



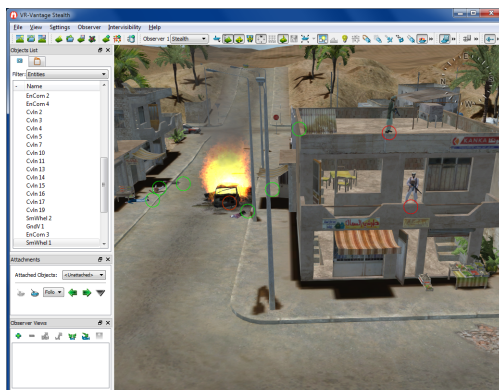
# VR-Vantage Capabilities

## Visualize from Every Vantage Point

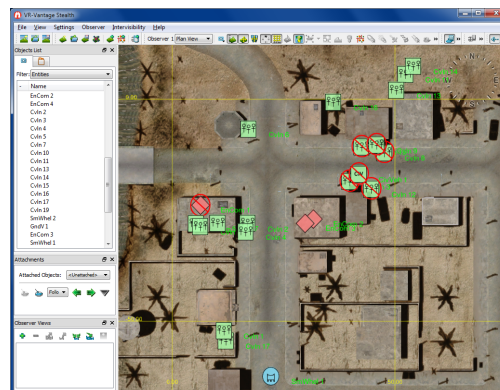
VR-Vantage is VT MÄK's line of visualization products, designed to meet your needs for visualizing the simulated world. The VR-Vantage product line consists of the VR-Vantage Toolkit and the following applications that are built with it:

- **VR-Vantage Stealth.** VR-Vantage Stealth is MÄK's information station where you can view the virtual world in 2D and 3D. Whether you need it for situational awareness, simulation debugging, or after action review, VR-Vantage Stealth provides the most data about your networked virtual world and presents it in a clear and accessible way. With VR-Vantage Stealth, you can quickly achieve a “big picture” understanding of a battlefield situation while retaining an immersive sense of perspective.
- **VR-Vantage IG.** VR-Vantage IG is MÄK's configurable desktop Image Generator (IG) for out-the-window (OTW) scenes, camera views, and sensor channels. Its built-in distributed rendering architecture supports many different display configurations — from simple desktop deployments to multichannel displays for virtual cockpits and training systems.
- **VR-Vantage PVD.** VR-Vantage PVD is MÄK's 2D solution for interacting with the virtual battlefield. This plan view display is the right interface for providing situational awareness and managing entities in large areas, as well as keeping track of a single trainee's location. Whether you're building a command center or a cockpit trainer, VR-Vantage PVD presents a tactical view of your simulation environment using raster graphic maps or terrain database maps, operational and analytical graphics, and MILSPEC 2525b entity icons. Used stand-alone or integrated into your training systems, VR-Vantage PVD is the high performance mapping solution.

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 also 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, so you can leverage value-added plug-ins built by the OSG community and MÄK partners.



VR-Vantage Stealth



VR-Vantage PVD

# VR-Vantage Capabilities

VR-Vantage Capabilities Matrix

## VR-Vantage Capabilities Matrix

Table 1: VR-Vantage capabilities

| Feature                        | IG | Stealth | PVD | Feature                                        | IG | Stealth | PVD |
|--------------------------------|----|---------|-----|------------------------------------------------|----|---------|-----|
| 2D entity labels               |    | ●       | ●   | Muzzle flash                                   | ●  | ●       |     |
| 2D icons                       |    | ●       | ●   | Object instancing                              | ●  | ●       | ●   |
| 3D colorized model set         |    | ●       |     | Overlays                                       | ●  | ●       | ●   |
| 3D entity labels               | ●  | ●       |     | Persistent settings                            | ●  | ●       | ●   |
| 3D model set                   | ●  | ●       |     | Procedural terrain                             | ●  | ●       |     |
| Advanced lighting              | ●  | ●       |     | Radio comms lines                              | ●  | ●       | ●   |
| Aggregate visualization        | ●  | ●       | ●   | Remote graphics                                | ●  | ●       | ●   |
| Attach modes                   | ●  | ●       | ●   | Saved display engines                          | ●  | ●       | ●   |
| Audio effects                  | ●  | ●       | ●   | Saved views                                    | ●  | ●       | ●   |
| Contour lines                  | ●  | ●       | ●   | Scenes                                         | ●  | ●       | ●   |
| Communications lines           | ●  | ●       | ●   | Sensors (IR, NVG)<br>- CameraFX<br>- SensorFX* | ●  | ●       |     |
| Depth of Field                 | ●  | ●       |     | S-57 data                                      | ●  | ●       | ●   |
| DI-Guy characters              | ●  | ●       |     | Shadows                                        | ●  | ●       |     |
| Dynamic ocean                  | ●  | ●       |     | Sky, cloud, precipitation                      | ●  | ●       |     |
| Emitter volumes                | ●  | ●       | ●   | SpeedTrees                                     | ●  | ●       |     |
| Ephemeris model                | ●  | ●       |     | Statistics viewer                              | ●  | ●       | ●   |
| Explosions                     | ●  | ●       |     | Stereoscopic display                           | ●  | ●       |     |
| File caching                   | ●  | ●       | ●   | Streaming terrain                              | ●  | ●       | ●   |
| Fire and detonation lines      | ●  | ●       | ●   | Streaming video (H.264, MPEG)                  | ●  | ●       | ●   |
| Flames                         | ●  | ●       |     | Tactical graphics                              | ●  | ●       | ●   |
| GL Studio cockpits             | ●  | ●       |     | Tactical smoke                                 | ●  | ●       |     |
| Ground clamping                | ●  | ●       |     | Terrain agility                                | ●  | ●       | ●   |
| GUI-based model mapping        | ●  | ●       | ●   | Terrain elevation coloring                     | ●  | ●       |     |
| Height-above-terrain lines     | ●  | ●       |     | Terrain scaling                                |    | ●       |     |
| Icon scaling                   |    | ●       | ●   | Texture mapping                                | ●  | ●       |     |
| Inset views                    | ●  | ●       |     | Track histories                                | ●  | ●       | ●   |
| Interest management (HLA)      | ●  | ●       | ●   | Trajectory smoothing                           | ●  | ●       | ●   |
| Intervisibility                |    | ●       | ●   | Trailing effects                               | ●  | ●       |     |
| Laser designators              | ●  | ●       | ●   | View control support                           | ●  | ●       | ●   |
| Measurement unit customization | ●  | ●       | ●   | Visualized feature data                        |    | ●       | ●   |
| Model scaling                  |    | ●       |     | Zoom observer (3D)                             | ●  | ●       |     |
| Multiple observers             | ●  | ●       | ●   |                                                |    |         |     |

\*SensorFX provides physics-based sensors views. It is a separate cost plug-in to VR-Vantage.

# VR-Vantage Capabilities

## Visualization Features

### Visualization Features

VR-Vantage Stealth and IG deliver a realistic, three-dimensional view of terrain, vehicles, and other objects. 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.

VR-Vantage PVD provides a 2D plan view style view of the terrain to give you the big picture of a simulation.

### Visualizing Terrain

Simulations must exist in the context of a simulated world. VR-Vantage applications support multiple terrain formats and allow you to combine elevation data, imagery, and feature data to support your simulated environment.

### Terrain Agility and Composability

VR-Vantage allows you to build your terrain at runtime using a variety of database and vector formats. VR-Vantage supports the following broad types of terrain:

- **Terrain models.** Static terrains, such as OpenFlight, built using terrain construction applications.
- **Paging terrain.** Large area terrains that page in multiple terrain pages, such as MetaFlight.
- **Direct from source.** Terrains composed by combining various types of terrain elevation, imagery, and features data.

VR-Vantage can load OpenFlight DTED, CTDB, MetaFlight, and GDB (MÄK 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.

- **Open streaming terrain.** Terrains that stream data from various public and private data sources, such as VR-TheWorld.

### Streaming and Paging Terrain

For simulation on large terrains, VR-Vantage can stream data from external servers or from local directories. VR-Vantage applications can stream elevation and imagery from terrain servers such as VR-TheWorld or other WMS-C (Web Mapping Service-Cached, from Open Geospatial Consortium) and TMS (OSGeo's Tile Map Service) servers.

VR-Vantage uses osgEarth to import streaming terrain elevation and imagery data. (osgEarth is an open source plug-in to OpenSceneGraph, maintained by Pelican Mapping at <http://osgEarth.org>.) For MetaFlight, VR-Vantage has its own pager.

# VR-Vantage Capabilities

## Visualization Features

### Procedural Terrain

A procedural terrain applies high-resolution geo-typical textures to the terrain based on soil-type information rather than using satellite imagery. This approach provides high-quality visuals over large areas without the performance cost of high-resolution imagery. Customer who use procedural terrain will usually insert locally accurate imagery and features for the areas that they are particularly interested in.

### Visualization of Feature Data

VR-Vantage PVD can display the raw feature data of shape files (or VMAP) based on color coded mappings. VR-Vantage can import S-57 data and display buoys and beacons.

### Vegetation

VR-Vantage uses SpeedTree software and content for animated real-time 3D foliage and vegetation. Speed-trees can move with the wind. SpeedTree is developed by Interactive Data Visualization (IDV) (<http://www.speedtree.com>).



### Props

Props are terrain elements, such as buildings, light poles, and vegetation, that you can manipulate through the VR-Vantage GUI. VR-Vantage can import shape (or other feature data file formats) and create geometry (props) for point features in the source data.

## Visualizing Entities

VR-Vantage includes many 3D vehicle models, some of which display movement of articulated parts, such as turrets and landing gear. These models can change to show a damaged or destroyed state. You can provide your own models and map them to entity types. (For more information, please see “[3D Models, Terrain, and Graphical Content](#),” on page 24.)

VR-Vantage can render real time shadows for entities and lifeforms based on the position of the sun.

### Realistic 3D Human Characters

VR-Vantage uses DI-Guy software and content for human character animation. VR-Vantage comes with DI-Guy functionality built-in, and with a large set of DI-Guy characters, appearances, and animations.

### 2D Icons

In PVD mode, VR-Vantage displays entities using MIL-STD 2525B icons. You can adjust the size of icons to declutter the display.

# VR-Vantage Capabilities

## Visualization Features

### Cockpits

VR-Vantage uses GL Studio software and content to render interactive cockpit instrumentation displays. VR-Vantage is delivered with several generic cockpit displays. GL Studio is developed by DiSTI (<http://www.dist.com>).

### Trajectory Smoothing

VR-Vantage can smooth the trajectories of moving vehicles to compensate for discontinuous positional data.

### Surface Entity Movement

When dynamic ocean is enabled, surface entities bob up and down with ocean swells. Destroyed entities sink beneath the waves.

### Ground Clamping

The ground clamping feature can ensure that all ground entities are placed correctly on the terrain surface.

### Inset Views

You can display inset views of individual entities. [Figure 1](#) shows a helicopter hovering over the runway in the Makland terrain with an inset view of an M1A2 tank driving through the town.



Figure 1. Inset view

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.

# VR-Vantage Capabilities

## Visualization Features

### Sound Effects

Sound effects increase the immersive effect of the visual environment. 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.

### Visualizing the Environment

VR-Vantage creates a realistic atmospheric environment with sun, moon, clouds, lighting, and precipitation. For the marine environment, VR-Vantage renders realistic ocean effects, including waves, swells, wakes, and spray effects. Users can choose among several pre-configured environment conditions or adjust any of the weather and marine features to create custom weather conditions. VR-Vantage can display weather conditions sent from VR-Forces.

VR-Vantage uses SilverLining software and content to compute lighting and to render the atmosphere. SilverLining is developed by Sundog Software (<http://www.sundog-soft.com>). VR-Vantage uses the Triton SDK, also from Sundog Software, to create the marine environment.

### Configurable Weather

VR-Vantage lets you set the wind direction and speed, precipitation type and intensity, visibility, and cloud cover. Wind direction and speed affects dynamic ocean. Together with the lighting effects, varying cloud cover creates dramatic environment visualization.



You can render the effect of rain splashing on the observer's camera. When dynamic ocean is enabled, waves can splash onto the observer view.



# VR-Vantage Capabilities

## Visualization Features

### Animated Flags and Windsocks

Flags on terrain and ships, and windsocks respond to wind speed and direction.



### Dynamic Ocean

VR-Vantage supports a variety of dynamic ocean effects, 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.** Lets you calm shallow water to visualize offshore wind and calm harbors.



### Lighting

#### Date and Time

VR-Vantage uses a full-year ephemeris model that changes the position of the sun and moon as a function of date and time of day. You can set the date and advance time in real time or at a faster or slower rate. VR-Vantage can also advance time based on messages from VR-Forces.

# VR-Vantage Capabilities

## Visualization Features

### Lighting Effects

VR-Vantage supports the following lighting effects:

- Dynamic lighting (illustrated).
- Forward+ lighting
- Ocean planar reflection.
- Lens flare.
- Crepuscular (sun) rays.
- Atmospheric scattering.
- Shadows.

VR-Vantage supports light lobes as defined in an OpenFlight model. The light can be cast onto the terrain, and the lights can rotate or move like a search-light or a lighthouse.

Because cast lights are computationally expensive, VR-Vantage will only display the N closest lights to the observer, where N can be configured.

VR-Vantage also supports light points. These are single points that light up in the dark, but do not cast light on objects near it. For example, light points could be used for air port or sea channel navigation. Lights can be colored, can blink, and can be directional.



### Shadows

VR-Vantage can display shadows for entities, props, and terrain features. You can enable and disable them and configure many aspects of shadow quality.

### Shader-Based Effects Texture Maps

VR-Vantage supports several types of shader-based effects texture maps. 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 quality of your simulation without the overhead of high polygon counts.



VR-Vantage supports the following types of shader-based effects 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.

# VR-Vantage Capabilities

## Visualization Features

- **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.

## Depth of Field

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. In [Figure 2](#), 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.



Figure 2. Depth of field

## Visualizing Entity Effects and Interactions

VR-Vantage displays:

- Smoking and flaming effects for damaged entities.
- Trailing effects, such as footprints, wakes, missile trails, and dust clouds.
- Muzzle flashes.
- Tactical smoke.
- Smoke and flame for detonation impacts.

# VR-Vantage Capabilities

## Visualization Features

### Visualizing Sensors

VR-Vantage offers two options for visualizing sensors: CameraFX and SensorFX.

#### CameraFX

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. 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 add the SensorFX Plug-in to VR-Vantage.



No sensor (daylight)

IR sensor

Figure 3. CameraFX

# VR-Vantage Capabilities

## Visualization Features

### SensorFX

SensorFX is an extra cost plug-in for the VR-Vantage 3D applications. SensorFX changes VR-Vantage from a visual scene generator to a sensor scene generator. SensorFX models the physics of light energy as it is reflected and emitted from surfaces in the scene and as it is transmitted through the atmosphere and into a sensing device. SensorFX also models the collection and processing properties of the sensing device to render an accurate electro-optical (EO), night vision or infrared (IR) scene.

Extensive Sensor Coverage SensorFX enables you to credibly simulate any sensor in the 0.3-16.0um band with VR-Vantage, including:

- FLIRs / Thermal Imagers: 3-5 & 8-12um.
- Image Intensifiers / NVGs: 2nd & 3rd Gen.
- EO Cameras: Color CCD, LLTV, BW, SWIR.

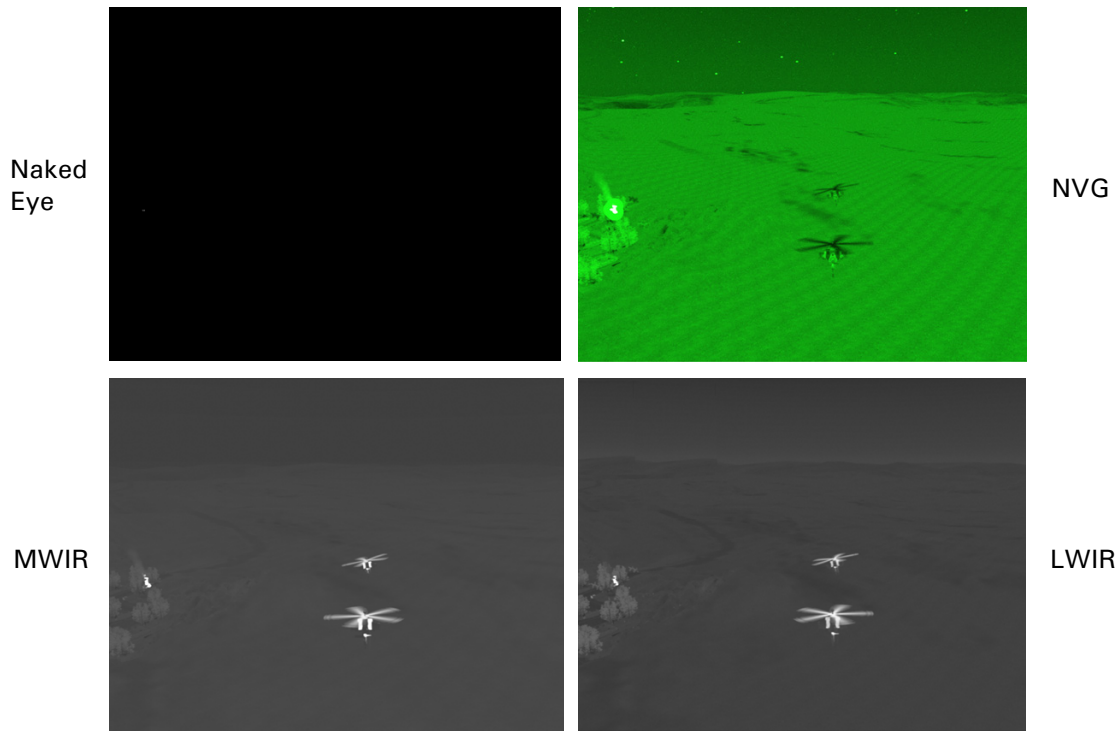


Figure 4. SensorFX

# VR-Vantage Capabilities

## Analysis

### Analysis

VR-Vantage is more than just an IG. It has many features that provide information about the simulation being observed.

### Entity Information

VR-Vantage provides detailed information about the participants in a simulation.

#### Entity Labels

In the 3D projection, you can display entity information on translucent entity labels. You can easily display and hide the labels or pin them to the display. In the 2D projection, entity information is displayed adjacent to entity icons. You can quickly change the information that is displayed to maintain a balance of available information and screen clarity.

#### Track Histories

Track histories allow you to track the paths of entities during an exercise, including the path of munitions as they travel from the shooter to the target. When track histories are enabled, each entity leaves a track ribbon. The ribbon is colored based on the entity's force.

#### Fire and Detonation Lines

When an entity fires a munition, a line is drawn between the entity and the location of the detonation. The line is colored blue, red, or green for the force of the shooter.

#### Emitter Volumes

VR-Vantage can display emitter volumes for entities that have emitter systems (Figure 5). You can configure the colors used to represent emitter frequencies and you can configure the size of the emitter volumes.

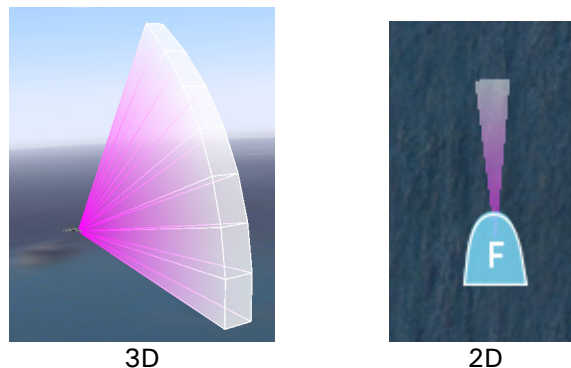


Figure 5. Emitter volumes

# VR-Vantage Capabilities

## Analysis

### XR Mode

XR mode uses the XR model set to enhance your ability to understand the layout of forces in the 3D projection. The 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.

VR-Vantage Stealth also supports scaling of the realistic 3D model set.

In [Figure 6](#), in Stealth observer mode, the nearer fighter is barely visible and the distant fighters not visible at all. In XR mode, model scaling and colorized models show all entities. Height-above-terrain lines provide information about their relationship to the terrain.

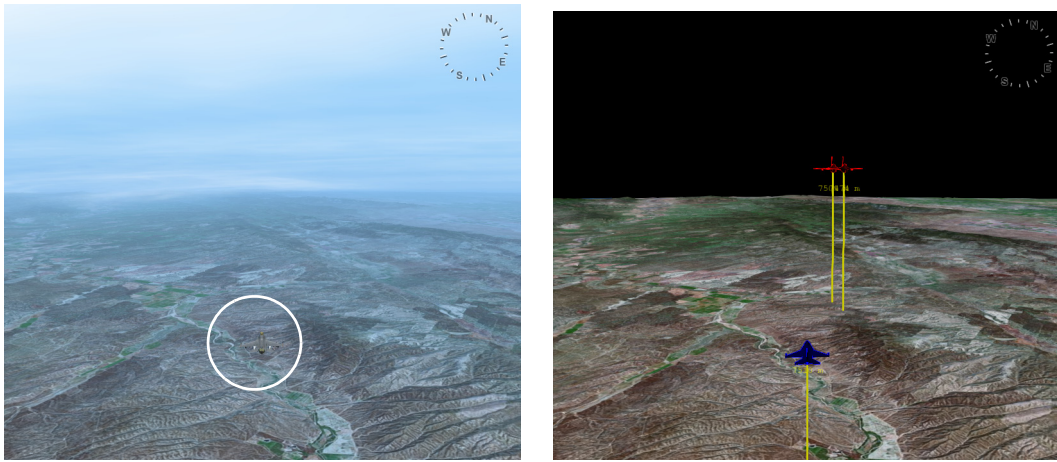


Figure 6. XR observer mode

Terrain scaling complements model scaling by helping to illustrate the relationship of models to the terrain from a great distance.

### Tactical Graphics

VR-Vantage displays tactical graphics (routes, points, areas and so on) published by VR-Forces. It also displays graphics sent using the Remote Graphics API.

### Overlays

Tactical graphics and other visual data is displayed on overlays. You can enable and disable the display of graphics individually and by overlay.

# VR-Vantage Capabilities

## Analysis

### Intervisibility

You can display intervisibility (line-of-sight or LOS) between points, within areas, and between a point and all entities 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 entity.

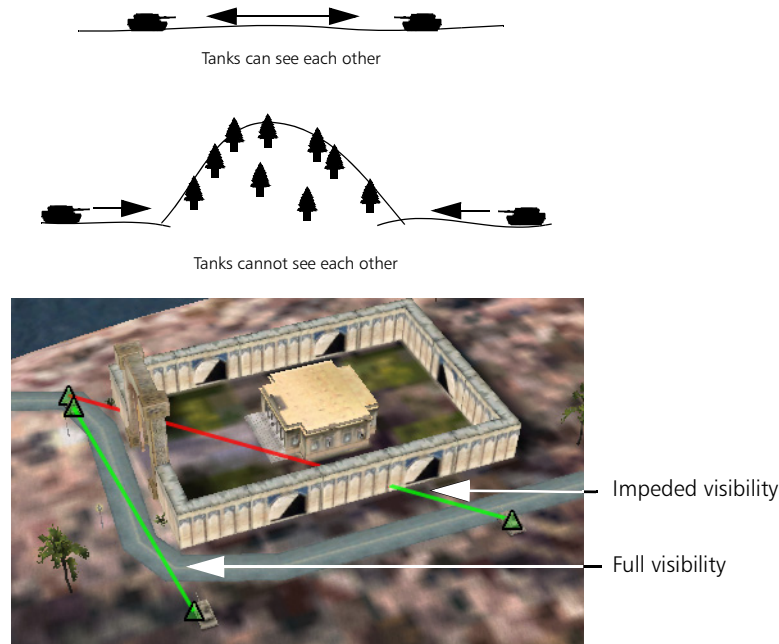


Figure 7. Intervisibility

### Terrain Details

VR-Vantage lets you visualize the terrain in ways that helps you understand it better:

- **Elevation coloring.** Color a terrain by elevation for height cues.
- **Contour lines.** Display contour lines for viewing slope.
- **Wireframe mode.** View the scene in wireframe mode to help understand its structure.

# VR-Vantage Capabilities

## Connectivity

### Connectivity

VR-Vantage can receive data from external simulations and be controlled by them using standard simulation protocols. You can connect and disconnect from simulations during runtime and can configure the connections in the graphical user interface (GUI).

### Support for DIS, HLA, and CIGI

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, and 6.
- 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.

### 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.

### VR-Vantage Application Concepts

This section describes, at a high level, major functional capabilities of VR-Vantage applications. These core capabilities are more than just “features” (which are described later). They are integral to what a VR-Vantage application is and can do. The capabilities are:

- Multi-channel displays.
- The observer.
- Flexible, intuitive graphical user interface (GUI).
- Remote control.
- Configurable startup.

# VR-Vantage Capabilities

## VR-Vantage Application Concepts

### Multi-Channel Display

VR-Vantage can display a simulation in multiple windows, with multiple channels. It can control the display of VR-Vantage displays on remote computers. This gives you the flexibility to create configurations such as wrap-around views for a cockpit simulation, multiple monitor wall displays (security camera displays), or multi-channel displays on large screens. [Figure 8](#) shows how you can create multiple channels in one screen. Each has its own observer and can be independently manipulated.



Figure 8. Multiple channels in one window

[Figure 9](#) shows how you could arrange multiple screens for a wrap-around display,.

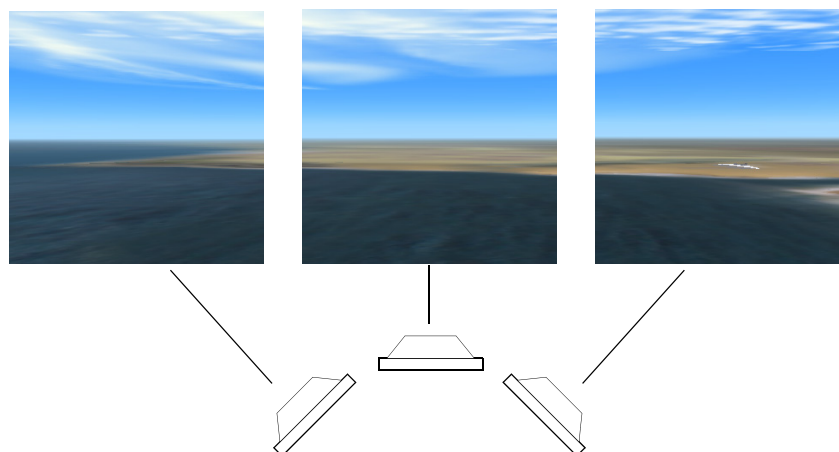


Figure 9. Wrap-around display

[“Inset Views,”](#) on page 5 shows multiple windows on one screen.

# VR-Vantage Capabilities

## VR-Vantage Application Concepts

### 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.
- **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 VR-Vantage 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.

### Support of Stereoscopic Displays

VR-Vantage supports anaglyphic stereo and polarized stereo.

### 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 an entity or prop. When the observer is attached to an entity, it moves automatically with that entity. 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.

### Intuitive Navigation Modes

Control the absolute position of the observer with the keyboard and mouse. Conveniences like speed scaling and terrain constraint make navigation easy. Or attach to various entities for a more focused view.

### 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.

# VR-Vantage Capabilities

---

## VR-Vantage Application Concepts

### Observer Zoom

The zoom feature lets you zoom in on an area of the terrain without moving the observer. It is as if the observer were using binoculars to get a closer look at an object.

### Saved Views

VR-Vantage supports saving an observer's view at any moment. Saved views can be loaded at startup or during runtime.

## 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 screen, 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.

### Persistent Settings

The VR-Vantage GUI remembers the settings as users change them. Settings can be restored back to factory defaults or back to the state they were in when the application was started. Most settings can be exported from one VR-Vantage application and imported into another.

## Remote Control

You can control VR-Vantage remotely through view control messages, which can be generated by MÄK applications such as VR-Forces, and by programs that use the VR-Vantage Control Toolkit.

## Startup Options

On Windows, you can start VR-Vantage applications using any of the usual methods, including the Start menu, desktop icons, and taskbar icons. On both Windows and Linux, you can start VR-Vantage applications from the command line and using batch files or shell scripts. VR-Vantage has many command-line arguments that you can use to customize startup and quickly load preferred configurations.

# VR-Vantage Capabilities

## Performance Features

## Performance Features

VR-Vantage has features that help you understand its performance. It also has features that help you improve performance.

VR-Vantage can display two kinds of performance statistics — OSG statistics, and VR-Vantage statistics. The Function Profiler allows developers to profile a function or portion of a function and view on-screen results.

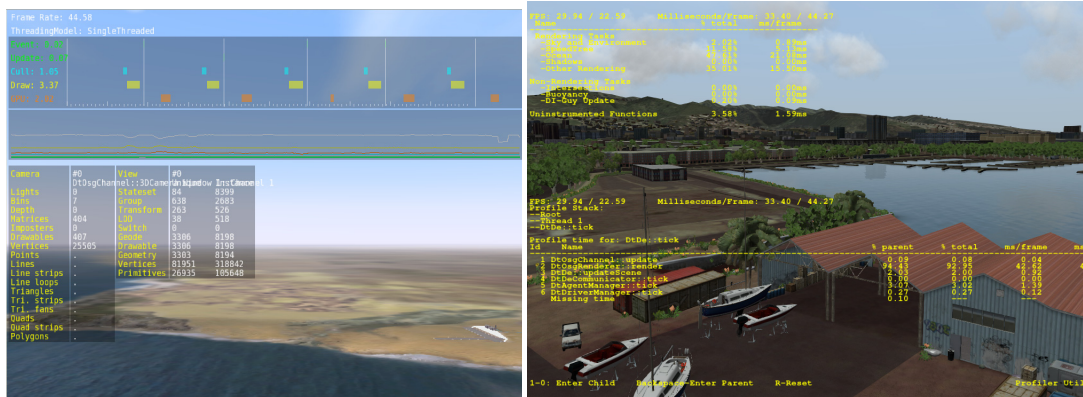


Figure 10. Performance statistics

The StatsHandler class in the osgViewer library can gather and display the following rendering performance information.

- **Frame rate.** `osgViewer` displays the number of frames rendered per second (FPS).
- **Traversal time.** `osgViewer` displays the amount of time spent in each of the event, update, cull, and draw traversals, including a graphical display.
- **Geometry information.** `osgViewer` displays the number of rendered `osg::Drawable` objects, as well as the total number of vertices and primitives processed per frame.

## File Caching

VR-Vantage saves loaded files into a fast loading format so that they are quickly loaded from disk the next time.

## Object Instancing

VR-Vantage keeps instances of objects in memory and clones or references them instead of loading new copies of the same data from disk again.

## Configurable Render Settings

VR-Vantage lets you decide which of the visual features you want to enable and disable, such as advanced lighting, wake and spray effects, shadows, and the ocean height map. This lets you optimize visual quality or performance depending on your needs.

# VR-Vantage Capabilities

---

VR-Vantage Toolkit

## Filtering Entities Using Interest Management

VR-Vantage can use interest management to improve its performance in simulations that have high entity counts. Interest management is an implementation of HLA data distribution management (DDM). When interest management is enabled, VR-Vantage filters out all entities that are more than a specified distance from the observer.

## VR-Vantage Toolkit

The VR-Vantage Toolkit allows developers to design and deploy realtime 2D and 3D visual applications for the simulation and training domain. It is a cross-platform C++ API based on OpenSceneGraph (OSG). Through the VR-Vantage Toolkit, developers can extend existing VR-Vantage applications, create their own applications or embed visuals into existing applications. The VR-Vantage Toolkit also gives you the power to embed any of the VR-Vantage GUI capabilities directly into your simulation applications. Because it is based on OpenSceneGraph, you can leverage value-added plug-ins built by the OSG community and MÄK partners.

## The VR-Vantage Control Toolkit

The VR-Vantage Control Toolkit lets developers send VR-Vantage control messages to VR-Vantage applications. Move the observer, change a setting, or even create graphics on a channel. It can all be done remotely through an HLA or DIS interface.

# VR-Vantage Capabilities

Technical Details

## Technical Details

Table 2 lists supported operating systems and other system requirements.

Table 2: System requirements

|                 | Windows                                                                                   | Linux                                                                                     |
|-----------------|-------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------|
| OS version      | Windows Vista<br>Windows 7, 8                                                             | Red Hat Enterprise Workstation Linux 6                                                    |
| Compiler        | Visual C++ v10.0                                                                          | gcc 4.1                                                                                   |
| Processor       | X86 (32 bit)<br>X86-64 (64 bit)                                                           | X86 (64 bit)                                                                              |
| RAM             | 2 GB minimum; 4-8 preferred                                                               | 2 GB minimum; 4-8 preferred                                                               |
| Hard disk space | 16 GB +                                                                                   | 16 GB +                                                                                   |
| Graphics        | NVIDIA® GeForce 8 series or higher.<br>NVIDIA® Quadro FX: “high end” or “ultra-high-end”. | NVIDIA® GeForce 8 series or higher.<br>NVIDIA® Quadro FX: “high end” or “ultra-high-end”. |
| Shader support  | NVIDIA CG 2.1                                                                             | NVIDIA CG 2.1                                                                             |

## Export Classification

VR-Vantage and all of its modules are classified as EAR 99 by the US Commerce Department.

## VR-Vantage Product Licensing

Each VR-Vantage product is sold separately; however, VR-Vantage Stealth includes a license to run VR-Vantage IG. Please contact your MÄK salesperson for details. MÄK license management uses FLEXlm. Licenses can be floating, node-locked, or dongle-based.

## Localization

The VR-Vantage graphical user interface can be localized using the Qt Linguist tool, which is included with VR-Vantage.

## Coordinate Systems

VR-Vantage supports the Geocentric, UTM, Lambert, Flat Earth, and Cartesian coordinate systems.

# VR-Vantage Capabilities

## Technical Details

### VR-Vantage File Format Support

VR-Vantage supports a large variety of 2D image and 3D model file formats. OSG supplies a Quick-Time plugin for loading movie files, and a plugin for loading font files using the FreeType library.

VR-Vantage supports the following 3D model formats:

- MÄK terrain database (.gdb)
- MÄK encrypted data format (.medf)
- MÄK Stealth format (.mtl)
- MÄK terrain project or terrain descriptor file (.mtp, .mtd)
- OpenFlight (.flt)
- MetaFlight (.mft)
- DTED (.dt0, .dt1, .dt2, .dted)
- CTDB 4, 7, 7L (.c4b, .c7b, .c7l)
- ESRI Shapefile (.shp)
- Terrex Terrapage (.txp)
- 3D Studio (.3ds)
- Alias Wavefront (.obj)
- Carbon Graphics' Geo (.geo)
- OSG (.osg and .ive)
- COLLADA (.dae)
- Quake (.md2)
- NewTek LightWave (.lwo and .lws).

The .osg format is an ASCII text representation of a scene graph that you can edit in a text editor. The .ive format is a binary format, which is optimized for fast loading.

VR-Vantage has several file formats of its own. MEDF is the MAK Encrypted Data Format. MEIF is the MAK Encrypted Image Format. MTF is the MAK Terrain Format. And MSF is the MAK Scene Format.

You can import images in the following formats:

- GEO Tiff (.tif, .tiff)
- JPEG (.jpg, .jpeg)
- PNG
- RGB
- Windows bitmap (.bmp)
- GIF
- ECW.

You can load additional raster formats if you load them in a .earth file using the GDAL driver. For a list of formats supported by GDAL, go to <http://www.gdal.org/> and select the Supported Formats link.

# VR-Vantage Capabilities

## Third-Party Software and Content

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

- SilverLining™: real-time sky, 3D cloud, and ocean rendering 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, Terrex (now Presagis), 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.

---

### Silverlining and Triton (Environment)

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).

### GL Studio (Cockpits)

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

### SpeedTrees (Vegetation)

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 MÄK 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.

# VR-Vantage Capabilities

## 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>)
- The TerraSim\_SampleUrban terrain database comes from TerraSim (<http://www.terrasim.com>)
- The HarrisAfghanVillage terrain database was created using source data from Harris Inc.
- The Berkeley Database was developed by Terrex (now Presagis) (<http://www.presagis.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.

## OpenSceneGraph (3D Rendering)

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).

MÄK will provide modified source code for OSG upon request, as per the OSGPL license. Please contact [support@mak.com](mailto:support@mak.com) to obtain the download links for our modified source.

## osgEarth (Streaming Terrain)

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).

MÄK will provide modified source code for osgEarth upon request, as per the OSGPL license. Please contact [support@mak.com](mailto:support@mak.com) to obtain the download links for our modified source.