center of the scene. If attached, reset the attach position and orientation.				
Toggle Speed Scaling Mode (MSL or AGL)	unmapped	unmapped	unmapped	unmapped
Level observer.	K 5	K 5	Shift + Space + ↓	K 5
Toggle Force North Orientation	unmapped	unmapped	unmapped	unmapped
Toggle Orient To Target Constraint	unmapped	unmapped	unmapped	unmapped
Toggle Flashlight	F	F	F	F
Reset the camera.	K 0	K 0		K 0
Increase View Magnification	=	=	=	unmapped
Decrease View Magnification	-	-	-	unmapped
Reset View Magnification	0	0	0	unmapped
Increase View Resolution	unmapped	unmapped	unmapped	=
Decrease View Resolution	unmapped	unmapped	unmapped	-
Detach.	z	z	z	z
Toggle Hide Model On Attach	unmapped	unmapped	unmapped	unmapped
Attach to the next entity allowed by the attachment filter.	Period (.)	Period (.)	Period (.)	Period (.)
Attach to next entity or prop.	>	>	>	>

Copyright © 2018 VT MAK. All rights reserved. Document ID: VRV-2.3-QRC-180226

Table 3: Observer Keyboard Mappings

Observer Movement	Observer	Level Observer	Laptop	2D Level Observer
Attach to the previous entity allowed by the attachment filter.	Comma (,)	Comma (,)	Comma (,)	Comma (,)
Attach to previous entity or prop.	<	<	<	<
Set Mimic Attach Type	unmapped	unmapped	unmapped	unmapped
Set Tether Attach Type	unmapped	unmapped	unmapped	unmapped
Next attach mode.	Shift+z	Shift+z	Shift+z	Shift+z
Increase linear speed.	K 7	K 7	K 7	K 7
Decrease linear speed.	K 1	K 1	K 1	K 1
Increase orbit speed.	K +	K +	K +	K +
Decrease orbit speed.	K –	K –	K –	K –
Increment Rotation Speed Gain	unmapped	unmapped	unmapped	unmapped
Decrement Rotation Speed Gain	unmapped	unmapped	unmapped	unmapped
Toggle speed scaling on or off.	K *	K *	K *	K *
Toggle Prop Transparency	T	T	T	T
Toggle Terrain Scaling	unmapped	unmapped	unmapped	unmapped
Increment Terrain Scale	unmapped	unmapped	unmapped	unmapped
Decrement Terrain Scale	unmapped	unmapped	unmapped	unmapped
Toggle Track Histories	unmapped	unmapped	unmapped	unmapped
Toggle Entity Labels	L	L	L	L
Toggle Rotate Entity Symbols	unmapped	unmapped	unmapped	unmapped
Toggle Tactical Graphics	Shift+G	Shift+G	Shift+G	Shift+G
Toggle Entity Height Above Terrain Lines	unmapped	unmapped	unmapped	unmapped
Toggle Tactical Graphics HAT Lines	unmapped	unmapped	unmapped	unmapped
Toggle Entity Height Above Terrain Mode	unmapped	unmapped	unmapped	unmapped
Toggle Tactical Graphics Height Above Terrain Mode	unmapped	unmapped	unmapped	unmapped
Toggle constrain to terrain on or off.	/	/	/	/
Toggle Constrain To Center	U	unmapped	unmapped	unmapped
Clear the current movement mode. This key is an escape mechanism if keys are not behaving as you expect. This is typically due to a loss of the window context.	c	c	c	c
Next Observer Mode	]	]	]	]
Previous Observer Mode	[	[	[	[
Toggle view controls on or off.	Shift+c	Shift+c		Shift+c
Clear All Trajectory Histories.	unmapped	unmapped	unmapped	unmapped
Toggle OSG Framerate statistics	F2	F2	F2	F2
Print OSG statistics	Shift+[	Shift+[	Shift+[	Shift+[