# III. Simulation Objects

The chapters in this section explain how to add simulation objects to a scenario, how to select them, and how to display information about them.

# 2. Introduction to Simulation Objects

Simulation objects are the entities and units that VR-Forces simulates as part of a scenario.

## 2.1. Simulation Objects

The players in a simulation are called simulation objects. There are two broad classes of simulation objects:

- Entities represent a discrete individual vehicle or person, such as a tank or a submarine.
- Units represent a hierarchical group of entities, units, or both, such as a platoon or a company.

The concepts described in this section are generally applicable to entity-level modeling and aggregate-level modeling. For details about the differences between the two, see "2.7. Entity-Level Modeling and Aggregate-Level Modeling" on page 10.

Simulation objects can act in and respond to their environment. They execute tasks and they have behaviors. Each simulation object is part of a force and may be a member of an organizational unit.

The VR-Forces application comes with a set of simulation object models that you can instantiate in a simulation. Application developers can create additional simulation objects with the VR-Forces toolkit. You can also create new simulation object types in the Simulation Object Editor. For information about creating new simulation objects, see "Creating and Editing Simulation Models".

## 2.2. How Simulation Objects Are Identified

VR-Forces identifies simulation objects by UUID (universal unique ID), name, echelon ID, object ID, and label. Table 1 summarizes the characteristics of each type of identifier.

**Table 1. Simulation object identifiers**

| Identifier | Characteristics |
|---|---|
| UUID | • Unique. Assigned by VR-Forces.<br>• Persists from exercise to exercise. |
| Name | • Persists from exercise to exercise.<br>• Set by user.<br>• Limits on the length of the character string.<br>• Does not have to be unique. |
| Echelon ID | • Must be unique.<br>• Persists from exercise to exercise.<br>• Assigned by VR-Forces. Changes if an entity is organized into a unit. |

**Table 1. Simulation object identifiers (continued)**

| Identifier | Characteristics |
|---|---|
| Object ID | • Unique.<br>• Does not persist from exercise to exercise.<br>• No user control. |
| Label | • Does not have to be unique.<br>• Persists from exercise to exercise.<br>• Controlled by user.<br>• No limit on character length. |

### 2.2.1 UUIDs

UUIDs are unique identifiers generated by VR-Forces to ensure that every simulation object can be uniquely identified and that the identification persists each time you run the scenario. In particular, the use of UUIDs supports importation of scenarios because while simulation objects in different scenarios might have the same names, they will not have the same UUIDs, so it is possible to import them and not change their names if you do not want to. (The default is to rename duplicate simulation objects.)

### 2.2.2 Simulation Object Names

A name is a text string that, by default, consists of the simulation object's short name, as defined in the object parameter database, and an integer. For example, if you create a new scenario and add two M1A2 tanks and two Apache helicopters, the default names are **M1A2 1**, **M1A2 2**, **AH-64A 1**, and **AH-64A 2**. When VR-Forces generates a name, it picks an integer that makes the name unique. However, if you import a scenario and do not specify that duplicate names be renamed, it is possible that more than one simulation object will have the same name.

You can edit a simulation object name to make it meaningful to you.

Simulation object names have restrictions on their length:

- Entities – 11 characters.
- Units – 31 characters.

If you are using a character set that requires multiple ASCII characters to represent a visible character, entity and unit names will need to be correspondingly shorter.

### 2.2.3 Echelon IDs

An echelon ID is a text string that identifies the simulation object's place in the force hierarchy. For example, if you create an **M1A2 1** in Blue force (or Force 1), the echelon ID would be **1 M1A2, 1 Force**. The syntax of echelon IDs and the details of simulation object organization are described in "2.3. How Simulation Objects Are Organized" on the next page. You cannot edit a simulation object's echelon ID. The echelon ID changes if you change the force hierarchy by adding, removing, or editing simulation objects.

### 2.2.4 Object IDs

Each simulation object has a unique object ID (global object designator). The global object designator is equivalent to the HLA object name in HLA and the DIS entity identifier in DIS. The object ID is used to identify the simulation object internally and for network communications.

### 2.2.5 Labels

A label is a text string that you can use to identify simulation objects without the restrictions of uniqueness or character length that affect the other identifiers.

You can also identify simulation objects using extended labels – additional text strings that you can give descriptive names to. For details, see "5.3. Using Extended Labels" on page 73.

> In the VR-Forces GUI, entity labels refer to informational graphics displayed next to simulation objects. The label described in this section is one of the symbol decorations displayed in Plan View observer mode as part of the entity label. For more information about the different types of labels and information displays, see "5.2. Displaying Entity Labels" on page 65 .

## 2.3. How Simulation Objects Are Organized

All simulation objects simulated by VR-Forces exist in the context of a hierarchy. At the top-most level, simulation objects are grouped according to force ID (friendly, opposing, neutral, and so on). At lower levels, simulation objects are grouped according to military organization, in which a simulation object has a designation in the hierarchy, such as 2nd Platoon, B Company. At each echelon level in the organization, a simulation object has a single superior simulation object. Units contain one or more subordinate entities, units, or both. The hierarchy's lowest level contains an individual entity.

Each simulation object has an echelon ID in the format:

```
designator [category] [echelon], designator [category] [echelon],
  ..., force_number Force
```

where:

- *designator* is a number or letter that differentiates the simulation object from other members of the echelon, such as 1 (M1A1), 2 (M1A2), and so on.
- *category* is an optional descriptor for the simulation object, such as M1A2.
- *echelon* is an optional descriptor for the organizational unit, such as "Platoon".
- *force_number* is the force the simulation object belongs to, based on the force type enumerations. The force is the highest level in the hierarchy.

When you create a new simulation object, it exists only within the default force. Therefore, its echelon ID would be similar to:

```
1 M1A2, 1 Force
```

If a simulation object becomes part of a unit, the unit information becomes part of the echelon ID, and the designator may change, for example:

```
2 M1A2, 2 Plt, 1 Force
```

### 2.3.1 How Echelon IDs Are Assigned

The short name and echelon labels that make up an echelon ID, such as "M1A2", "Plt", and so on, are specified in the VR-Forces object parameter database and are associated with object-types. When you create a simulation object of a certain type, the short name and echelon name associated with that type are assigned to the simulation object. You cannot change the labels interactively. You can change the labels by editing the object parameter database.

Designators are assigned by VR-Forces. You have no direct control over them. When you create new simulation objects, they are assigned designators sequentially, starting with one (1). If you aggregate a group of entities, the unit leader is assigned the designator 1, and the other entities in the unit are assigned successive designator numbers.

## 2.4. How Simulation Objects Communicate

VR-Forces includes a default communication model called simple-radio-comm-model. Using this simple communication model, radio messages always reach their intended destination (as long as the destination is on a reachable network), and messages are all passed instantly. The simple radio communication model has one parameter, which determines how the radio networks are set up. There are three connection-modes available:

- **aggregate** mode specifies that each unit has its own radio network. Messages sent from a simulation object can reach any simulation object that is in the same organization, at the same level (the simulation object's peers in the organization), and can also reach the simulation object's direct superior. All other simulation objects are outside the network and are unable to receive messages sent by the simulation object. This is the default mode.
- **force** mode specifies that there is a radio network for each force in VR-Forces. Messages sent from a simulation object can reach any other simulation object of the same force.
- **all** mode specifies that all simulation objects in VR-Forces are part of one large radio network. In this case, any simulation object can receive messages sent by any other simulation object.

The communication model is configured in *./data/simulationModelSets/<sms>/commModelParams.mtl*. For details about configuring radios, see "Configuring Radios".

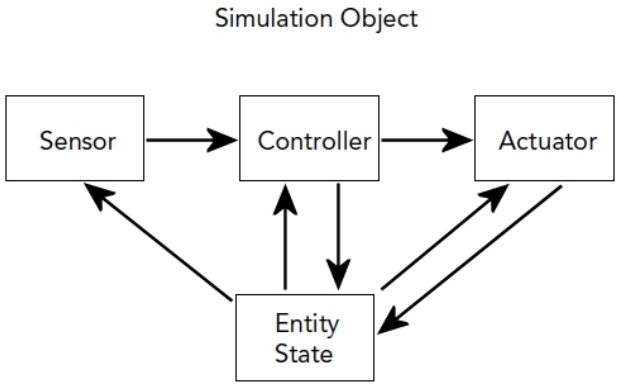
### 2.4.1 Custom Communication Models

You can create new communication models and install them in VR-Forces. This is useful if you want to define your own connectivity model or write your own algorithms for determining whether radio messages are delivered. For details, please see the API documentation.

## 2.5. VR-Forces Uses a Component Architecture

VR-Forces simulation objects use a component architecture similar to that used in many robotics applications. There are three basic types of components supported: sensors, controllers, and actuators. Components communicate with each other through data ports and port groups, which can provide data as simple as a single number (such as a throttle value) or more complex data such as a list of simulation objects that have been detected. The port groups make it possible to multiplex and prioritize multiple data sources to a single data recipient. For example, the automotive actuator of ground vehicles can receive inputs from a collision-avoidance controller, move-to-point controller, and follow-route controller, but it does not want inputs from all of those controllers at once.

**Figure 1. Simulation object component architecture**




The component architecture supports any number of sensors, controllers, and actuators in a simulation object. Components are specified in the object parameter database by using component descriptors. Component descriptors indicate the type of component that should be created. They can also provide parameter information to a component.

For more information, see "Object Parameters".

### 2.5.1 Sensors

Sensor components provide models of the simulated environment, which are used by controller components to make decisions and perform tasks. The simplest sensor might provide (simulated) ground truth, while more sophisticated ones could use complex models for sensing electromagnetic emissions, RADAR, or a cognitive model to simulate a soldier or crew member's perception of the simulated world. Sensor components can get information from:

- The virtual network.
- From a terrain database.
- By monitoring the state of the simulation object model.
- Many other potential sources.

The use of sensors is optional.

### 2.5.2 Controllers

Controller components typically provide control inputs to actuator components (although this is not required). Controllers can use information provided by sensors to perform their functions, but this is not a requirement either. For example, the controller that implements the various forms of the Wait task neither requires sensor information nor provides control inputs to any actuators.

### 2.5.3 Actuators

Actuator components:

- Provide the physical model of the simulation object being simulated.
- Change the simulation object's state.
- Interact with other simulation objects (such as by sending messages).

Actuators can use control inputs provided by controller components as parameters to its model each simulation frame. Actuators may:

- Send messages on the radio system.
- Generate events on the virtual network, such as detonations.
- Modify the state of the simulation object: position, velocity, damage, and so on.

Actuators can also use data directly from a sensor, when modeled or filtered data is preferred over simulated ground truth for a model that does not require a controller.

## 2.6. Simulation Object Behaviors and Tasks

A behavior is something that a simulation object knows how to do on its own. Behaviors describe how a simulation object reacts to a specific condition or situation or how it performs a task. A task is something that you tell a simulation object to do. A simulation object does not innately know that it should move to a point. However, if you task it to move to a point, it knows how to locate the point and how to move to it.

Behaviors can override task assignments. For example, if an entity is moving along a route and detects another entity in its path, the collision-avoidance behavior overrides the movement task until the entity successfully avoids the obstacle and can return to moving along the route.

VR-Forces has behavioral models for the included simulation object types. These models simulate the behaviors you would expect from live participants and objects functioning in the real world. You can change the parameters for the behavior models. With the VR-Forces toolkit, you can change the models themselves and add additional models. For more information, please see the *VR-Forces Developers Guide*.

- No semi-automated forces (SAF) system, such as VR-Forces, can completely emulate the behavior of a complex real-world entity. The behaviors provided by VR-Forces handle the lower-level activities necessary to complete a task and allow you to focus on the higher level activities involved in assigning tasks and preparing plans.
- If you do not want a simulation object to engage in any independent intelligent behaviors, you can disable them with the AI Enabled set data request. For details, see "AI Enabled".
- You can also write behaviors for units. For details, see "Writing Advanced Scripts with the Behavior Engine".

## 2.7. Entity-Level Modeling and Aggregate-Level Modeling

VR-Forces supports two different approaches to modeling simulation objects: entity-level simulation and aggregate-level simulation.

In entity-level simulations, VR-Forces models the interaction among individual entities, such as ships, tanks, automobiles, airplanes, and people. When you create a simulation object, you are usually creating one tank, one ship, one human, or one of another object. Combat takes place between entities based on their weapon systems and their ability to withstand munitions.

In an entity-level simulation, you can combine entities into higher echelon units, such as platoons and companies, but when they engage in combat, they do so by modeling the interactions of their members at the individual entity level.

Aggregate-level simulations focus on the high-level movement and interaction of large numbers of personnel and equipment in hierarchical units. For the most part, they do not model individual simulation object interactions and combat. When you create a simulation object, you are usually creating a company, a brigade, or another object representing an entire group. Aggregate-level combat is modeled on the relative strengths and weaknesses of simulation objects in personnel, equipment, and location, not on the engagement of individual entities.

As in entity-level simulations, in an aggregate-level simulation, you can combine entities and units into higher echelon units.

The type of modeling used in a scenario depends on the systems that simulation objects use, which are configured in the simulation model set it uses. If you use *EntityLevel.sms* or an SMS similar to it, it uses entity-level modeling. If you use *AggregateLevel.sms*, or an SMS similar to it, it uses aggregate-level modeling. Therefore, when you create a scenario, you must consider which type of modeling you want to use and you must remember which type of modeling you are using when you populate the scenario.

Aggregate-level modeling only works with HLA Evolved and MAK FOM extensions. (If you try to create a scenario with *AggregateLevel.sms* and you are using an unsupported protocol, VR-Forces displays an error message.) For a detailed discussion of the aggregate warfare model, see "How Aggregate-Level Modeling Works".

Aggregate-level SMSs can have simulation objects that represent individual entities. However, they use data-driven models that are different from those used by similar entities in *EntityLevel.sms*.

Although it is possible to create a scenario that uses both types of simulation model sets, the resulting behavior will be unpredictable because simulation objects that use aggregate-level modeling cannot interact with simulation objects that use entity-level modeling. Combining these SMSs is not supported and not recommended.

## 2.8. Collision, Obstacle, and Feature Avoidance

By default, in entity-level scenarios, simulation objects avoid other simulation objects, features, and obstacle control objects. If, while carrying out a task, a ground, lifeform, or rotary-wing entity detects that it is on course to collide with another entity, feature, or obstacle, it alters its route to avoid the collision. Then it returns to its task.

- When a unit moves in entity-level scenarios, the individual members of the unit use collision avoidance.
- In aggregate-level scenarios, there is no collision avoidance between simulation objects. If a unit encounters an obstacle, it stops, sends a message to the console, and awaits further instructions.

When an entity avoids an obstacle, it does not necessarily follow the outline of the obstacle. Instead, it avoids the obstacle's bounding volume, which is conceptually the smallest rectangle that bounds all of the points in the obstacle.

Ground entities avoid features, such as buildings and plants. By default, entities avoid other entities (some cultural features are created as entities), buildings, water, and vegetation. (LCACs do not avoid vegetation by default.)

Collision avoidance has these qualifications:

- To specify that ground entities should not move through the terrain (for example, through a wall or through a hill), set the **check-terrain-collisions** parameter in the automotive actuator component (powertrain) to True. For performance reasons, it defaults to False, so that entities that do not care about terrain collisions do not perform the check.
- If an entity is tasked to move to a point that is inside an obstruction, or very close to one, it might never reach the point, but it will continue to try to reach the point as long as the task is in effect. This behavior also applies to entities following routes that have vertices inside an obstruction.
- If a terrain has geometric features as well as point vector features, the feature avoidance may not work because the points are placed on the terrain on "top" of the existing geometry, which means they are on top of the geometric feature.
- Do not place entities inside obstacles. Their behavior will be unpredictable.

- Although you can assign a force to an obstacle when you create it, it has no effect on entity behavior. Entities avoid all obstacles.

You can configure the objects that an entity will avoid by editing its entry in the object parameter database. For details about configuring obstacle avoidance, see "Configuring Obstruction Avoidance".

## 2.9. Entity Movement and Soil Type

In VR-Forces, the movement of simulation objects is affected by the surfaces on which they are moving. The speed of movement over a given surface is configured in the **soil-list** block in a simulation object's object parameter database entry or in its movement system definition file.

Table 2 lists the soil types supported for terrain databases and how they map to the roughness types you can configure in the **soil-list** block for VR-Forces simulation objects.

**Table 2. Soil type/roughness type mappings**

| Soil type | Roughness type |
|---|---|
| ocean | deep-water |
| deepLake | deep-water |
| shallowLake | shallow-water |
| deepRiver | deep-water |
| shallowRiver | shallow-water |
| shallowPond | shallow-water |
| shallowStream | shallow-water |
| dryGround | hard-packed |
| pavedRoad | paved-road |
| gravelRoad | gravel |
| dirtRoad | gravel |
| asphalt | hard-packed |
| USRailroad | hard-packed |
| softSoil | sand |
| swamp | muck |
| mud | muck |
| forest | hard-packed |
| grass | hard-packed |
| cultivatedFields | hard-packed |
| orchards | hard-packed |
| rock | rocks |

**Table 2. Soil type/roughness type mappings (continued)**

| Soil type | Roughness type |
|---|---|
| boulder | rocks |
| sand | sand |
| building | hard-packed |
| tree | hard-packed |

You can change the way a ground vehicle moves on each soil roughness type by editing the appropriate movement system definition file in *./data/simulationModelSets/EntityLevel/vrfSim/systems/movement*:

- *ground-tracked.sysdef*
- *ground-tracks-amphibious.sysdef*
- *ground-wheels-off-road.sysdef*
- *ground-wheels-road.sysdef*

Near the end of each movement sysdef file, inside the **meta-data-entry-list** block, there is a **soil-list** block that lists the soil roughness types (paved-road, hard-packed, gravel, and so on). Here you can adjust the **acceleration-factor** and **stopping-factor** values for each roughness type; VR-Forces uses those values to calculate how fast a vehicle can accelerate, brake, and turn on that type of surface. (Do not modify the **max-speed-factor** value; it is not used and changes have no effect.)

The **acceleration-factor** affects the surface's drag on a vehicle. Use values ranging from 0 to 1, where lower values increase drag. A value of 1.0 means the surface exerts no drag (a paved road, for example); a value of 0.0 means the surface's drag prevents the vehicle from moving at all (deep water).

The **stopping-factor** affects the traction the vehicle has on the surface. Use values ranging from 0 to 1, where lower values reduce traction. A value of 1.0 means the surface provides maximum traction (a paved road, for example); a value of 0.0 means the surface provides no traction (deep water).

For example, to improve a road vehicle's ability to accelerate in snow, you might increase the **acceleration-factor** in the snow section from

```
(meta-data-entry-list
 (soil-factors
  (soil-list)
   ...
   (snow
    (acceleration-factor 0.050000) ; for deep snow
```

to

```
   (snow
    (acceleration-factor 0.080000) ; for deep snow
```

In addition, you can modify these parameters to control the way weather affects ground vehicle movement over different soils:

- **deep-snow-depth**:
  Depth of snow for full mobility degradation effect; in meters. Default is 1.0.
- **saturation-water-depth**:
  Rain accumulation for full wet effect; in meters. Default is 0.2.
- **water-slide-depth**:
  Depth of accumulated rain that makes a hard surface slippery; in meters. Default is 0.02.
- **water-slide-factor**:
  Fraction of traction retained on a hard, wet surface. Choose a value in the range 0–1. Default is 0.65.
- **hard-freeze-temperature**:
  Temperature at which shallow water freezes enough to support a vehicle; in Celsius. Default is -5.0.

You can find the parameters in *./data/simulationModelSets/EntityLevel/physicalWorldParams.mtl*.

### 2.9.1 Ground Vehicle Movement on Slopes

There is a "cost" associated with ground vehicles moving up a slope. VR-Forces bases its calculations of what is too steep on the slope. If it is above the maximum slope that a ground vehicle is able to scale, that vehicle does not attempt it. To adjust the factor, edit the **slope-avoidance-factor** parameter in the appropriate ground movement *.sysdef* files in *./data/simulationModelSets/EntityLevel/vrfSim/systems/movement*.

# 3. Creating and Placing Objects

This chapter describes generic procedures for creating and placing simulation objects, tactical graphics, and simulation effect objects. The later chapters assume that you are familiar with these procedures.

## 3.1. Creating Objects

VR-Forces lets you create entities, tactical graphics, simulation effect objects (terrain, weather, clutter, and so on), cultural features, and props. You can create most of these from the Create menu or the Create Object Palette with the exception of props, which you create only from the Props Palette. The procedure for creating an object is essentially the same regardless of the type of object:

1. Select the object that you want to create.
2. Specify the location and properties of the object.

However, there are several ways to select the object you want to create and several ways to specify an object's location and properties. This chapter describes the different generic ways to create objects. Additional procedures specific to different object types are described in later chapters.

When you create an object, VR-Forces adds a tab to the right side of the window, below the object creation palette(s). The tab is labeled Create *object*, where *object* is whatever object you selected to create. If you click that tab, it opens a dialog box where you can specify the properties of the object you are creating. Table 3 lists the properties that you can set for each object type. You can also edit these same properties after you create an object.

**Table 3. Editable object properties**

| | Entity | Tactical Graphic | Cultural Feature | Simulation Effect Object | Prop |
|---|---|---|---|---|---|
| Altitude | X (specific entities) | X | | X (specific objects) | |
| Force | X | X | X | X (specific objects) | |
| Heading | X | | X | | |

**Table 3. Editable object properties (continued)**

| | Entity | Tactical Graphic | Cultural Feature | Simulation Effect Object | Prop |
|---|---|---|---|---|---|
| Label (and extended labels) | X | X | X | X | |
| Location | X | X | X | X | X |
| Name | X | X | X | X | |
| Overlay | X (Edit only) | X | X (Edit only) | X | |
| Publish Object | | X | | X (specific objects) | |
| Style | | X | | | |
| Visual Markings | X (specific entities) | | | | |

## 3.2. Selecting the Object to Create

You can select the object to create on the Create menu or on an object palette (Create Object Palette or Props Palette). Once you have selected the object, placing the object is the same regardless of how you started the process. For details, see "3.3. Placing Objects" on page 26.

The Create menu does not contain the full list of objects configured in VR-Forces. It only lists the simulation objects and tactical graphics that you have marked as "Favorites". Marking favorite objects allows you to quickly access those objects you use most frequently. For details, see "3.17. Adding an Object to the Favorites List" on page 44.

### 3.2.1 Selecting the Object from the Create Object Palette

VR-Forces has palettes that you can use to quickly choose the object that you want to create:

- Create Object Palette. You can create many different objects from this palette:
  - Simulation objects. Entities and cultural features
  - Tactical graphics. Control objects, lines, circles, points, text, volumetric lines and areas.
  - Simulation effect objects. Terrain, weather objects, clutter areas, various types of hazards, obstacles, and pedestrian-oriented objects. (Some objects are aggregate-level only.)
- Props Palette. Create props that can be saved as part of the terrain. (The default window layout does not display the Props Palette. You can open it from the View menu.)

Figure 2 shows the location of the palettes (with the Props Palette enabled).

**Figure 2. Object creation palette tabs**

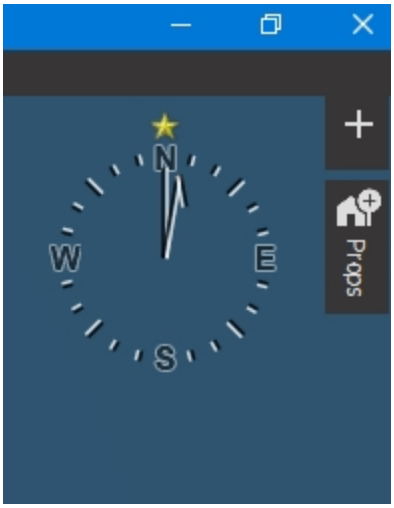
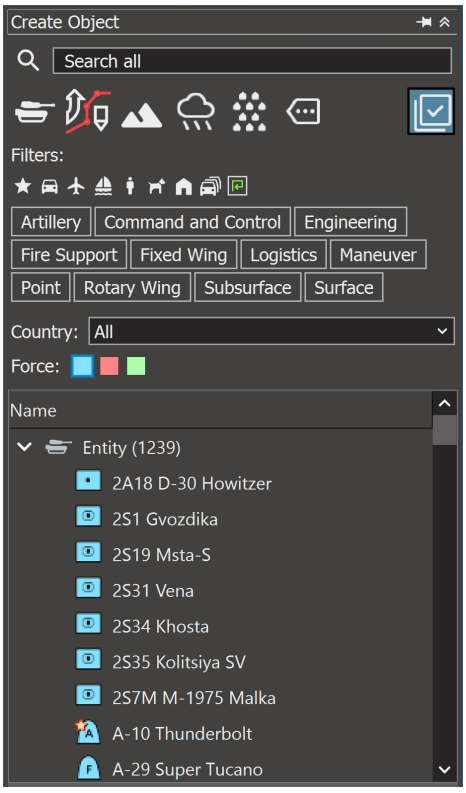



Figure 3 shows the Create Object Palette when it is open.

**Figure 3. Create Object Palette**




The objects in the palette are identified by their 2D icon and their name. You can use images other than the default icons to identify objects. For details, see "Using an Alternative Icon List".

➤ **To select the object to create on the Create Object Palette:**

1. If you are using multiple simulation engines, in the Selected Simulation Engine Toolbar select the simulation engine on which you want to simulate the object. You can also use Automatic (the default option) to have VR-Forces add the simulation object to the least burdened sim. engine.
2. Click the tab to open the palette (Figure 3).

   The palette lists all objects that can be created. There is no preconceived force for any of them. You can filter the palette list by type of object (entity, tactical graphic, terrain, weather, clutter, or simulation object), text string, tag, and country of use. (The Props Palette organizes props in a tree list.)
3. On the Force line, click the button that represents the force for which you want to create this object.
4. Optionally, filter the list of objects. (For information about how to filter object creation palettes, see "3.2.3 Filtering the Create Object Palette" on the next page.)
5. Optionally, filter the list of entities by selecting a country in the Country list.

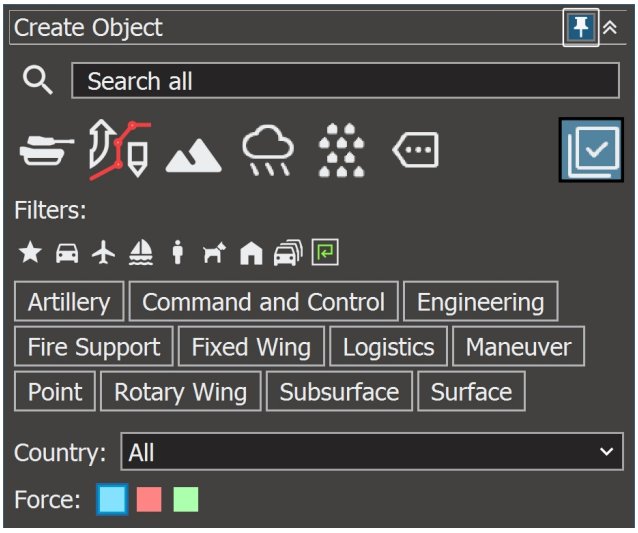
   > ⓘ The Country filter lists simulation objects of the selected tag that are used by the specified country. VR-Forces does not provide an exhaustive list of countries or the platforms that they use. You can add additional countries and categorize simulation objects in the Simulation Object Editor. For details, see "Editing a Simulation Object's Country List".
6. Optionally, enter a search string to further filter the list.
7. In the list of objects, select the object that you want to create. Information about that object displays at the bottom of the palette. The cursor changes to draw mode and a Create *object* tab is added to the window. For details about draw mode, see "3.2.5 Draw Mode" on page 24.
8. Create the object, as described in "3.3. Placing Objects" on page 26.

### Pinning the Object Palette to the Window

Ordinarily, after you select an object to create, the palette closes. However, you can pin the palette to keep it open so that you can select additional objects to create without needing to click the palette to reopen it. If you pin a palette, it becomes dockable and undockable and you can drag it off the window like the other panels. (For details about undocking panels, see "Dockable Panels and Toolbars".)

➤ To pin a palette so that it stays open after you select an object, click the pin button (📌) in the palette's title bar (Figure 4).

**Figure 4. Palette pin**




## 3.2.2 Selecting the Object to Create on the Create Menu

Waypoints, routes, areas, and objects marked as Favorites are available on the Create menu.

➤ **To select the object to create on the Create menu:**

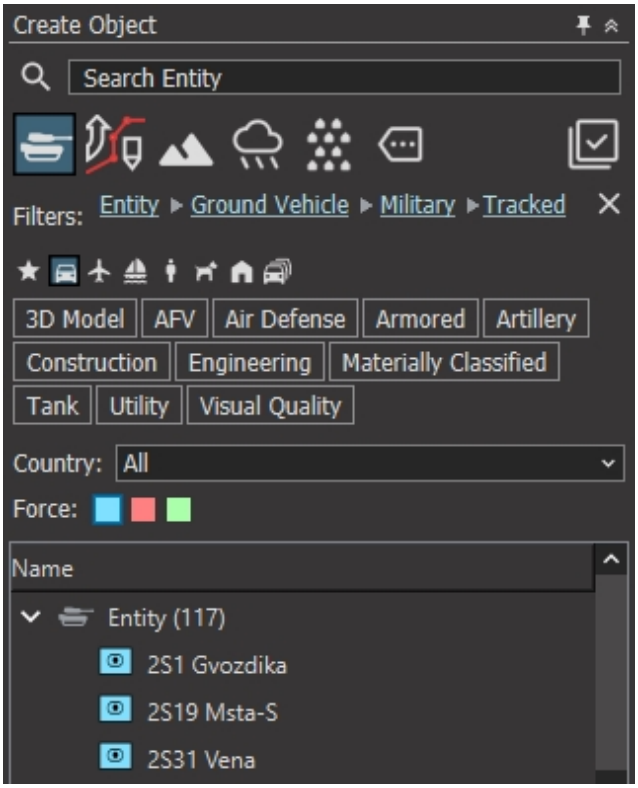
1. If you are using multiple simulation engines, in the Selected Simulation Engine Toolbar, select the simulation engine on which you want to simulate the object. You can also use Automatic (the default option) to have VR-Forces add the simulation object to the least burdened sim. engine.
2. To create an object or graphic, choose **Create >** *type* **>** *object type*, where:
   - *type* is a category of object, such as Entity or Tactical Graphic.
   - *object type* is a specific object within the category, such as M1A2 SEP Abrams or Weather (Area).

   The cursor changes to draw mode and a Create *object* tab is added to the window. For details about draw mode, see "3.2.5 Draw Mode" on page 24.
3. Create the object, as described in "3.3. Placing Objects" on page 26.

## 3.2.3 Filtering the Create Object Palette

You can filter the Create Object Palette to make it easier to find the object that you want to create. You can filter objects by type, tags, country of use, and text strings.

**Figure 5. Create Object palette filtering**




Types are the kinds of objects you can create: entities, tactical graphics, and simulation effect objects: terrain, weather, clutter, and others. When you select a type, the Tag window updates to show only the tags that apply to objects of that type. The object list is also filtered, listing only objects of that type.

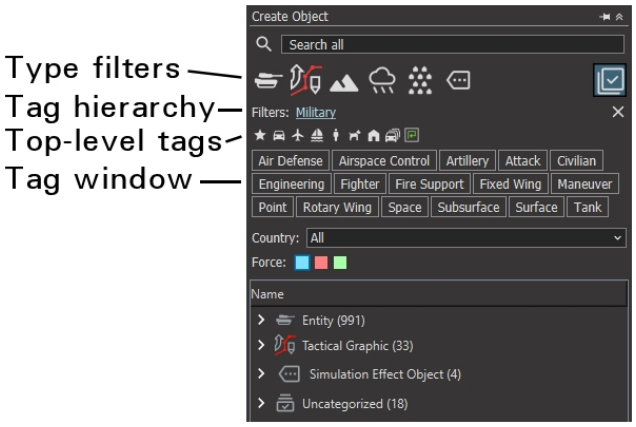
Tags are categories, such as land, air, military, civilian, and so on.

Many object types are used by multiple countries. If you want to see the objects used by a particular country, you can select the country of interest. (This filter does not apply to tactical graphics.)

Using the Search field, you can enter a text string to search for object names. As you type, objects that contain that string are listed. The Tag window updates to show only the tags that apply to the listed objects.

Simulation objects are assigned to tags and countries in the Simulation Object Editor. For details, see "The Simulation Object Editor and SMSs" and "Editing Common Parameters".

**Figure 6. Create Object Palette filters**




➤ **To filter the objects listed in the Create Object palette:**

1. To filter by type, select the type you want to filter by. For details, see "Filtering the Create Object Palette by Type" below.
2. To filter by tags, select the tags you want to filter by. For details, see "Filtering the Create Object Palette Using Tags" on the facing page.
3. To filter by country, select a country in the Country list.
4. To filter by text, enter a text string in the Search box.

### Filtering the Create Object Palette by Type

All objects have a particular type:

- Entity (). Ground vehicles, aircraft, surface and subsurface vehicles, humans, animals, cultural features, and units.
- Tactical Graphic (). Lines, circles, points, text, volumetric lines and areas, and some standard military tactical graphics.
- Simulation effect objects. These are objects that can affect the simulation. There are four types of simulation effect objects:
  - Terrain (). These objects make changes to the terrain. Some can change the appearance of the ground in dynamic terrain, while others can affect how human characters navigate the terrain.
  - Weather (). Local weather zones.
  - Clutter (). Areas and points that simulate pattern of life.
  - Simulation Effect Object (). Other simulation effect objects that are not one of the other three types.

If you want to search for simulation objects that are not in a category, right-click the type button and choose Not. The type displays with a not icon on the Filters line.

To clear the type filter, click All ().

### Filtering the Create Object Palette Using Tags

All objects are categorized using tags. An object can have multiple tags. For example, a ground vehicle can have the Ground Vehicle tag, but it can also have other tags such as Military, Tracked, Materially Classified, and so on. You can filter for any of these tags.

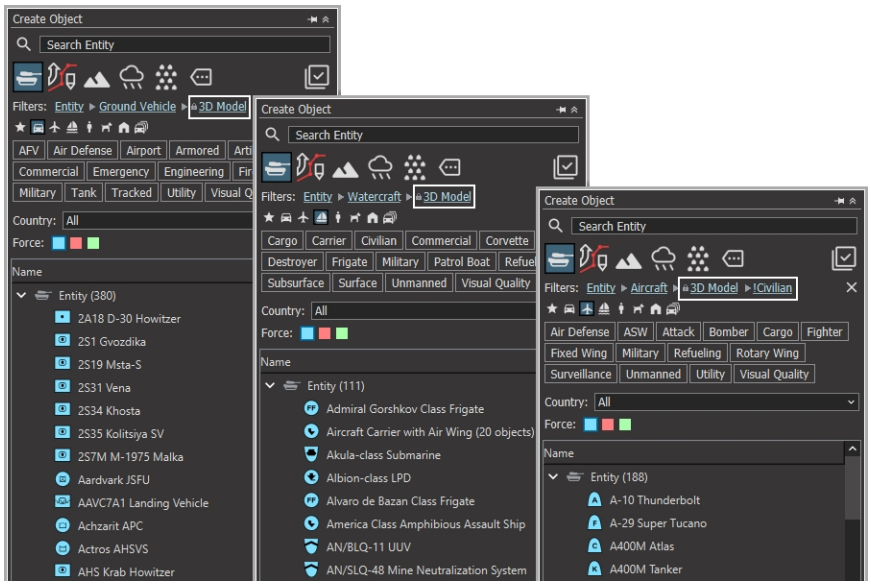
Tags are listed in the Filters section of an object creation palette. The top row of tags (represented by buttons) is static during any VR-Forces session (Figure 6). It contains the highest ranking tags. As you select tags, the Filters: line lists the tags you have selected from most inclusive to least inclusive. You can clear any level of filtering by clicking the label to its left.

The Tag window lists the tags that apply to simulation objects in the objects list that have not yet been filtered out. This list redisplays as you select additional tags. (If the number of tags that could be listed in the Tag window exceeds the space available, VR-Forces displays tags based on frequency of use and ranking. For details about ranking tags, see "Adding a New Tag".)

VR-Forces remembers the tags selected for each top level tag. So if you set up some tags for Ground Vehicle and then filter using the Watercraft tag, if you go back to the Ground Vehicle tag, the tags you selected previously are still selected.

You can make tags "sticky". A sticky tag applies to all top level tags. For example, suppose you only want to see simulation objects that have 3D models, regardless of the platform. You could make 3D Models a sticky tag. Then no matter which top level tag you selected, it would be filtered for the 3D Models tag. Sticky tags displays with a lock icon on the Filters line (Figure 7). You can also set a filter to look for simulation objects that are not in a category. Not tags display with a no circle icon on the Filters line.

**Figure 7. Sticky and Not tags**



➤ **To filter a list using tags:**

1. Click a top level tag, or All.
2. Select additional tags in the Tag window to narrow the filtering. As you select tags, the object list redisplays and the tag list redisplays to show the tags that apply to the remaining members of the object list. As you select additional tags, the hierarchy of tags on the Filters line updates.
3. If you always want to use a certain tag, such as 3D Model, or Materially Classified, right click the tag button and choose Sticky. The tag is displayed with a lock icon on the Filters line. It applies to all top level tags.
4. If you want to search for simulation objects that are not in a category, right-click the tag button and choose Not. The tag is displayed with a not icon on the Filters line. It applies to all top-level tags.
5. To clear tags, in the breadcrumb path on the Filters line, click the tag to the left of the tag you want to clear. To clear all tags, click the X button.

### 3.2.4 Previewing Object Information

The Create Object Palette provides a preview of the selected object.

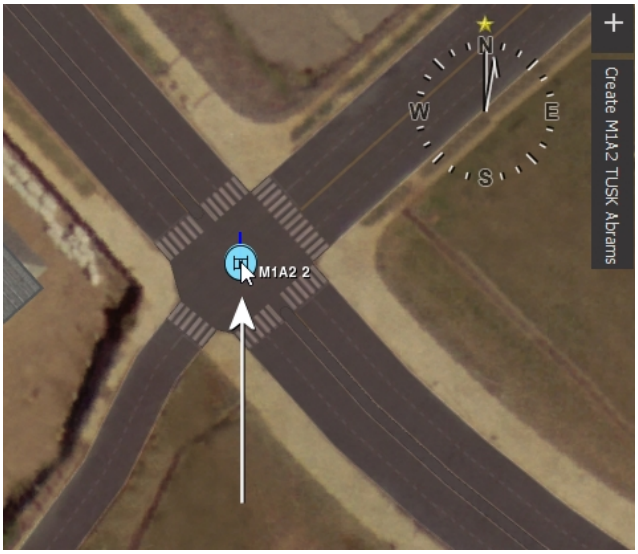
- Entities display an image of the 3D model or, if there is no 3D model, the 2D icon. It also shows the tags defined for that entity.
- Tactical graphics display the tags, and some also display a summary of the tactical graphic's functionality.
- Simulation effect objects (terrain, weather, clutter, and so on) display a summary of the effect, and any tags defined for the object.

If you click one of the tags for the selected object, that is immediately applied as a filter to the objects list in the Create Object Palette.

### 3.2.5 Draw Mode

When you create an object, the cursor changes to draw mode. If you are creating a simulation object or a cultural feature in the 2D view, the cursor changes to a 2D icon that represents the chosen simulation object (Figure 8).

**Figure 8. 2D entity creation and Create *object* tab**




In the 3D view, the cursor changes to the 3D model for the chosen entity (Figure 9). If you are zoomed out, 3D models may be hard to see. If you are creating multiple human characters of a type that randomizes the appearance of the character that will be created, such as Civilian Male, after each character you create, the model attached to the cursor changes to show you the appearance of the next character appearance that was randomly selected.

**Figure 9. 3D entity creation**

Top-down view | Ground-level view

When you create a tactical graphic, the cursor displays the waypoint symbol when you create a point and a vertex symbol when you create any type of line or area. Figure 10 illustrates tactical graphics draw mode cursors.

**Figure 10. Tactical graphics cursors in draw mode**


When you create a prop, the cursor changes to show the model for the prop. In the 2D view, the model is a top down view and is proportional to the terrain, so it might be difficult to see if you are zoomed out (depending on the size of the prop).

## 3.3. Placing Objects

After you select the object that you want to create, the Create *object* tab is added to the right side of the window below the palette(s).

There are two basic methods for placing simulation objects:

- Click to Create. Each mouse click creates an instance of the selected object. For multi-vertex tactical graphics, each click specifies a vertex. Click to Create is best for rapidly creating objects using the default values for the object's properties.

  The create object graphic is attached to the mouse cursor. Although you can use the Create *object* dialog box to specify an object's name or other properties, as you move the cursor into the dialog box, the creation location (as represented by the object graphic) moves with it, which can be disconcerting. You can set heading and altitude dynamically, so there is no need to use the Create *object* dialog box for these properties.

- Click to Locate. Mouse clicks specify the coordinates, but do not cause the simulation object to be created. The create object graphic is placed at the specified coordinates, but is not attached to the mouse cursor.

  If you want to explicitly set the object's location, and particularly if you want to set the altitude relative to sea level, you must use this method. This method may also be more comfortable for persons who want to take a deliberative approach to specifying an object's properties before creating it.

Once you have clicked to place a simulation object, an icon indicates the location where it will be created as a placeholder. This tells you that your click was successful while VR-Forces creates the object. The indicator at the bottom right of the main window also displays the process of adding the object.

### 3.3.1 Placing an Object Using Default Values (Click to Create)

Although you can specify an object's properties using the Click to Create method, it is best suited for quickly creating objects using default properties. Objects are placed on the default overlay for the object type.

> ⓘ You can use this method to automatically embark simulation objects and attach tactical graphics. For details, see "6.1. Embarking and Disembarking Simulation Objects" on page 86 and "Attaching a Tactical Graphic Automatically".

➤ **To place an object with each mouse click:**

1. Select the object that you want to create as described in "3.2. Selecting the Object to Create" on page 17. A Create *object* tab is added to the side of the window below the palettes.
2. For simulation objects and waypoints, click on the terrain for each object you want to place. You can create multiple single-point objects without exiting create mode. For details about disabling this create mode, see "3.7. Locking the Mouse to the Object Being Created" on page 33.

   For multi-vertex tactical graphics, click on the terrain to place each vertex. Double-click to place the last vertex in a route or area. (Do not try to close areas by clicking on the starting point. Doing so only creates another vertex near the first one.)

   To draw a circles, ellipses, boxes, or text, see the sections in "Creating and Editing Tactical Graphics".

   > ⚠ 
   > - If new objects or vertices are not displayed as you click, it is possible that Attach Object to Mouse is disabled. To restore Click to Create mode:
   >   a. Click the Create *object* tab. The Create *object* dialog box opens.
   >   b. Select the Attach Object to Mouse check box.
   > - If tactical graphics are not displayed after you create them, check to see whether the display of tactical graphics has been disabled. For details, see "Displaying Tactical Graphics".

3. Optionally, you can set the altitude of an object as you create it. For details, see "3.4.1 Setting Altitude Dynamically" on page 30.
4. Optionally, you can set the heading of an object as you create it. For details, see "3.5. Specifying an Object's Heading Dynamically" on page 32.
5. Optionally, you can set the pitch of some objects as you create them. For details, see "3.6. Specifying an Object's Pitch Dynamically" on page 33.

   > ⓘ If you want to specify any of the object's properties in the Create *object* dialog box while using Click to Create, you can do so. However, be aware that as you move the cursor into the dialog box, the object graphic will move also. Once you specify the desired properties, click on the terrain to create an object with those properties. The properties will be applied to any additional objects you create until you change them or close the dialog box.

6. For simulation objects and waypoints, right-click or press **Esc** to exit create mode.

## 3.3.2 Specifying an Object's Properties Before You Create It (Click to Locate)

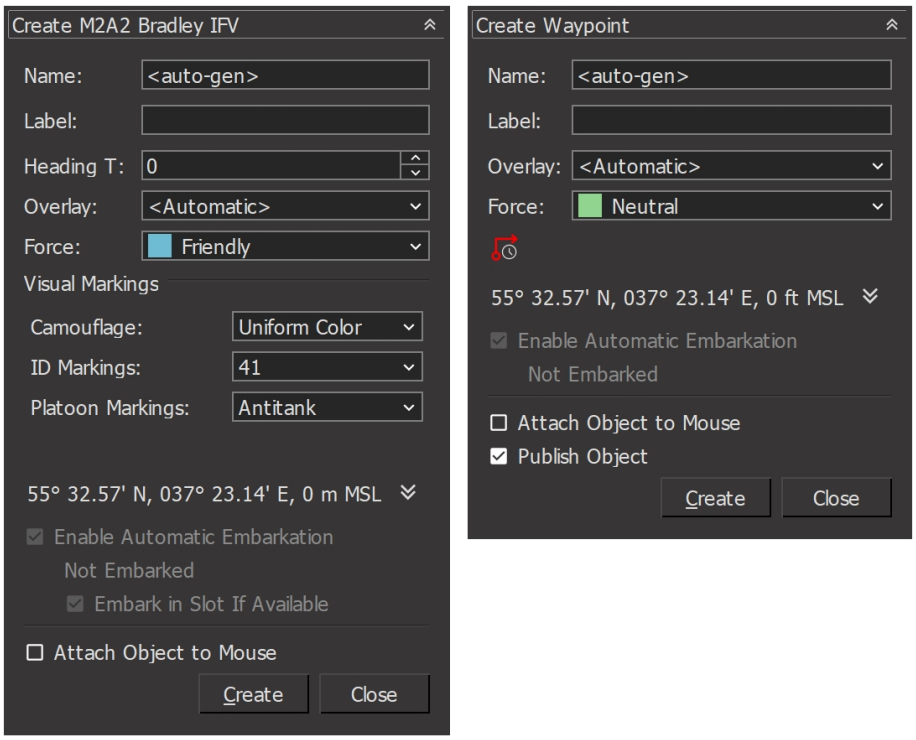
In the Click to Locate method, clicking on the terrain sets the coordinates of the object, but does not create it. You can specify all of the object's properties in the Create *object* dialog box before creating it. Table 3 on page 16 lists the properties that you can set for the different types of objects.

> VR-Forces remembers how you use the Create *object* dialog box on a per-object basis. For example, if you create a waypoint and open the Create Waypoint dialog box to specify its properties, the next time you create a waypoint, the Create Waypoint dialog box opens automatically. If the dialog box opens and you do not want it open for this object creation or for future creations of the current object type, click the Create *object* tab. The dialog box will close and VR-Forces will still be in object creation mode. The next time you create this object type, the dialog box will not open automatically.

➤ **To specify the properties of an object when you create it:**

1. Select the object that you want to create as described in "3.2. Selecting the Object to Create" on page 17. A Create *object* tab is added to the window below the palette (Figure 8 on page 25).
2. Click the Create *object* tab. A Create *object* dialog box that is appropriate for the object opens (Figure 11 below).

**Figure 11. Create *object* dialog boxes**



3. Clear the Attach Object to Mouse check box. For details about this option, see "3.7. Locking the Mouse to the Object Being Created" on page 33.
4. To specify the object's name, type a name in the Name box. For details about how simulation objects are named, see "2.2. How Simulation Objects Are Identified" on page 4.
5. Optionally, specify a label.
6. Optionally, if extended labels are available, specify them. For information about extended labels, see "5.3. Using Extended Labels" on page 73.
7. To specify the object's heading, enter a value, in degrees, in the Heading box. (Simulation objects only.)
8. For aircraft, you can enter the Current Speed and specify whether that speed is the ground speed, true airspeed, indicated airspeed, or Mach number.
9. To specify the overlay on which to place an object, select an overlay from the Overlay list.
10. Specify a force.
11. Optionally, if visual markings are available, specify them.
12. To specify a simulation object or point object's coordinates, click on the terrain or type coordinate values in the dialog box. To specify the altitude, follow the procedure in "3.4. Specifying an Object's Altitude" on the next page.

    To specify the vertices of a route or area:

    a. Click on the terrain to place the first vertex, or type the coordinates in the dialog box.
    b. Click the Add button (+) to add a vertex.
    c. Click to locate the vertex or type the coordinates.
    d. Continue adding vertices until you are finished specifying the route or area. You can insert vertices anyplace in the object by selecting an existing vertex and clicking the Add button.

    To draw circles, ellipses, boxes, or text, see "Creating and Editing Tactical Graphics".
13. For tactical graphics, specify whether to publish the object. For information about publishing tactical graphics, see "Publishing Tactical Graphics".
14. Click Create. The object is created. You can continue to create additional objects of this type with the same properties.
15. To close the dialog box, click Close.

### 3.3.3 Default Placement of New Entities

By default, ground, lifeform, rotary-wing, and fixed-wing entities are placed on the ground. Surface and subsurface entities are created at sea level.

Ground-based entities are placed at the highest possible terrain intersection at the location. If a terrain supports buildings with multiple levels, the entity is placed at the highest level. Once the entity begins moving, if the next calculated location resides in-

between two levels of terrain, and if the entity is configured to respect terrain intersections when moving, and there is no terrain blocking its movement, the entity places itself at the closest elevation to the desired location.

### 3.3.4 Placing Entities Inside Buildings

One of the benefits of creating entities in a 3D observer mode is the ease of placing them in the precise location and orientation that you want them to be in, particularly inside buildings. However, placing entities inside small enclosed spaces can be hindered by the location of walls relative to the observer. You can work around this problem by adjusting the near clipping plane to remove the display of walls and floors until you see the location where you want to place entities. For details about changing the clipping plane, see "Setting the Clipping Planes".

> ⓘ If you enable view constraints, it restricts your ability to move the observer into buildings. See "Enabling and Disabling View Constraints (3D Only)".

### 3.3.5 Placing Simulation Objects on Other Simulation Objects

If you want to place a simulation object on or in another simulation object, such as an aircraft on an aircraft carrier or a dismounted infantry entity in a truck, then you must embark the simulation object. Creating a simulation object on top of another simulation object will not provide the results you want. For example, if you create a fixed-wing entity and manipulate its location and altitude so that it looks like it is on the deck of an aircraft carrier, as soon as you start the simulation, the jet will start flying at that altitude because it is not attached to the carrier. It is simply created at a particular location. For complete details about embarking simulation objects, see "6.1. Embarking and Disembarking Simulation Objects" on page 86.

## 3.4. Specifying an Object's Altitude

You can specify an altitude for air platforms, sub-surface entities, the vertices of waypoints and routes, and props when you create them or edit their properties. (You can also set altitude for simulation objects using the Location set data request, Fly Altitude tasks, and the Move to Location task. You can set altitude dynamically or in the various dialog boxes for setting object properties or actions.)

### 3.4.1 Setting Altitude Dynamically

You can set altitude dynamically for simulation objects that support an altitude, for waypoints, for individual vertices of a route, and for props that have not been saved into a terrain. When you set altitude dynamically, it is set relative to the terrain.

➤ **To set altitude dynamically:**

- If you are creating an object, press and hold **Alt** and scroll the mouse wheel, or press and hold **Alt**+left mouse button and drag the mouse.
- If you are editing a simulation object or vertex:

1. Double-click the object to make it editable.
2. Press and hold **Alt** and scroll the mouse wheel, or press and hold **Alt**+left mouse button and drag the mouse.
3. Click to set the new altitude.

> Optionally, you can press **Ctrl**+**Alt** and then click and drag to move the object by 5 meters at a time.

## 3.4.2 Setting Altitude in the Create Object or Edit Object Dialog Box

When you set altitude in a dialog box, you have the following options:

- Above Sea Level. The distance above sea level. If the terrain is higher than the altitude entered, the entity will be below the surface of the ground.
- Above Terrain. The distance above the terrain at this location.
- If you are creating or editing a route, you can adjust the altitude of all of the vertices at the same time and in the same way. For details, see "3.4.3 Specifying the Altitude for All of the Vertices in a Route" below.

➤ **To set altitude in a dialog box:**

1. If you are creating or editing an object:
   a. Click the Create *object* (or Edit *object*) tab. A dialog box opens.
   b. Clear the Attach Object to Mouse check box.
2. Expand the location group box by clicking the arrow next to the coordinates and altitude.
3. Under Altitude, select one of the options.
4. Enter an altitude in the box.

## 3.4.3 Specifying the Altitude for All of the Vertices in a Route

When you create a route or edit its vertices, you may want to change the altitude of all of its vertices in a similar way. VR-Forces lets you:

- Adjust all vertices by the same amount.
- Set all vertices to the same height above sea level.
- Set all vertices to the same height above the terrain.

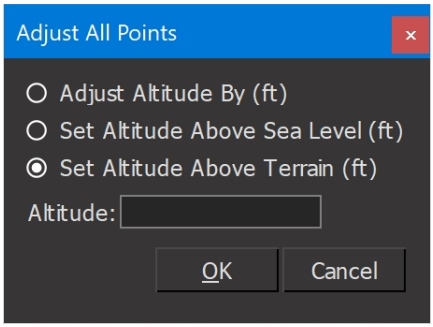
➤ **To change the altitude for all of the vertices in a route:**

1. If you are creating a route, open the Create Route dialog box.

   If you are editing a route:
   a. Select the route.
   b. Choose **Objects > Edit**. The Edit Route dialog box opens.

2. Click the Adjust Points button (). The Adjust All Points dialog box opens (Figure 12).

**Figure 12. Adjust All Points dialog box**




3. Select the desired adjustment action:
   - Adjust Altitude By. Changes the altitude of each vertex by the amount specified in the Altitude box.
   - Set Altitude Above Sea Level. Sets the altitude of each vertex to the amount above sea level specified in the Altitude box. This could result in some vertices being below ground.
   - Set Altitude Above Terrain. Sets the altitude of each vertex the amount above the terrain specified in the Altitude box.
4. Enter the desired altitude adjustment value in the Altitude box.
5. Click OK.

## 3.5. Specifying an Object's Heading Dynamically

When you create a simulation object, cultural feature, or prop, you can set its heading in the Create *object* dialog box, or dynamically. Afterward, you can edit a simulation object or cultural feature's heading directly or use the Heading set data request. For more information, see "Heading".

> ⓘ In the 2D view, as you change the heading for a simulation object you can see the heading indicator change heading.

➤ To set an object's heading dynamically as part of the creation process, hold down the **Shift** key and the left mouse button, then drag the mouse.

➤ To edit the heading of an object that has already been created, double-click it to put it into edit mode. Then hold down the **Shift** key and the left mouse button, then drag the mouse.

> ⓘ Optionally, you can press **Ctrl**+**Shift** and then click and drag to adjust the heading by 15 degrees at a time.

## 3.6.  Specifying an Object's Pitch Dynamically

In entity-level scenarios, some simulation objects, cultural features, and props allow you to set the pitch in the Create *object* dialog box or dynamically. Afterward, you can edit the object's pitch directly or use the Pitch set data request. For details, see "Pitch".

> There is no indication of the pitch in the 2D view, therefore you may want to switch to see the model's pitch and adjust it as needed.

➤ To set an object's pitch dynamically as part of the creation process, press and hold **Shift**+**Alt** and the left mouse button, then drag the mouse.

➤ To edit the pitch of an object that has already been created, double-click it to put it into edit mode. Then press and hold **Shift**+**Alt** and the left mouse button, then drag the mouse.

> Optionally, you can press **Ctrl**+**Shift**+**Alt** and then click and drag to adjust the pitch by 15 degrees at a time.

## 3.7. Locking the Mouse to the Object Being Created

By default, when you create simulation objects or waypoints, you can create multiple objects by continuously left-clicking without exiting create mode. This is a very convenient feature if you want to create objects using default naming conventions and do not need to specify specific location coordinates or overlay assignments during the creation process. (You can always edit an object's properties after you create it.)

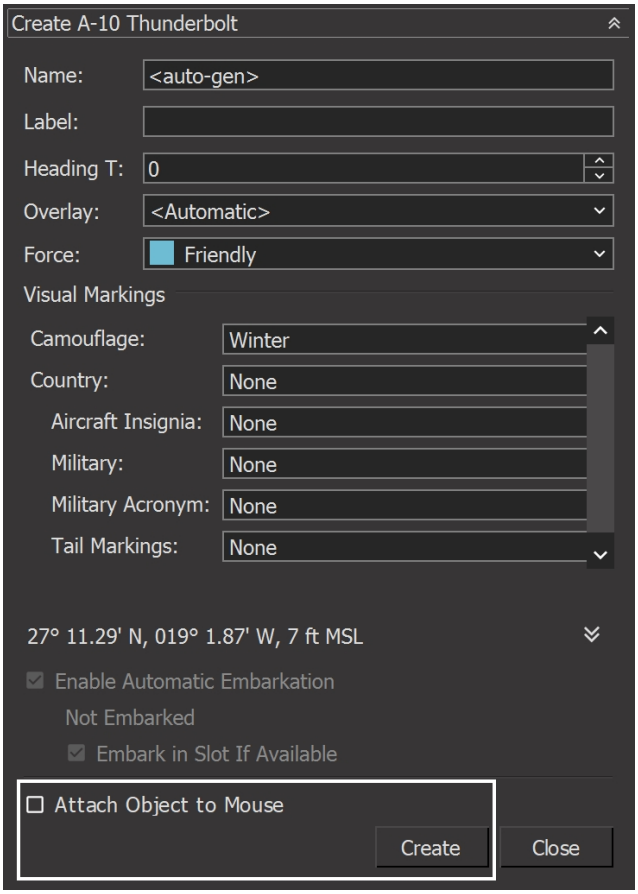
However, if you want to use the Create *object* dialog box to specify location or other properties for each object as you create it, this feature can be inconvenient because as you move the mouse to the Create *object* dialog box, the placement location changes with the mouse movement. Disabling this feature can give you greater control over the creation process, because clicking to set the location of the object or a vertex does not create it – it sets the coordinates, but lets you make further changes in the Create *object* dialog box before you create the object.

If you disable an object's attachment to the mouse when you are creating objects, you create them by clicking Create in the dialog box rather than clicking on the terrain.

> If you have cleared the Attach Object to Mouse check box in the Create *object* dialog box, you must still specify the minimum number of points for the object before the Create button is enabled. For simulation objects this is only the location of the object, but for tactical graphics such as lines, circles, and areas, you may need to specify two, three, or more points. Since the object is not attached to the mouse, you will not see dynamic resizing of the object as you specify the points.

➤ To enable or disable the mouse's attachment to the creation location, in the Create *object* or Edit *object* dialog box, select or clear the Attach Object to Mouse check box (Figure 13). When you clear the check box, a Create button is added to the dialog box. The setting is carried over to future creations of that object.

**Figure 13. Create *object* dialog box**




## 3.8. Simulation Object Resources

When you create a simulation object, it has a full complement of resources, as defined in the object parameter database. You do not have to assign resources to a simulation object when you create it. However, if you want a simulation object to start a simulation with less than a full set of resources, you can use the Resources set data request to set its resources at some amount less than 100%. For details about the Resources set data request, see "Resources".

## 3.9. Moving Objects

You can move an object by dragging it, changing its location in the Edit dialog box or, for simulation objects only, sending a Location set data request.

For information about setting the location of a simulation object, see "Location".

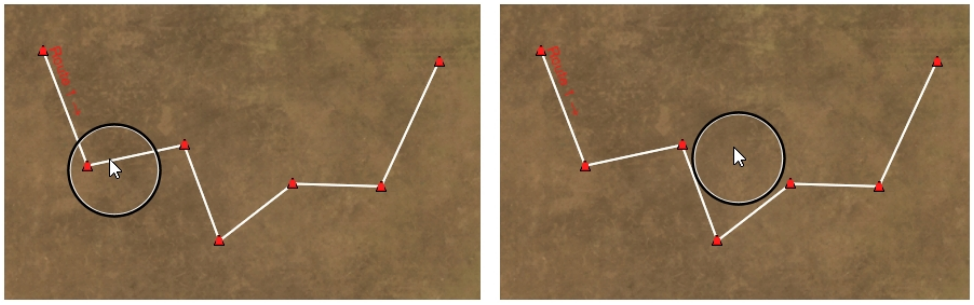
> ⓘ You can only move objects simulated by VR-Forces.

### 3.9.1 Dragging an Object to a New Location

When you move objects, they are centered on the mouse location:

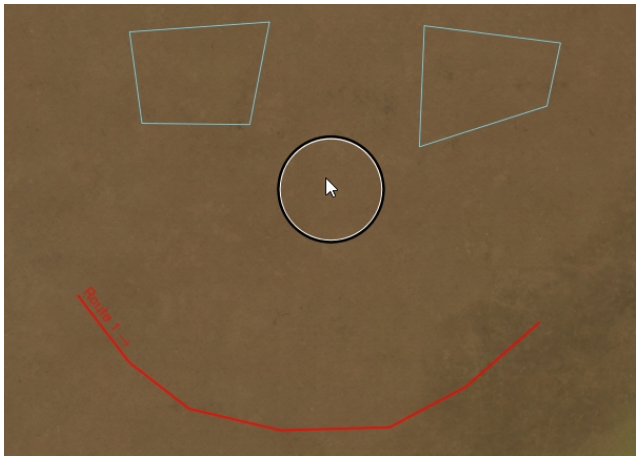
- If you select an object by double-clicking, it attaches to the mouse at the location where you clicked it. (For simulation objects and point objects, the objects are centered on the mouse location, but for lines or areal objects, the attachment is at the point where you clicked (left image in Figure 14).)
- If you select an individual object and move it by choosing the Move command:
  - Point objects are centered on the mouse, just as when you double-click them.
  - Linear and areal objects attach to the cursor at the center of mass (right image in Figure 14).

**Figure 14. Mouse attachment when moving individual lines**



- If you select multiple objects and move them by choosing the Move command:
  - Point objects attach to the cursor at their center of mass.
  - Multiple objects have their centers of mass averaged and the center point attaches to the cursor (Figure 15).

**Figure 15. Moving multiple objects**



If you click the terrain and then **Shift**+click or **Ctrl**+click an object, the terrain is selected and the Move command will not be available.

➤ **To drag an object from one point on the terrain to another:**

1. Double-click the object, or select the object and choose **Objects > Move**.
2. Move the mouse to change the location of the object.
3. Left-click the mouse.

ⓘ
- You can drag multiple objects and they can be of different types.
- To drag multiple objects, you must select the objects and then choose **Objects > Move**.
- There may be indicators showing that the object is moving from its original location to the new one. This tells you that the move was successful while VR-Forces repositions the object.

### Canceling a Drag-and-Drop Move

➤ To cancel a move made by dragging an object, press **Esc** before you click the left mouse button.

## 3.10. Editing Simulation Objects

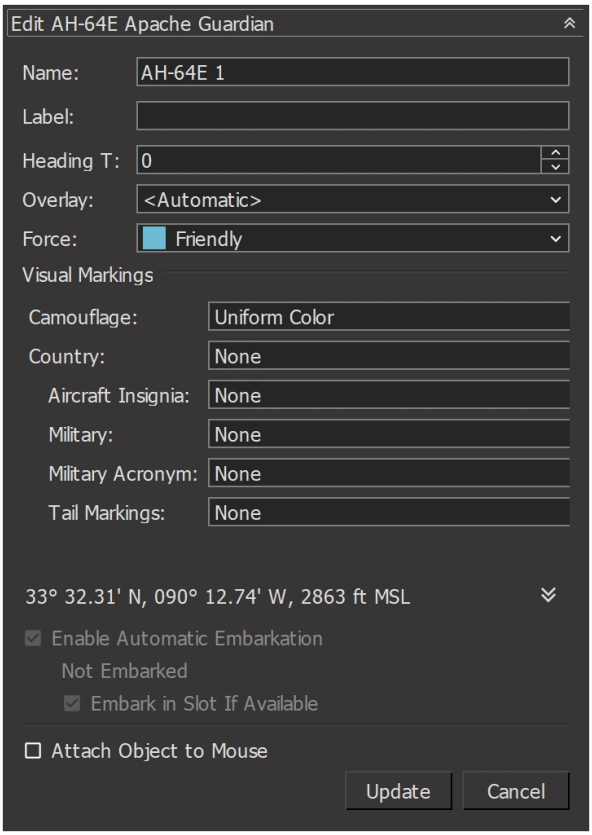
You can edit these simulation object properties:

- Name
- Label (and extended labels if available)
- Force
- Visual Markings (if available)
- Location
- Heading
- Altitude
- Overlay

➤ **To edit a simulation object:**

1. Select the simulation object.
2. Choose **Objects > Edit**. The Edit Object dialog box opens (Figure 16).

**Figure 16. Edit Object dialog box**




3. Change the properties as desired.
4. Click OK.

You can also change the location, altitude, and heading dynamically, through a variety of movement tasks or with set data requests. For details, see:

- "3.4.1 Setting Altitude Dynamically" on page 30.
- "3.5. Specifying an Object's Heading Dynamically" on page 32.
- "3.9. Moving Objects" on page 34.
- "Movement Tasks for Entity-Level Scenarios".
- "Tasks for Aggregate-Level Scenarios".
- "Setting the State of Entity-Level Simulation Objects".

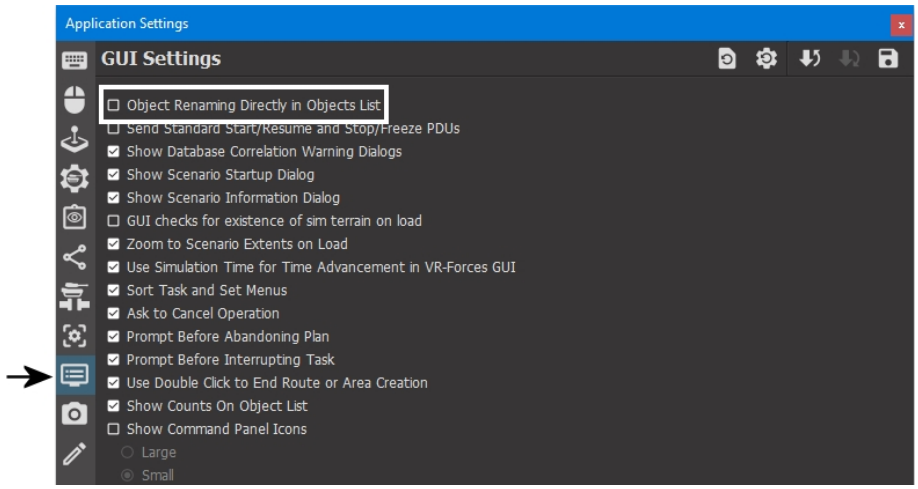
## 3.10.1 Editing an Object's Name in the Object List

You can edit an object's name in an Edit dialog box or directly in an object list. The ability to edit an object's name in an object list is disabled by default. This prevents you from inadvertently going into edit mode when you want to select an object.

➤ **To enable and disable editing an object's name in an object list:**

1. Choose Settings Application Settings. The Application Settings dialog box opens.
2. Select the GUI Settings page (Figure 17).

**Figure 17. GUI Settings page**




3. To allow editing of names in an object list, select the Object Renaming Directly in Objects List check box. To prevent editing of names, clear the check box.
4. Click Close

➤ **To edit an object's name in an object list:**

1. Click the name of the object, then click again. (This is two single clicks, not a double-click. The two clicks should be about one second apart.) The name becomes editable.
2. Edit the name.
3. Click someplace else in the window.

## 3.11. Copying and Pasting Objects

You can copy objects and paste the copies elsewhere in a scenario or even in another scenario. The Paste Special option allows you to specify which entity characteristics you paste.

VR-Forces copies the following data about an object:

- Environmental objects:
  - Original name
  - Label
  - Force
  - Color

- Entities:
  - Original name
  - Label
  - Force
  - Heading
  - Appearance (lights, landing gear, and so on)
  - Rules of engagement
  - Plan
- Units copy the same information as entities plus the list of subordinates.

VR-Forces does not copy:

- The current task.
- State information, except as noted previously.
- Load balancing status. All objects are pasted into the currently selected sim. engine.
- Embarkation structures. If, for example, you copy an aircraft carrier that has planes embarked, only the carrier is copied, not the embarked entities.

> ⚠ The copy/paste process is resource intensive. It is primarily for use during scenario development. Use during a simulation could affect performance.

## 3.11.1 Copying Objects

➤ **To copy objects:**

1. Select the objects you want to copy.
2. Choose **Objects > Copy**.

## 3.11.2 Pasting Objects

A simple paste pastes all of the attributes of an object that were copied. It pastes objects as follows:

- Tactical graphics are placed on the currently selected overlay. If you copied objects that were on different overlays, that differentiation is not preserved.
- Copied objects maintain the state at the time that they were copied. Therefore, if you paste objects some time after they were initially copied, the copies do not reflect any changes that may have taken place to the original simulation objects. For example, if a simulation object was copied and then deleted, the copied version of the entity is still pasted.
- Embarkation structures are not copied, only the parent entity.

> ⚠ Pasting plans may have unexpected consequences because their object references do not change when you copy them. For example, suppose the plan for Entity A refers to Waypoint 1. You copy Entity A and Waypoint 1 and paste them as Entity B and Waypoint 2. If you copy the plan, Entity B's plan will still refer to Waypoint 1. If you expected it to refer to Waypoint 2, you will not see the behavior you expected. You can choose not to paste plans. For details, see "3.11.3 Pasting Specific Entity Characteristics" below.

➤ **To paste objects:**

1. Choose **Objects > Paste**. The objects in the paste buffer display on screen. They float as you move the cursor to show you where they will be placed when you paste them.
2. Click on the terrain where you want the objects to be pasted. They are pasted in the relative positions in which they were copied. If an object has an altitude, it is pasted using the same altitude above the terrain as the original object.
3. Optionally, continue pasting additional copies if desired.
4. Right-click to exit paste mode.

## 3.11.3 Pasting Specific Entity Characteristics

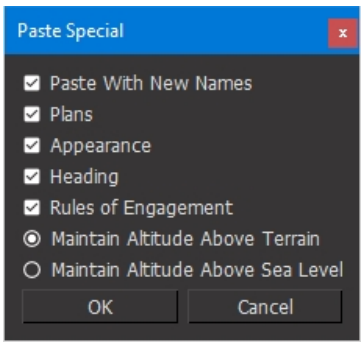
When you paste simulation objects, you can specify which of these entity characteristics you want to paste with the entity:

- Plans
- Appearance
- Heading
- Rules of engagement
- Entity name
- Altitude. You can specify that the previous altitude be applied as above terrain or as above sea level.

➤ **To paste particular entity characteristics:**

1. Choose **Objects > Paste Special**. The Paste Special dialog box opens.

**Figure 18. Paste Special dialog box**

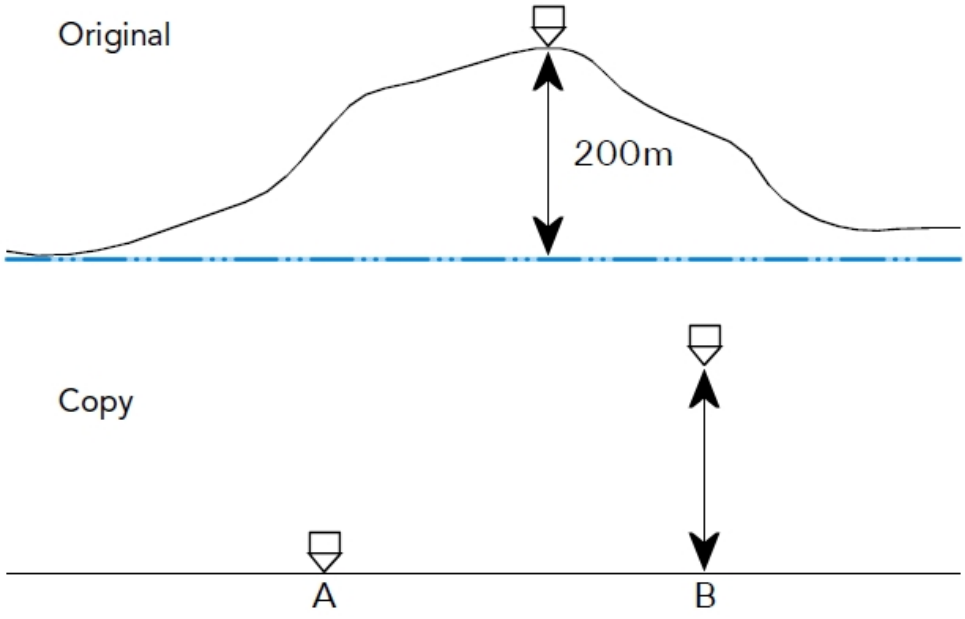


2. By default, all check boxes are selected. Clear the check boxes for the options you do not want to paste.
3. Choose the altitude option that you want to use.
4. Click OK. The objects in the paste buffer display on screen. They float as you move the cursor to show you where they will be placed when you paste them.
5. Click on the terrain where you want the objects to be pasted. They are pasted in the relative positions in which they were copied.
6. Optionally, continue pasting additional copies if desired.
7. Right-click to exit paste mode.

### Pasting Altitude

When you paste objects, if the pasted object can have an altitude value, such as a fixed-wing entity, a waypoint, or the vertices of a route, the default behavior is to replicate the original altitude above the terrain. If you use the Paste Special operation, you can choose to apply the original altitude as being above sea level. Figure 19 illustrates these alternatives. A waypoint is placed on the terrain. It is 0 meters above the terrain. However, the terrain itself is 200m above sea level. If the waypoint is copied and pasted at point A and the altitude above the terrain is maintained, the point is placed at 0m above the terrain. If the point is pasted at point B and altitude above sea level is maintained, it is placed 200m above sea level.

**Figure 19. Pasting altitude**




## 3.12. Deleting Simulation Objects

➤ **To delete a simulation object:**

1. Select the object you want to delete.
2. Choose **Objects > Delete** or press the **Delete** key.

- If you delete a unit, all of its subordinates are also deleted.
- If you delete a simulation object that has an intervisibility object pinned to it, the intervisibility object is also deleted. For details, see "Deleting Intervisibility Objects".

## 3.13. Simulation Object Groups

Simulation object groups (SOG) are a convenient mechanism for creating reusable configurations of simulation objects and tactical graphics. You create them in the Simulation Object Editor and they are listed on the Create Object Palette with other simulation objects. VR-Forces places them in the relative positions that you specified when you created the SOG. VR-Forces has example SOGs, including a carrier group, a Patriot missile battery, a barracks with several soldiers, and others. These examples demonstrate SOGs with tactical graphics, cultural features, and embarkation.

The Barracks With Guards SOG has some tactical graphics, but they are hidden by default. To see them, click the overlay icon at the bottom of the window, or select the check box on the Objects List Panel, Overlays View.

The Create Object Palette has a tag for SOGs (Object Group), but like other simulation objects, they can have multiple tags. SOGs can be distinguished from other simulation objects in the lists because the object name includes the number of objects in the group.

For information about creating SOGs, see "Creating a Simulation Object Group".

## 3.14. Creating Random Versions of Similar Entity Types

VR-Forces has many different models for generic types of entities, such as civilian vehicles or commercial aircraft. When you are creating a scenario with these sorts of entities, you can choose the specific entity types you want to use. However, if you want to quickly populate a scenario with a variety of similar objects, it is simpler to have VR-Forces randomly create instances of the generic entity. This is done using the random simulation object.

A random simulation object is configured in the Simulation Object Editor with a list of specific simulation object types that it can draw from. When you create a scenario, you choose the random entity type on the Create Object Palette and when you click on the terrain to place the object, VR-Forces picks a specific simulation object from the list.

VR-Forces includes a random simulation object for civilian vehicles.

You can add your own random simulation objects in the Simulation Object Editor. See "Creating Randomized Simulation Objects".

## 3.15. Creating Cultural Features

Cultural features are objects that you can add to a scenario to simulate human artifacts such as buildings, bridges, and so on. They have the Cultural Feature tag on the Create

Object Palette. You add them to a scenario the same way you would add an entity or tactical graphic. Cultural features are listed in the Objects List View of the Objects List Panel. You can apply some set data requests to them, such as Location and Heading.

> ⚠ Some of the choices on the cultural feature lists are the same as those available on the Props Palette. However, cultural features are not the same things as props. Props are terrain objects and can be saved as part of a terrain. Cultural features are not part of the terrain; they are a type of simulation object and are saved in the scenario objects file.

## 3.15.1 Embedding Cultural Features in the Terrain

If you have a cultural feature, such as a house, that has object geometry, VR-Forces can treat that object as part of the terrain when you place it in a scenario. This means that entities can walk on it or in it (if it has interior geometry).

If you place the object outside of a navigation area, entities interact with it similarly to how they would interact with a prop. If you place the object in a navigation area, VR-Forces dynamically regenerates navigation data for the sectors in which the cultural feature is located. Entities can then use the navigation data to walk around or in the cultural feature more intelligently than if it was outside of a navigation area.

> ⓘ You do not need a license for generating navigation areas to dynamically regenerate navigation data. However, you must have a terrain on which navigation areas have been created, such as the terrains provided with VR-Forces. For information about navigation areas and navigation data, see "Generating Navigation Data".

This feature has the following requirements and constraints:

- The cultural feature must be configured in the Simulation Object Editor to use object geometry in the simulation engine and to be embedded in the terrain. For details, see "Configuring Cultural Features To Be Embedded in the Terrain".
- If you place the cultural feature in a navigation area, the area must be sectorized. If it is not sectorized, the navigation data will not be generated.
- Navigation data is generated dynamically. It is not saved with the terrain or with the scenario. When you reload the scenario, the navigation data is generated after the scenario loads. If you have many such cultural features in a scenario, this can take some time. During that time, you might not be able to use VR-Forces or other applications on your computer.
- Creating a navigation area and generating navigation data for a portion of a terrain in which you have placed a cultural feature does not generate navigation data for the cultural feature. The navigation data is generated based on the content of the terrain and the cultural feature is a simulation object, not part of the terrain. If you want the geometry of a cultural feature to be part of the terrain, add it as a prop.

➤ To embed a cultural feature in the terrain, create it in a scenario.

## 3.16. Creating Props

Props are terrain objects that you can select independently of the terrain. "Extracting Props from a Terrain", in the *Adding Content to MAK ONE Applications Reference Manual*, explains how to create them by extracting them from a terrain or a feature layer. You can also create them from the Props Palette, however, similarly to how you create simulation objects. This is a valuable feature if your terrains do not have any features that you can extract as props. Do note these important details:

- Props are not saved in a scenario. If you want to save a prop, you must save the terrain. You can save the terrain to a different name or overwrite the existing terrain. The next time you open the terrain, the prop will be part of it.
- When you create a prop, the sim. engine does not know anything about it because it is a terrain object. To have the sim. engine take the prop into account, you must save the terrain (as mentioned in the previous paragraph) and restart the scenario.
- Props are created on the GUI terrain. If the GUI and sim. engine are using different terrains, the sim. engine terrain will not include the props.
- You can export props to a shapefile, allowing you to add them to a terrain using an earth file. For details, see "Exporting Props and SpeedTrees to a Shapefile" in the *Adding Content to MAK ONE Applications Reference Manual*.
- You can also create cultural features that look exactly like the props that you can create. However, cultural features are not the same things as props. Cultural features are treated as simulation objects. They are listed with simulation objects and tactical graphics and are saved as part of the scenario.

> ⓘ The list of props that you can create from the Props Palette is based on the model definitions defined on the Model Definitions tab of the Visual Model Editors dialog box. Therefore, you can add new props to the Props Palette by adding new model definitions. For details, see "Adding Object Models as Props" in the *Adding Content to MAK ONE Applications Reference Manual*.

## 3.17. Adding an Object to the Favorites List

As mentioned in "3.2. Selecting the Object to Create" on page 17, the Create menu does not include all of the objects that you can create. It just lists the objects that you have designated as "Favorites". When you specify that a simulation object or tactical graphic be a favorite, it is automatically added to the Create menu. An option is added to the menu for the type associated with the favorite. As you add additional favorites, they are listed under all of the tags that apply to them.

In the Create Object Palette, objects that are favorites have a small star next to the icon. You can also filter the object list using the Favorites tag (★) to see which objects you have marked as favorites.

The favorites list is specific to a simulation model set. It is saved to *./data/simulationModelSets/<model_set>/gui/favorites.mst*. You can copy the file to other instances of VR-Forces or future releases and your favorites list will be enabled.

➤ **To add or remove an object from the Favorites list:**

1. Open the Create Object Palette.
2. Right-click the object.
3. On the pop-up menu, select **Add to Favorites**.

### 3.17.1 Removing an Object from the Favorites List

When you remove an object from the Favorites list, it is also removed from the Create menu.

➤ **To remove an object from the Favorites list:**

1. Open the palette that has the object you want to remove from the Favorites list.
2. Right-click the object.
3. On the pop-up menu, select **Remove from Favorites**.

# 4. Selecting Objects

This chapter describes ways to select simulation objects and tactical graphics.

# 4.1. The Objects List Panel

There is a great deal of similarity in how you select, move, and view the different types of objects. You can manage them on the terrain, to some extent from the keyboard, and in the Objects List Panel, which is the central repository of object information.

The views in the Objects List Panel are:

- Object View: displays simulation objects in the simulation in a hierarchical list by force. Also lists tactical graphics (Figure 20).
- Embarkation View: displays the embarkation relationships of simulation objects (Figure 20).
- Overlays View: lists overlays and the objects that are on them (Figure 21).
- Selection Groups View: lists selection groups and their members (Figure 21).

**Figure 20. Objects List Panel – Objects List View and Embarkation View**

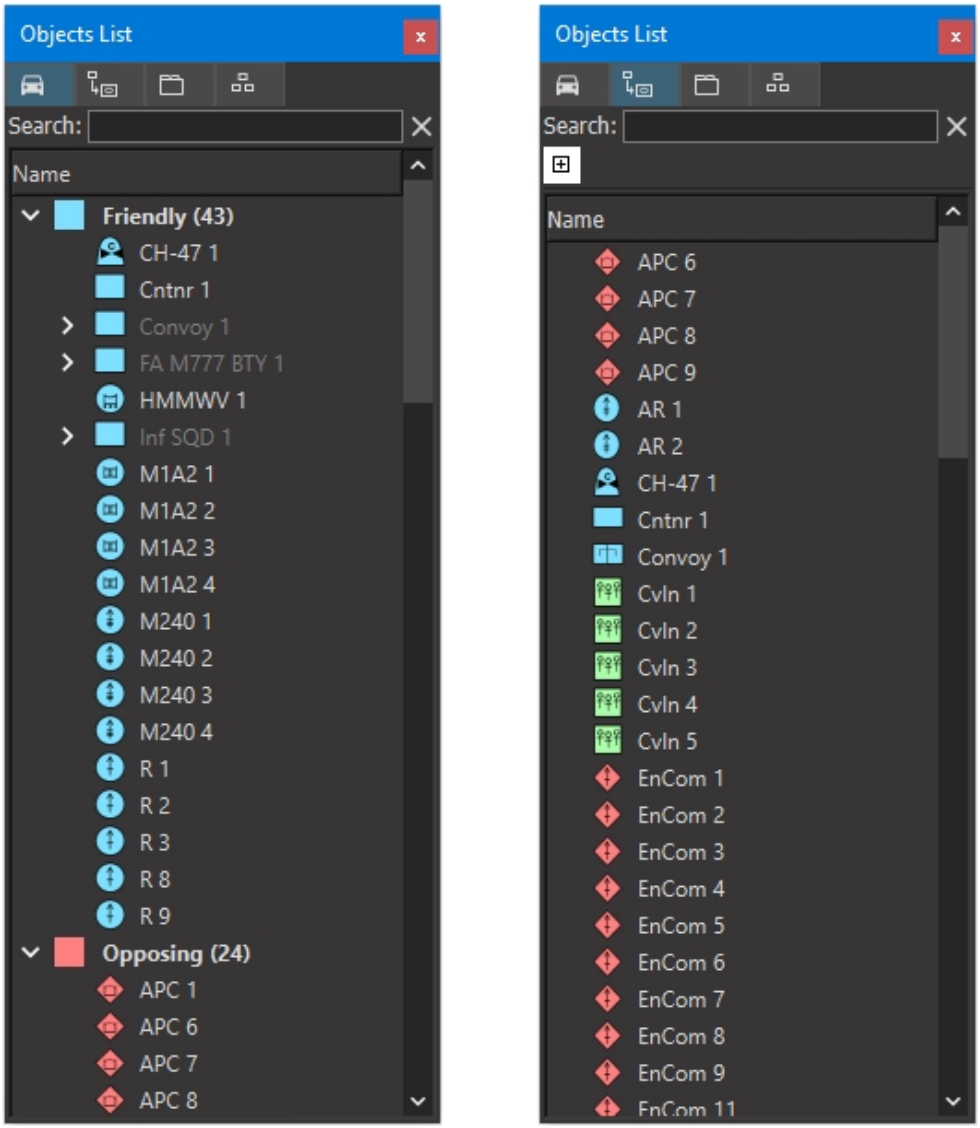
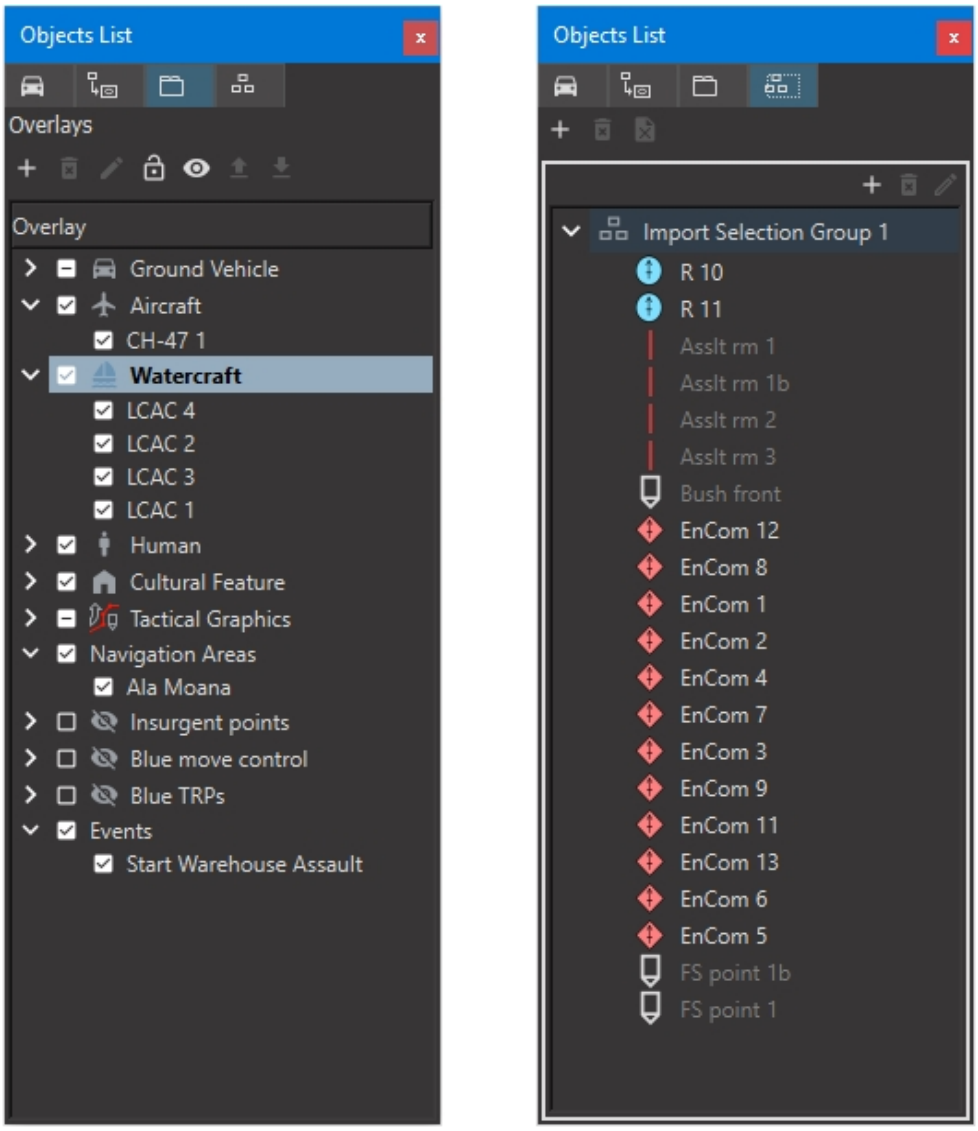


**Figure 21. Objects List Panel – Overlays View and Selection Groups View**




The objects in the objects list panel are identified by their 2D icon and their name. You can use images other than the default icons to identify objects. For details, see "Using an Alternative Icon List".

## 4.2. Selecting Simulation Objects, Tactical Graphics, and Props

VR-Forces displays the following kinds of objects that you might want to select:

- Simulation Objects. Any of the entity and unit types that you can create.
- Cultural features. Objects that have the Cultural Feature tag on the Create Object Palette.
- Tactical graphics. Control objects and the points, lines, areas, text, and symbols on the Create Object Palette and Create menu.

- Props. Any of the props that you can create from the Props Palette or which are already part of the terrain.

You can select all types of objects in the window. You can select simulation objects, cultural features, and tactical graphics in the Objects List Panel (Figure 22 below). You can select simulation objects in the Object Console Summary Panel, and you can select props on the Props page of the Terrain Settings dialog box (Figure 23).

**Figure 22. Selected 3D model**

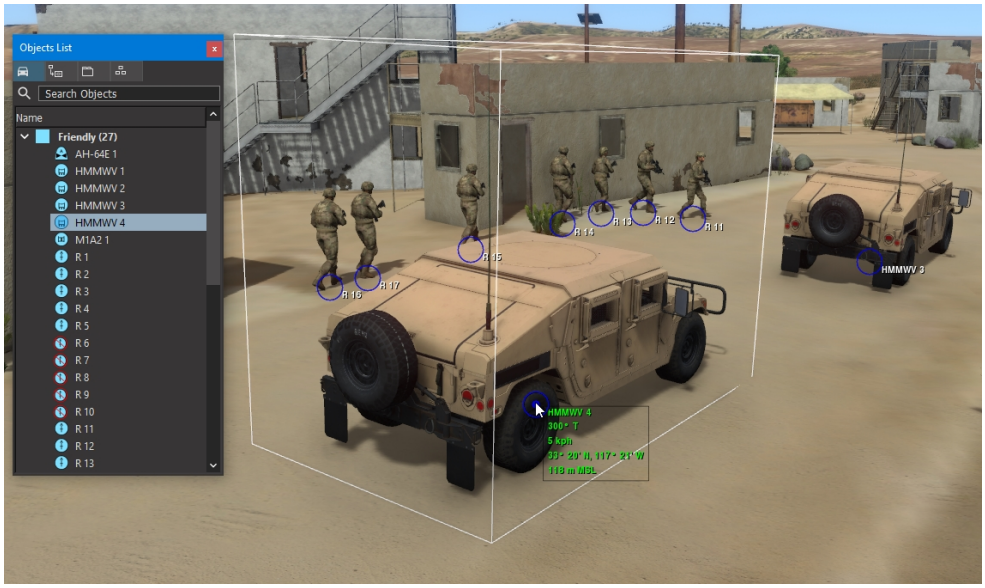
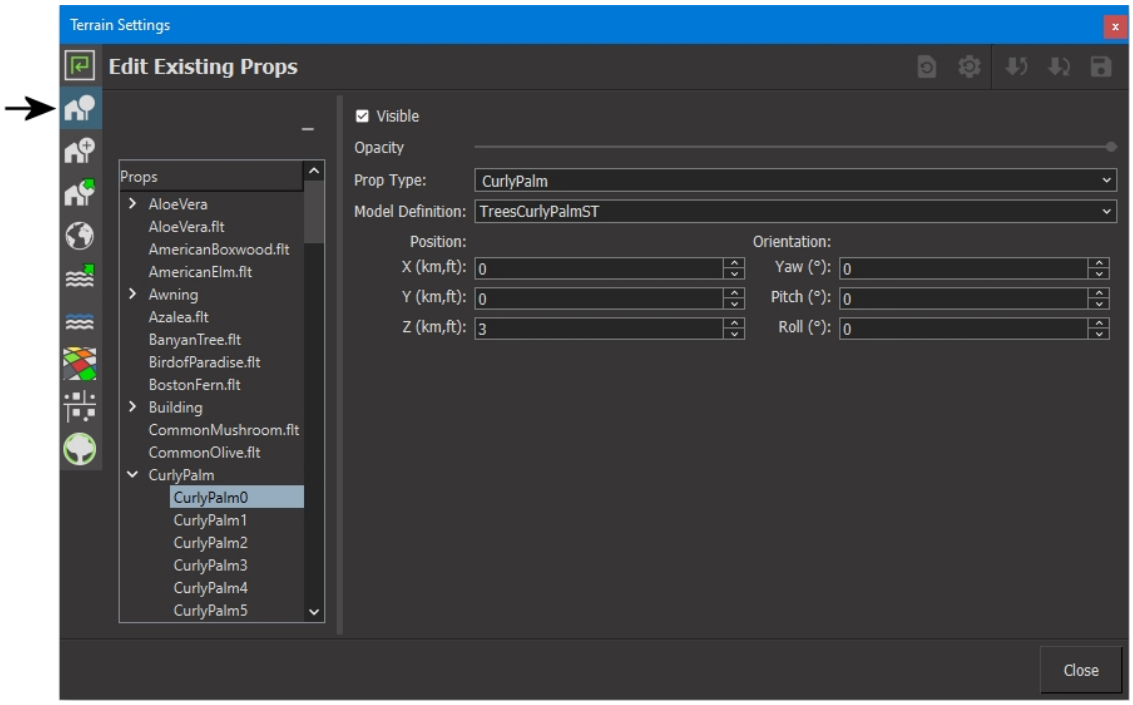
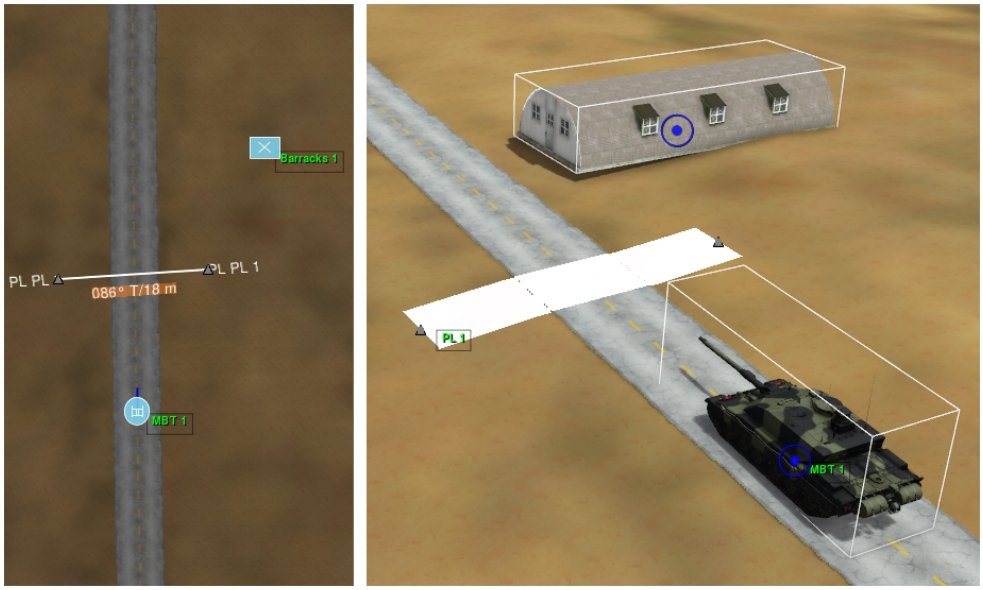



**Figure 23. Terrain Settings dialog box, Props page**



When you select a simulation object or tactical graphic in the 2D view, its border color changes. In the 3D views, all selected 3D objects display a bounding box (Figure 24). Tactical graphics outlines are highlighted.

**Figure 24. Selected objects (2D and 3D)**




## 4.2.1 Selecting Objects in the Window

To select objects in the window, use any of the following mouse actions:

| Mouse Action | Result |
|---|---|
| Click the object. | Selects an object (and unselects any other selected objects). |
| **Ctrl**+click the object. | Selects an object while keeping existing selections. |
| Double-click the object. | Selects the object and makes it editable. This is the equivalent of selecting an object and choosing **Objects > Edit**. |
| **Alt**+drag a bounding box. | Selects multiple objects by dragging a selection box around them. |
| Right-click the object. | If no objects are selected, right-clicking selects an object and displays a menu.<br>If one or more objects are selected, right-clicking a selected object maintains the selection and displays a menu. Right-clicking an unselected object changes the selection to the object you right-clicked. |

When you select an object in the window, the views on the Objects List Panel add a highlight to the entry for the object to indicate that it is selected. Note that when you select multiple objects with a selection box, the Objects List does not highlight the selected items until you release the mouse button.

> When visual item are displayed for simulation objects (radar coverage, range rings, sensor contacts, task visualizations, track histories, or task visualizations) and you then select multiple objects, the visual item displays only for the first selected object. To display them for more than one object at a time, use the appropriate pin option in the Objects > Decorations submenu.

## 4.2.2 Selecting Objects in the Objects List Panel

➤ **To select an object in the Objects List Panel:**

1. Select a view in the Objects List Panel (Figure 20 on page 48).
2. Locate the object you want to select. You can filter the list to make it easier to find particular objects. (For details, see "Filtering the Object Lists" below.) Expand units if necessary. (For details, see "Expanding and Collapsing Units".)
3. Click the entry for the object. The selected object is highlighted in the Objects List Panel and on the terrain.
4. Optionally, select multiple objects:
   - **Shift**+click to select a contiguous range of simulation objects.
   - **Ctrl**+click to select multiple non-contiguous simulation objects.

### Filtering the Object Lists

You can filter the display in the Objects List.

➤ To filter the display of objects, type a string in the Search box at the top of the list (Figure 20 on page 48).

## 4.2.3 Selecting a Simulation Object in the Object Console Summary Panel

The Object Console Summary Panel lists messages that have been sent to simulation objects from the simulation engine, from a simulation object's plan, from other simulation objects, and from scripts. (For information about the Object Console Summary Panel, see "5.9. Configuring Object Console Messages" on page 79.)

➤ **To select a simulation object in the Object Console Summary Panel:**

1. Right-click a message from the simulation object you want to select.
2. On the shortcut menu, choose Select Entity.

## 4.2.4 Unselecting Objects

To unselect objects in the window, choose one of the options in Table 4.

**Table 4. Unselecting objects on the terrain**

| Action | Result |
|---|---|
| Click in the window away from the selected object or objects. | Unselects all objects. |
| Right-click. | Exits edit mode, but object remains selected. |
| Click a selected object. | If multiple objects were selected, unselects all except the object you clicked. |
| **Ctrl**+click a selected object. | Unselects the object. If multiple objects were selected, does not unselect other objects. |

To unselect objects on the Objects List Panel, choose one of the options in Table 5.

**Table 5. Unselecting objects in the Objects List Panel**

| Action | Result |
|---|---|
| **If one object is selected:** | |
| **Ctrl**+click the object. | Unselects that object. |
| **If a group of objects is selected:** | |
| **Ctrl**+click an object. | Unselects the object without affecting other selections. |
| Click an object. | Selects the clicked object and unselects all other objects. |

## 4.3. Using Selection Groups

Selection groups are named subsets of the objects in an exercise. Their purpose is to allow you to quickly select specific objects without needing to locate them on the map or the Objects List View in the Objects List Panel and select them individually. Using selection groups can be helpful if you want to quickly select a group of objects that do not fit neatly into the default filter options on the Objects List Panel views, for example, a mix of friendly and opposing forces and tactical graphics. The group definitions are saved and loaded as part of the scenario.

An object can be in multiple selection groups.

### 4.3.1 Creating a Selection Group

You can create selection groups in any of the following ways:

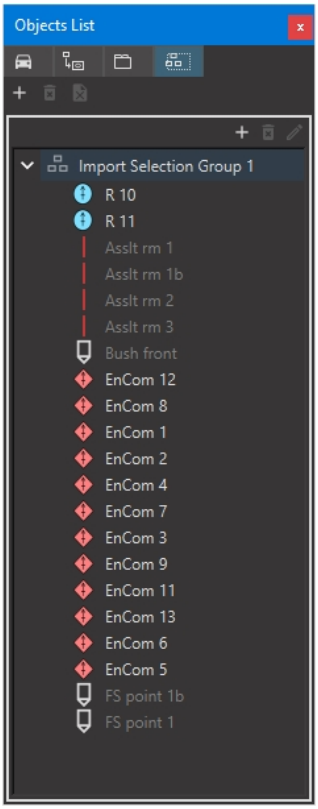
- In the Selection Groups View on the Objects List Panel.
- From Selection Groups menu options.
- Using keyboard shortcuts.

When you add a simulation object group to a scenario, VR-Forces automatically creates a selection group for all the objects in the simulation object group. For information about simulation object groups, see "3.13. Simulation Object Groups" on page 42.

➤ **To create a selection group in the Selection Groups Panel:**

1. On the Objects List Panel, select the Selections Groups View (Figure 25).

**Figure 25. Selection Groups tab**




2. Click the Add button (+). The New Selection Group dialog box opens.
3. Type a name for the selection group.
4. Click OK. The group is added to the group list.

### Creating a Selection Group from a Selection Group Menu Command

When you create a selection group from a Selection Group menu command, you can combine the procedures for creating a new selection group and adding an object to it.

➤ **To create a selection group from a Selection Group menu command:**

1. Select the objects you want to put in a new selection group.
2. Choose **Objects > Selection Groups > Add to New Selection Group**. The New Selection Group dialog box opens.

3. Type a name for the selection group.
4. Click OK. The selection group is created and the selected objects are added to it.

#### Creating a Selection Group from a Keyboard Shortcut

You can quickly create a selection group with a default name using a keyboard shortcut. When you create a selection group from the keyboard:

- If no objects are selected, an empty selection group is created.
- If objects are selected, a selection group is created with these objects as members.
- If the keyboard shortcut is for a selection group that already exists, a new selection group replaces the previous one. If the previous group had members and the new one is created without any objects selected, the new group is empty.

➤ To create a selection group from the keyboard, press **Ctrl** + *number*, where *number* is 1, 2, 3, 4, 5, 6, 7, 8, 9, or 0. A selection group is created named Selection Group *number* (0 creates Selection Group 10). (The keypad numbers do not support this operation.)

### 4.3.2 Selecting a Selection Group

When you select the objects in a selection group, they are selected in the views on the Objects List Panel and in the display window.

➤ **To select a selection group, use one of the following methods:**

- In the Selection Groups view, select a selection group in the Group list.
- Choose **Objects > Selection Groups > Select Selection Group >** *group_name*, where *group_name* is one of the groups listed on the pull right menu.
- If selection groups named with the format Selection Group *number* exist, pressing a number key selects the corresponding selection group. (The keypad numbers do not support this operation.)

### 4.3.3 Adding Objects to a Selection Group

➤ **To add objects to a selection group:**

1. Select the objects you want to add to a selection group.
2. Choose **Objects > Selection Groups > Add To Selection Group >** *group_name*, where *group_name* is one of the groups listed on the pull right menu.

### 4.3.4 Removing Objects from Selection Groups

There are two ways to remove an object from a selection group. If you remove all the objects from a selection group, the group continues to exist.

➤ **To remove an object from a selection group:**

1. Select the objects you want to remove from a selection group.
2. Choose **Objects > Selection Groups > Remove from Selection Group >** *group_name*, where *group_name* is one of the groups listed on the pull right menu.

➤ **To remove an object from a selection group in the Selection Groups view:**

1. Select the selection group from which you want to remove an object.
2. Select the object you want to remove.
3. Click the Remove Selected from Selection Group button (  ).

## 4.3.5 Renaming a Selection Group

You can rename selection groups. If you give a group the name Selection Group *number*, where *number* is 1, 2, 3, 4, 5, 6, 7, 8, 9, or 0, then when you press that number then the corresponding selection group becomes selected. (The keypad numbers do not support this operation.)

➤ **To rename a selection group:**

1. In the Selections Group view, double-click the name of the group you want to rename. It becomes editable.
2. Type a new name for the selection group.
3. Press **Enter**.

## 4.3.6 Deleting a Selection Group

Deleting a selection group does not delete the objects from the scenario. However, if you delete all the members of a selection group, the selection group is also deleted.

➤ **To delete a selection group:**

1. In the Selections Group view, select the group you want to remove.
2. Click the Delete button (  ).

# 5. Viewing Information About Objects

This chapter describes how to get and display information about simulation objects.

# 5.1. Displaying Information About an Object

You can display information about individual objects, including entities, units, environmental objects, and Stealths. You cannot edit this information. You can also display entity labels. For details about entity labels, see "5.2. Displaying Entity Labels" on page 65.

➤ **To display an object information dialog box:**

1. Select an object.
2. Choose **Objects > Information**, or press **I**.

ⓘ Other methods for opening the Information dialog box are:

- Select a message in the Objects Console Summary Panel and choose **Information** on the shortcut menu.
- Click any field in the summary panel to open the relevant dialog box page.
- Click the Show Information button (ⓘ) on the Last Selected Object Panel or Command Panel to open the State Data page.
- Click the current task on the Last Selected Object Panel to open the Information dialog box to the Tasks page.

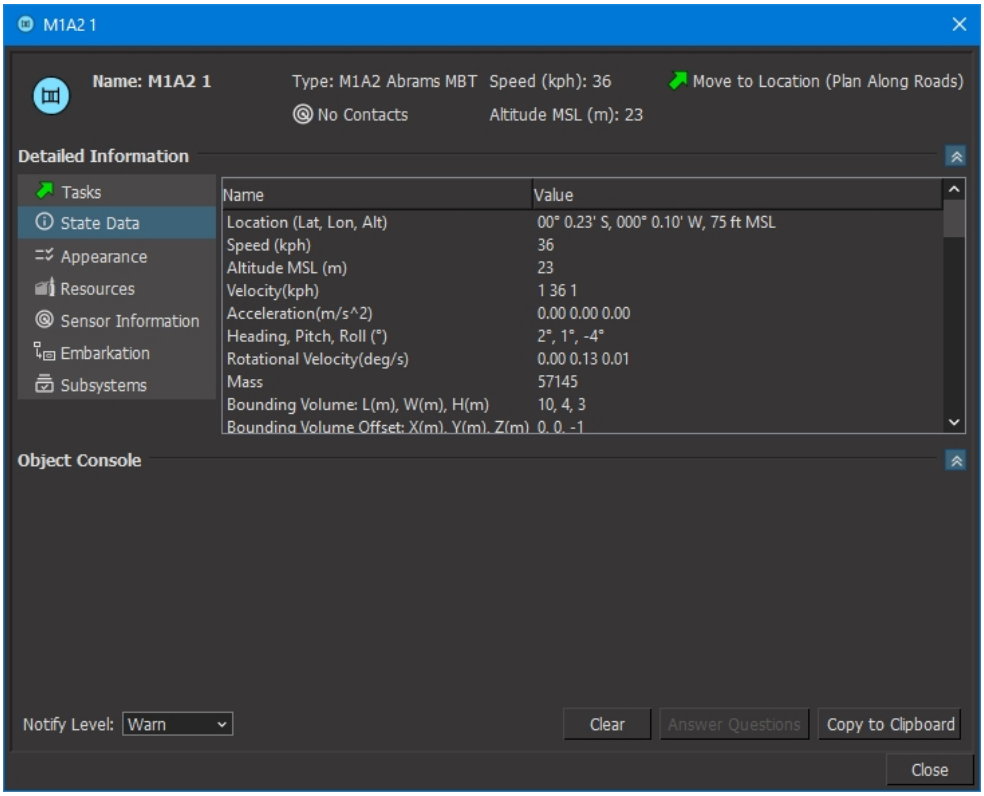
## 5.1.1 Information Dialog Boxes for Entity-Level Scenarios

An information dialog box in an entity-level scenario (Figure 26) has the sections:

- Summary:
  - Simulation object icon
  - Simulation object name
  - Current task
  - Sensor information
  - TAS (true airspeed)
  - Altitude
- Detailed information
- Object console

You can expand and contract the detailed information and object console sections to manage the amount of screen space used by the dialog box.

**Figure 26. Information dialog box for simulation object in entity-level scenario**




The Detailed Information section for simulation objects in an entity-level scenario has these pages:

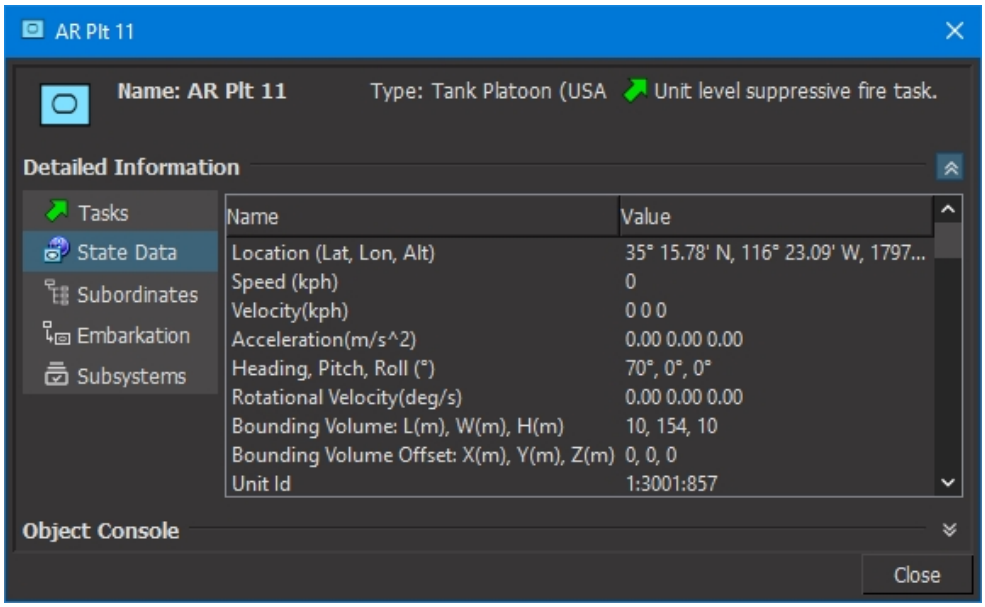
- Tasks. Shows the currently assigned task, subtasks that these tasks have initiated, and any enabled reactive tasks. You can cancel a task that the entity is currently performing by clicking the Cancel icon. You can skip other non-reactive tasks using the context menu. If a task has a graphical status output, there is an additional tab for it in the Tasks page and you can see a graphical display. All scripts that use the FSM packages or the Behavior Engine have graphical status output.
- State Data. Shows location, movement values such as speed and heading, identifying information, and other status information.
- Appearance. Shows the status of appearance attributes, such as lights, firepower, and concealment.
- Resources. Shows that amount of fuel and ammunition available.
- Sensor Information. Shows information about simulation objects being detected.
- Emitters. Shows details about emitters, if applicable.
- Acoustics. Shows details about active sonar and propulsion noise, if applicable.
- Embarkation. Lists embarked objects, if any.
- IFF. Shows IFF information, if applicable.
- Subsystems. Lists the entity's subsystems and indicates whether they are enabled, if applicable.

> The pages listed for Detailed Information are context sensitive. If a page does not apply to an entity type, it is not shown.

The Detailed Information section for units in entity-level scenarios (Figure 27) has the pages:

- Tasks.
- State Data.
- IFF (if applicable).
- Subordinates. Lists the subordinates.
- Acoustics (if applicable).
- Embarkation.

**Figure 27. Information dialog box for units in entity-level scenarios**




### Airspeed for Rotary- and Fixed-Wing Aircraft

The State Data page for rotary-wing and fixed-wing aircraft displays this speed information:

- Speed. Ground speed is the velocity of the aircraft.
- True airspeed. Calculated based on ground speed, temperature, pressure, and altitude.
- Indicated airspeed. Indicated airspeed (IAS) is calculated as 2% less than true airspeed for every 300 ft of altitude.
- Mach.

If you want to substitute your own calculations for airspeed, you can do so using the VR-Forces API.

Indicated airspeed, true airspeed, and mach are user-created state properties that are defined in *./data/simulationModelSets/EntityLevel/vrfSim/platforms/Fixed_Wing_Aircraft.ope* and *./data/simulationModelSets/EntityLevel/vrfSim/platforms/Rotary_Wing_Aircraft.ope*.

```
(DtRwReal IndicatedAirspeed 0.0 publish)
  (DtRwReal TrueAirspeed 0.0 publish)
  (DtRwReal MachNumber 0.0 publish)
```

Therefore, you can change the default value.

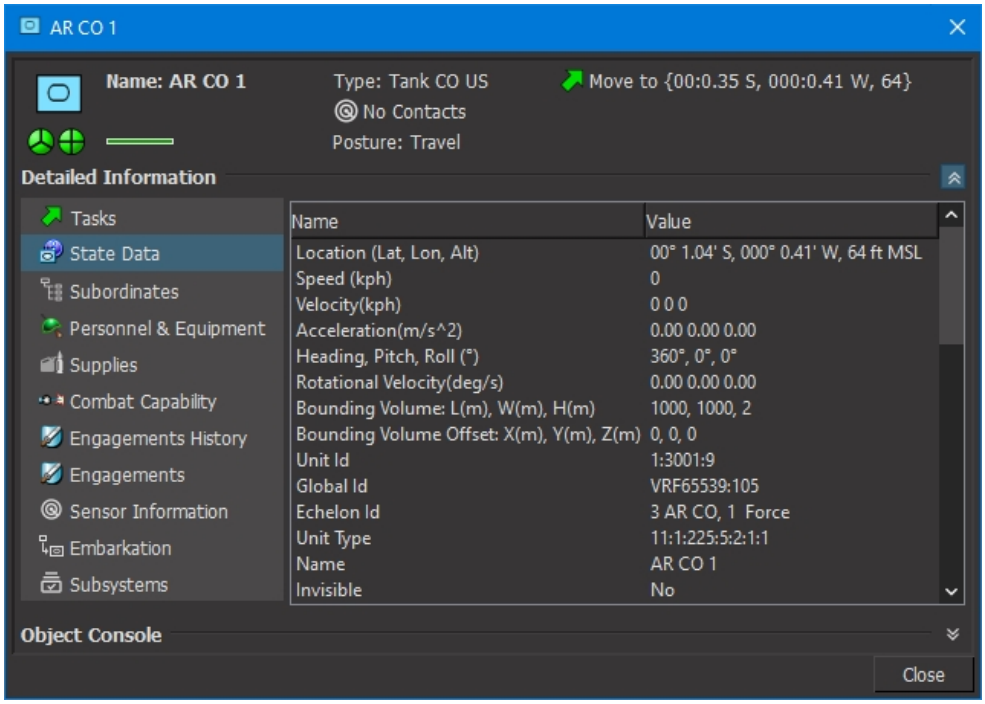
You can change the unit of measurement for true airspeed and indicated airspeed on the Display Settings dialog box, Display Unit Settings page.

### 5.1.2 Information Dialog Boxes for Aggregate-Level Scenarios

An information dialog box for units in an aggregate-level scenario (Figure 28) has the sections:

- Summary:
  - Simulation object icon
  - Simulation object name
  - Current task
  - Sensor information
  - Posture
  - Health status
  - Health and weapon status gumballs
- Detailed information
- Object console

**Figure 28. Information dialog box for simulation object in aggregate-level scenario**




The summary section for units in aggregate-level scenarios (Figure 28) includes a health status bar and gumball icons that display information about the health of the simulation object. Simulation object icons can also display gumballs. (Gumballs are disabled by default. Health status is enabled.)

Figure 29 describes the status represented by different gumball states.

**Figure 29. Gumball charts**


Figure 30 illustrates the health status bar and gumball charts for a simulation object icon.

**Figure 30. Icon with gumballs**


You can expand and contract the detailed information and object console sections to manage the amount of screen space used by the dialog box.

The Detailed Information section for simulation objects in aggregate-level scenarios has the pages:

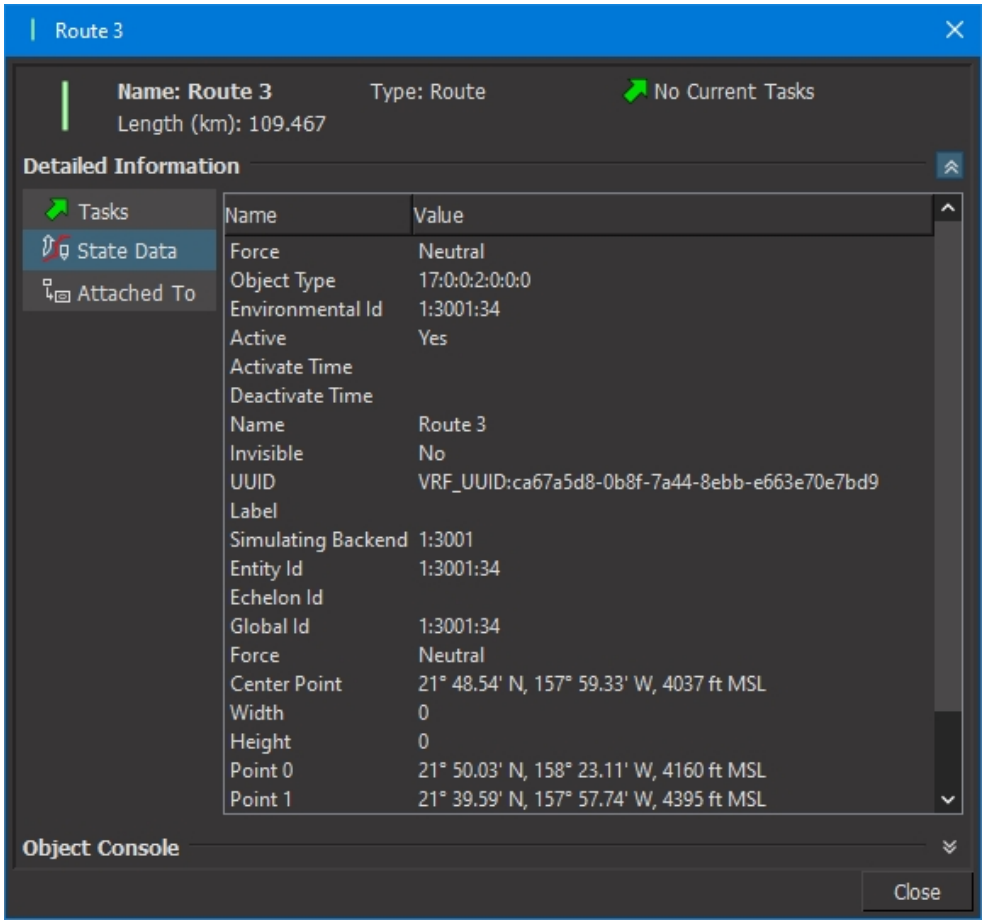
- Tasks.
- State Data.
- IFF.
- Subordinates. Lists the subordinates.
- Personnel and Equipment. The types of personnel and number of each. Casualties, by category.
- Supplies. Food, water, fuel, and ammunition. The Supplies tab displays information when a simulation object is receiving supplies or providing them to another simulation object. When a simulation object is receiving supplies, an up arrow is displayed next to the supply it is receiving.
- Combat Capability. Combat power by category (anti-air, anti-personnel, and so on) and defensive capability, by category.
- Engagements. Current attack and defense status.
- Sensor Information. The Sensors tab shows information about simulation objects being detected. The Jamming tab shows if the simulation object is being jammed (electronic warfare) or is jamming other simulation objects.
- Emitters.
- Acoustics.
- Embarkation.

### 5.1.3 Information Dialog Boxes for Tactical Graphics

The detailed information section for an environmental object (Figure 31), such as a route, has these pages:

- State Data. Shows the force and coordinates.
- Attached To. The simulation object this object is attached to, if any.

**Figure 31. Environmental object information dialog box**




## 5.1.4 Viewing Information for Multiple Objects

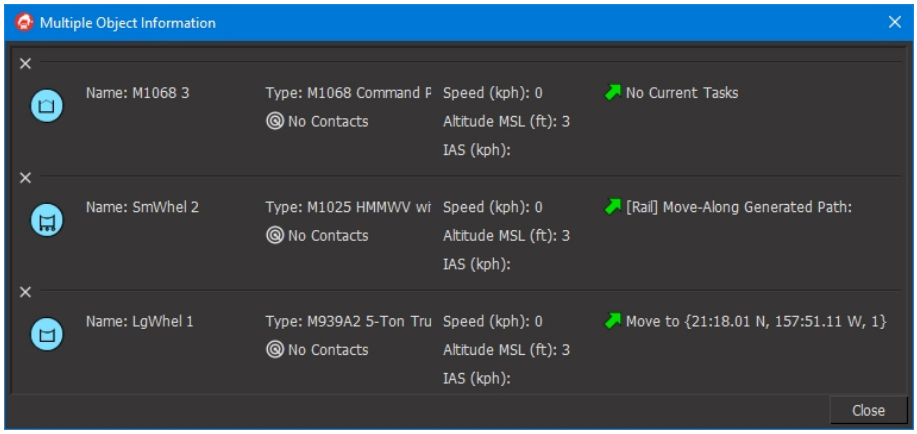
You can select multiple objects and view summary information for all of them. From the summary information window, you can jump directly to an object and you can open its Information dialog box.

If you select more than 20 simulation objects, the Multiple Object Information window displays information about the first twenty and then there is a button at the bottom of the window that allows you to load more objects. You can also close the information for each select object, which automatically loads another object's information.

➤ **To view information about multiple objects:**

1. Select the objects.
2. Choose **Objects > Information**, or press **I**. The Multiple Object Information window opens (Figure 32).

**Figure 32. Multiple Object Information window**




3. To zoom to the extents of an object, click its icon.
4. To open an Information dialog box for an object, click any of the details about the object, such as its name.
5. If you selected more than 20 objects and you want to see more of them in the Multiple Object Information window, click Show Next *x* of *y*.
6. To remove an object from the window, click the ✕ button.

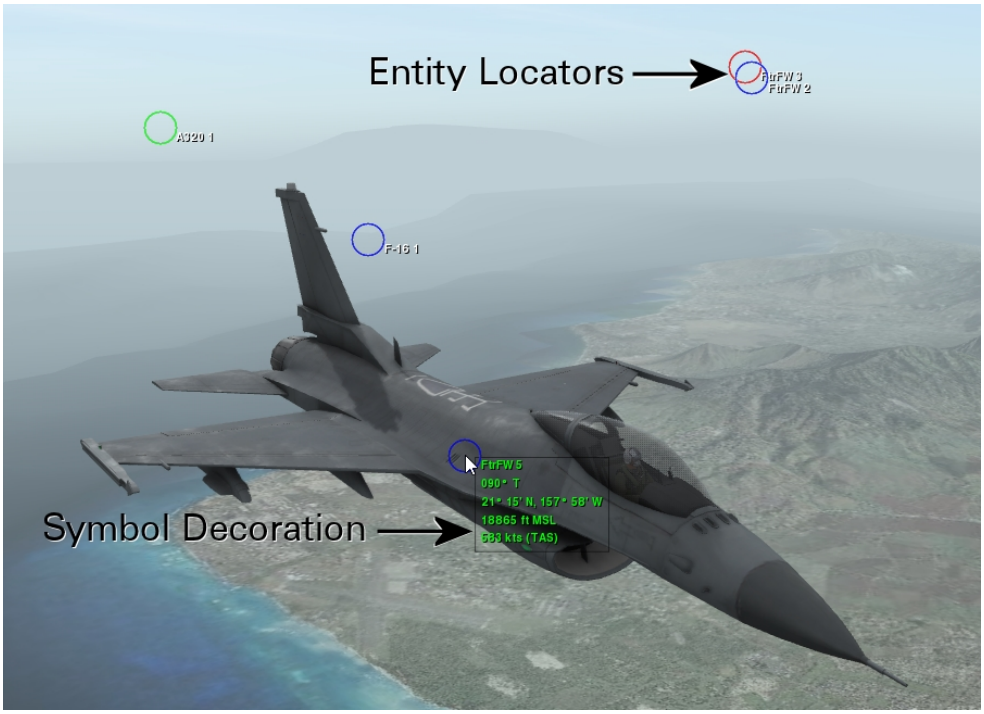
## 5.2. Displaying Entity Labels

VR-Forces provides extensive information about simulation objects in the Information dialog box ("5.1. Displaying Information About an Object" on page 58). It also provides an abbreviated set of information in the form of simulation object labels, also known as symbol decorations.

The symbol decorations that VR-Forces displays for simulation objects help you locate and identify entities (Figure 33). For 3D models, the decoration consists of a locator circle, which is always visible when labels are enabled, and a text label that you can display either all of the time or only when you mouse over an entity. All entities within the view have locator circles, regardless of whether the entity is clearly visible in the window or too far away to see. The color of an entity's label represents its force — the default colors are blue for friendly, red for opposing, and green for neutral.

> ⓘ 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, see "Creating and Editing Model Definitions" in the *Adding Content to MAK ONE Applications Reference Manual*.

**Figure 33. Symbol decorations for 3D models**




Symbol decorations are a set of individually enabled text labels (Figure 34). Entity labels for 2D icons are supported both in the 2D view and the 3D view. If you resize 2D icons, the labels are also resized.

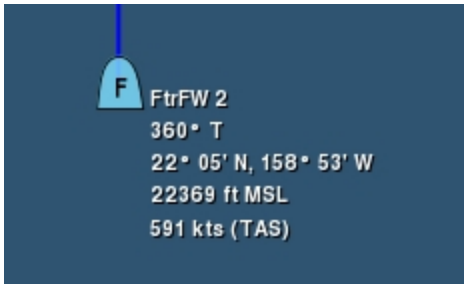
Symbol decorations can display the simulation object's:

- Acceleration
- Altitude above ground level – This displays only for aircraft, space, and lifeform entities.
- Altitude above mean sea level – This displays only for aircraft, space, ground vehicle, subsurface, and lifeform entities (not surface watercraft).
- Contact Radius and Contact Radius Indicator
- Heading and Heading indicator
- Label – If extended labels exist, each label can be displayed separately.
- Location
- Name
- Observed ... – Information observed about the entity.
- Orientation
- Reported By
- Speed – For aircraft, you can also show the ground speed (GS), indicated airspeed (IAS), true airspeed (TAS), and Mach number (MACH).
- TDL ... – Tactical data link information.
- Time of Last Contact

- Time Since Last Report
- Velocity indicator
- Weather – Elevation configuration, cloud state, precipitation intensity, visibility, wind direction, and wind speed.

If spot reports are enabled, you can display labels for spot reports. For details, see "Displaying Labels for Spot Reports".

**Figure 34. Label for a 2D icon**




You can specify the order of symbol decorations. See exampleSymbolDecoration in the *VR-Forces Developers Guide* to learn how to do this.

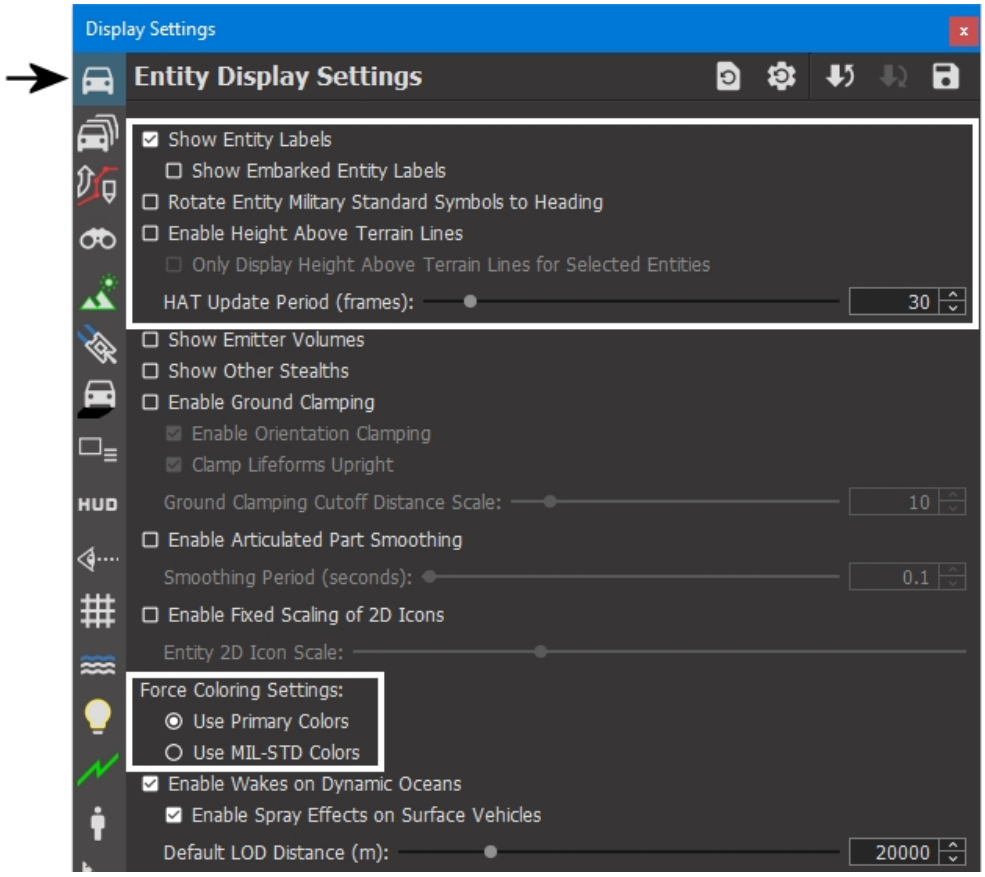
## 5.2.1 Turning Entity Labels On and Off

➤ **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 35).
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 Simulation Object Labels button ( ) on the Display Settings Toolbar.

**Figure 35. Display Settings, Entity Display Settings page**




### Turning Embarked Entity Labels On and Off

You can choose whether to display the labels for embarked entities.

1. Choose **Settings > Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page (Figure 35 above).
3. Select or clear the Show Embarked Entity Labels check box.

> ⓘ You can also enable and disable entity labels by choosing **Settings > Embarked Entity Labels** or by clicking the Embarked Object Labels button ( ) on the Display Settings Toolbar.

## 5.2.2 Pinning Symbol Decorations to the Window

You can pin symbol decorations to the window, making them stay visible after you move the cursor away from the entity. The pin applies to both the 2D and 3D views. The pin icon indicates the state of each entity's label:

- (sideways). The label for each selected entity is not pinned.
- (upright). The label for each selected entity is pinned.
- (angled). Multiple entities are selected and only some of their labels are pinned.

➤ **To pin or hide a symbol decoration in the window:**

1. Right-click an entity's locator circle or the entity name in the Objects List. The context menu displays.
2. Choose **Toggle Pinned Label**.

### Displaying Multiple Symbol Decorations

You can display multiple symbol decorations at one time.

➤ **To display multiple decorations, use either of the following methods:**

- Right-click and choose **Toggle Pinned Label** for each entity whose label or decoration you want to see.
- To display selected symbol decorations at all times for entities, on the Display Settings dialog box, Symbol Decoration Settings page, select the check box in the On/Off column.

## 5.2.3 Symbol Decorations

You can enable and disable the display of symbol decorations by label type. You can also specify which labels should show all of the time, and which should display only when you mouse over an entity. 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.

ⓘ By default, the symbol decorations for unselected entities are hidden when the observer is a certain distance away from them. To adjust that distance, see "Adjusting Display of Symbol Decorations by Altitude" on page 71.

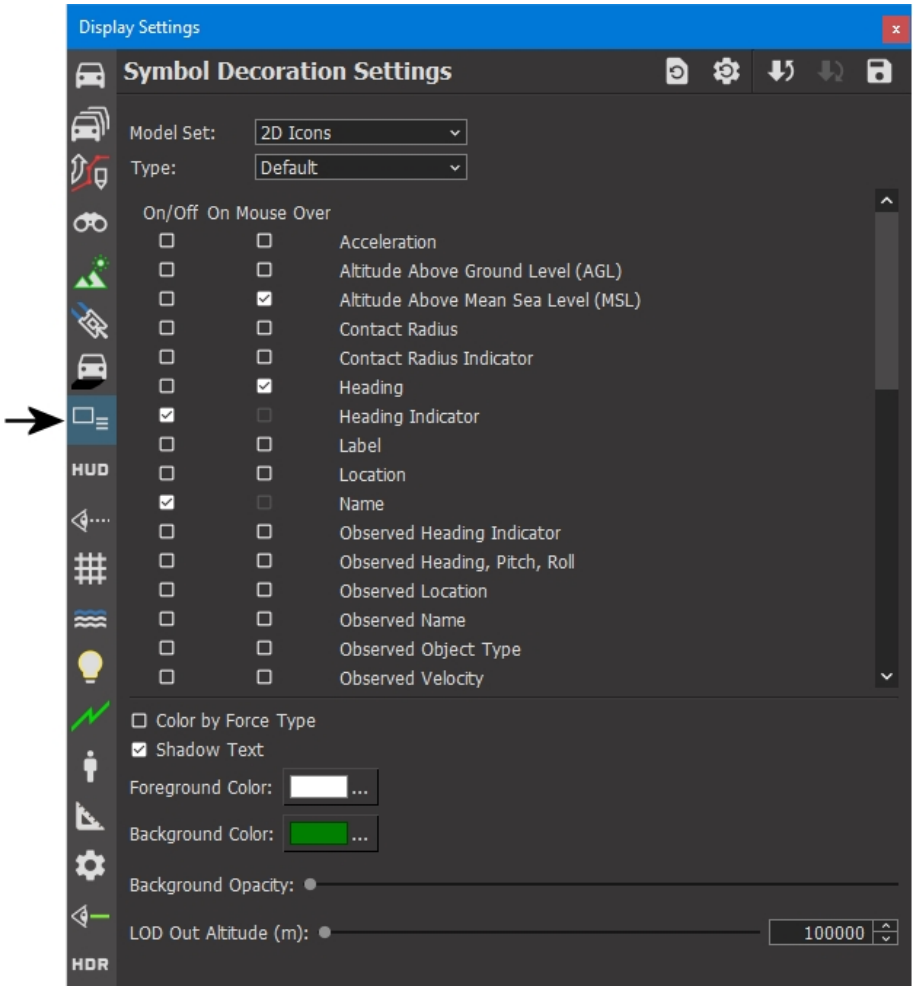
### Displaying Symbol Decorations

You can control what symbol decorations display based on the model set and also whether the labels display all the time or only when you mouse over a simulation object. Note that, when you select the On/Off check box to indicate that the label should display regardless of the mouse cursor's position, the On Mouse Over check box is disabled.

➤ **To enable and disable display of simulation object labels in Display Settings dialog box:**

1. Choose **Settings > Display**. The Display Settings dialog box opens.
2. Select the Symbol Decoration Settings page (Figure 36).

**Figure 36. Symbol Decoration Settings page**




3. Select the model set where you want the labels to display.
4. Select the check box for each label that you want to display.
   - To display the label all of the time, select the check box in the On/Off column.
   - To display the label only when the cursor is over a simulation object, select the check box in the On Mouse Over column.

   Clear the check box for each label that you do not want to show.

## Changing the Color for Label Text

Symbol decoration text uses 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 labels:**

1. Choose **Settings > Display**. The Display Settings dialog box opens.
2. Select the Symbol Decoration Settings page (Figure 36 above).

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.

### Adjusting Display of Symbol Decorations by Altitude

Symbol decorations are hidden when the observer rises to a specified altitude. The default is 100,000 meters, but you can change it. If you hover the mouse over or select a simulation object, the symbol decorations display regardless of the observer's altitude. This does not affect heading or velocity indicators.

➤ **To adjust the altitude at which symbol decorations are hidden:**

1. Choose **Settings > Display**. The Display Settings dialog box opens.
2. Select the Symbol Decoration Settings page (Figure 36 on the previous page).
3. Use the slider or the text box to adjust the LOD Out Altitude.

> 2D icons can also be resized based on the observer's altitude. See "Scaling 2D Icons by Altitude".

### Repositioning Labels

By default, text labels in the 2D view display below and to the right of simulation object icons. You can rotate the labels to other offset positions for a simulation object if they are cluttered together and you want to be able to read them clearly. The label can be in one of four positions: above and to the right, above and to the left, below and to the left, or below and to the right.

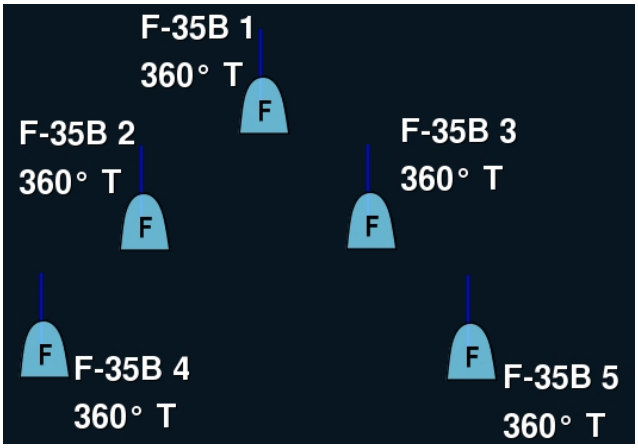
➤ To reposition a label, select the entity and then do one of the following:

- Open the context menu and then choose **Rotate Label**.
- Press **Ctrl**+**Shift**+**S**.

Repeat until the label is where you want it.

> You can select multiple entities and reposition their labels simultaneously.

**Figure 37. Repositioned entity labels**




## 5.2.4 Configuring Shadowing and Background Color for 2D Labels

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 label text. This may improve visibility when simulation objects are on light colored terrain.

**Figure 38. 2D label with background**


➤ **To set the color and opacity of the label background:**

1. Choose **Settings > Display**. The Display Settings dialog box opens.
2. Select the Symbol Decoration Settings page (Figure 36 on page 70).
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 labels:**

1. Choose **Settings > Display**. The Display Settings dialog box opens.
2. Select the Symbol Decoration Settings page (Figure 36 on page 70).
3. Select or clear the Shadow Text check box.

## 5.3. Using Extended Labels

One of the ways to identify a simulation object is by its label. (For more information, see "2.2.5 Labels" on page 6.) Extended labels are additional labels that you can specify on an application-wide basis and define individually for simulation objects. The contents of each label is sent over the network as part of the simulation object's information and is displayed in its Information dialog box.

You can add and remove extended labels and change the label name. Changing a label's name does not affect the text that is associated with the label for those simulation objects that use the label.

> Extended labels are not scenario-specific. They affect all scenarios. Therefore, if you change a label's name or index, or delete it, be sure that you want this change to apply to all scenarios in which you use the extended label.

Figure 39 shows an extended label (called A Special Label) added in the Application Settings dialog box. Figure 40 shows the text for the extended label for a specific entity entered in the Edit dialog box and then displayed on the screen.

**Figure 39. Extended label added to session**

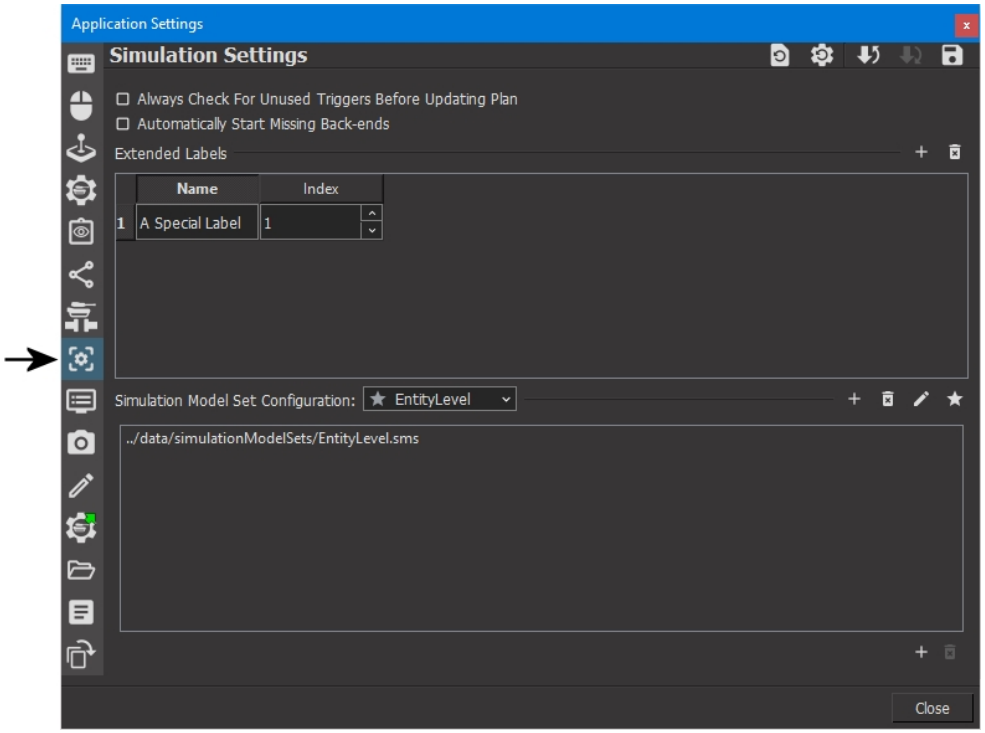
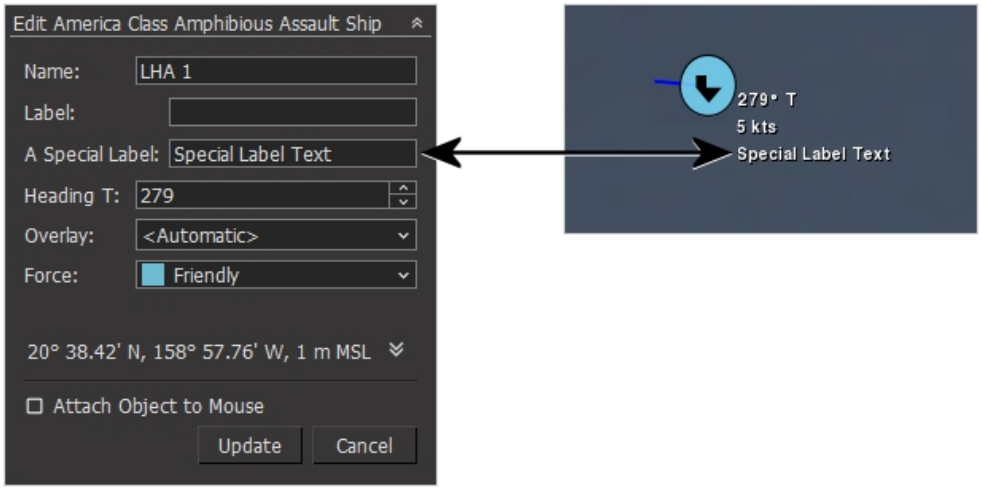


**Figure 40. Extended label text for a simulation object**




You can display extended labels in Plan View mode the same as other symbol decorations. For more information, see "5.2.3 Symbol Decorations" on page 69.

## 5.3.1 Adding an Extended Label

➤ **To add an extended label:**

1. Choose **Settings > Application**. The Application Settings dialog box opens.
2. Select the Simulation Settings page (Figure 39).
3. Click the Add button (+) on the Extended Labels line. A label is added to the list window. It has a generic name (for example, Label 1).
4. Double-click the label name to make it editable.
5. Type the name you want for this label.
6. Edit simulation objects to specify the text for this label.

## 5.3.2 Changing an Extended Label's Index

A label's index is used to store the label text and identify it when sent over the network. In Edit dialog boxes, extended labels are listed in indexed order. You can change an extended label's index. However, while changing an extended label's name does not affect the text that you save for a simulation object, changing its index could affect existing scenarios. Therefore, use caution in doing so.

➤ **To change an extended label's index:**

1. Choose **Settings > Application**. The Application Settings dialog box opens.
2. Select the Simulation Settings page (Figure 39 on the previous page).
3. Click the up or down arrow next to a label's index number. The index number changes and, if applicable, the label moves up or down in the list.

### 5.3.3 Deleting an Extended Label

Deleting an extended label affects all scenarios that use that label. (In other words, it is not a scenario-specific setting.) If you delete an extended label, it is immediately removed from all simulation objects that are using that label.

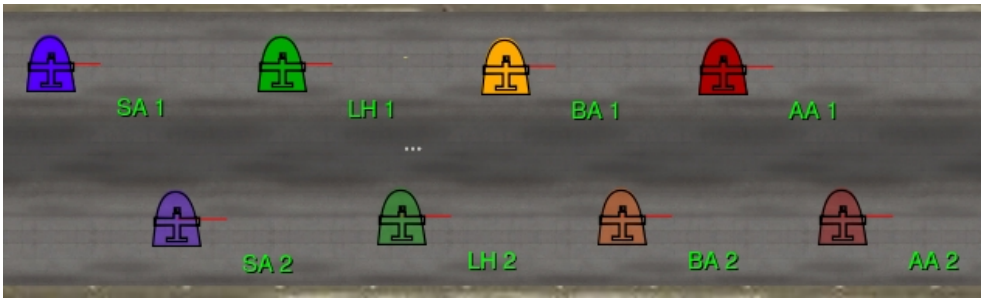
➤ **To delete an extended label:**

1. Choose **Settings > Application**. The Application Settings dialog box opens.
2. Select the Simulation Settings page (Figure 39 on page 73).
3. In the Extended Labels window, select the label you want to delete.
4. Click the Delete button ( ).

## 5.4. Customizing 2D Icon Colors

During a session, you can adjust the colors and transparency of the 2D icons. These changes only apply to the current session and VR-Forces GUI; they are not saved and they are not visible in other GUIs. You can use this to categorize the icons, for example, to differentiate the simulation objects that you are working with from those you are not.

**Figure 41. 2D icons with different color and transparency settings**




➤ **To change the color and transparency of a 2D icon:**

1. Select the simulation object.
2. Choose **Objects > Decorations > Set Icon Color**. The Set Icon Color dialog box opens.
3. Adjust the color and transparency (alpha channel) settings as desired.
4. Click OK.

## 5.5. Changing Force Colors

By default, labels, detonation lines, and other force-related simulation object graphics (including the 3D Colorized Model model set in VR-Forces) 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 pink (light red)).

You can also change the color of a force in the Simulation Object Editor. For details, see "Editing a Force".

➤ **To change force coloring:**

1. Choose **Settings > Display**. The Display Settings dialog box opens.
2. For entities and entity graphics, select the Entity Display Settings page (Figure 35 on page 68).

   For units, select the Unit Display Settings page (Figure 2).
3. Select the Force Coloring Settings option that you want.

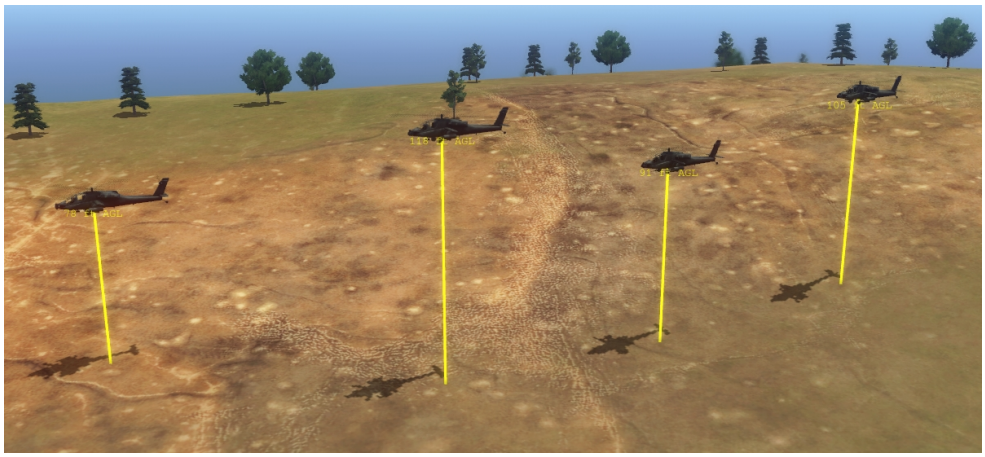
## 5.6. Displaying HAT Lines for Entities (3D Only)

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 42). They help you understand the relationship of the entity to the terrain.

HAT lines are shown only for entities that could be above the ground, such as fixed- and rotary-wing vehicles. You can show and hide these lines globally or you can specify that they display only for the selected entities.

As you create or edit an entity, VR-Forces displays HAT lines even if they are otherwise disabled.

**Figure 42. 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 35 on page 68).
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-Forces, you can also enable and disable entity HAT lines by choosing **Settings > Entity Height Above Terrain Lines** or by clicking the Entity Height Above Terrain Lines button ( ) on the Display Settings Toolbar.

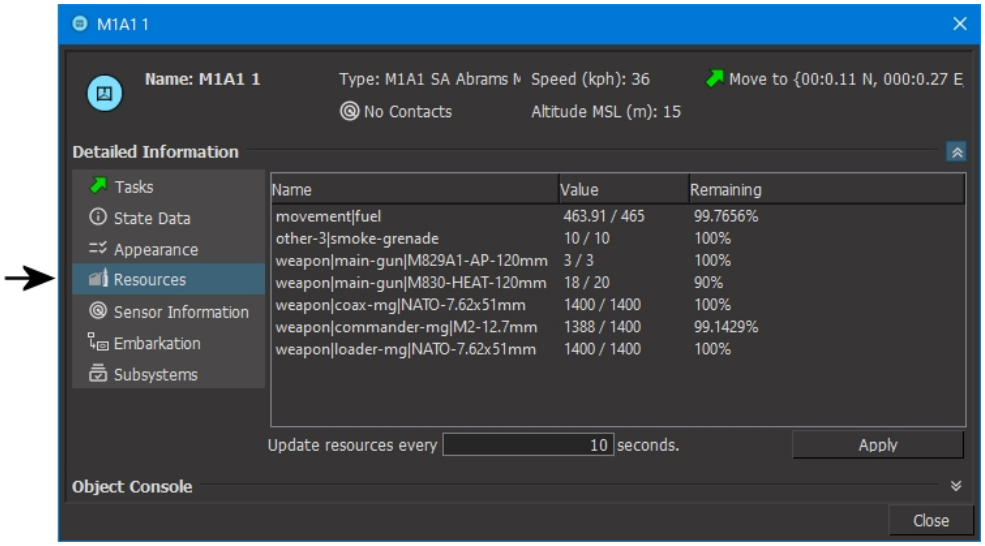
## 5.7. Displaying Entity Resources

VR-Forces tracks the amount of resources that a simulation object has. You can view the status of resources in the Information dialog box.

### 5.7.1 Viewing Numerical Resource Data

The Information dialog box lists the simulation object's current amount of resources, its starting amount, and the percentage remaining (Figure 43).

**Figure 43. Resource Information page**



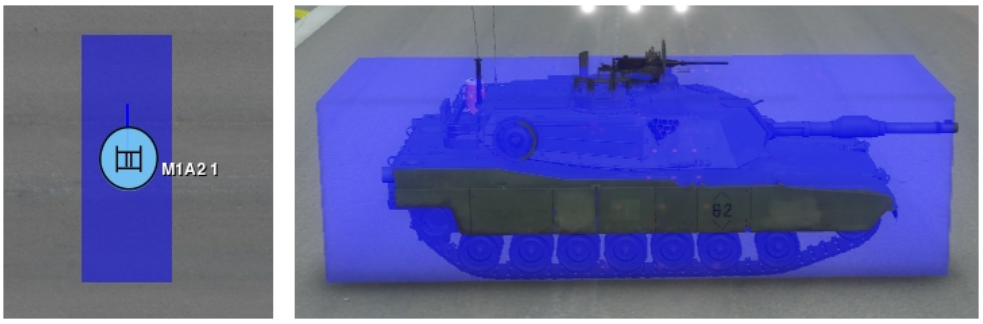

➤ **To view a simulation object's resource status:**

1. Select the simulation object.
2. Choose **Objects > Information**, or press **I**.
3. Expand the Detailed Information section.
4. Select the Resources page.

## 5.8. Displaying Bounding Volumes

You can display bounding volumes for simulation objects. Bounding volumes show the area around a simulation object that the sim. engine uses for intersection calculations. Bounding volumes are shown as translucent volumes in 3D and translucent rectangles in 2D (Figure 44). For simulation objects in aggregate-level scenarios, the bounding volume is equivalent to the footprint.

**Figure 44. Bounding volumes - 2D and 3D**




Bounding volumes for units and bounding volumes for individual entities are managed by different settings. For information about displaying bounding volumes for units, see "Displaying Unit Icons and Bounding Volumes".

## 5.8.1 Enabling and Disabling Bounding Volumes for Entities

Display of bounding volumes for entities is an observer setting.

➤ **To enable or disable bounding volumes for individual entities:**

1. Choose **Settings > Display**. The Display Settings dialog box opens.
2. Select the Observer Settings page.
3. Select the observer mode for which you want to enable or disable bounding volumes.
4. Select or clear the Bounding Volume check box.

ⓘ You can also enable and disable bounding volumes for entities by choosing **Observer > Bounding Volumes** or clicking the Bounding Volumes button (▣) on the Observer Settings Toolbar.

## 5.8.2 Showing Bounding Volumes Only for Selected Entities

You can configure VR-Forces so that it shows bounding volumes only for selected simulation objects.

➤ **To display bounding volumes only for selected entities:**

1. Enable bounding volumes, as described in "5.8.1 Enabling and Disabling Bounding Volumes for Entities" above.
2. Choose **Settings > Display**. The Display Settings dialog box opens.
3. Select the Entity Display Settings page.
4. Select the Show Bounding Volumes for Selected Entities check box.

# 5.9. Configuring Object Console Messages

The Information dialog boxes for simulation objects and tactical graphics have a console that displays messages sent from the simulation engine, from a simulation object's plan, from other simulation objects, and from scripts. You can set the notification level that controls the display of these messages. You can save them to the clipboard, and you can clear the display. These messages are also sent to the Object Console Summary Panel.

Scripted tasks can ask questions. You can answer these questions from the Object Console Summary Panel.

## 5.9.1 Setting the Notification Level for Console Messages

Console messages are displayed only if they match or exceed the notification level set for that object. The notification levels are:

- 0 – Only fatal messages are printed.
- 1 – Warning messages and fatal messages are printed.
- 2 – Some diagnostic information is printed with warnings and fatal messages.
- 3 – Verbose information is printed.
- 4 – Debug information is printed.

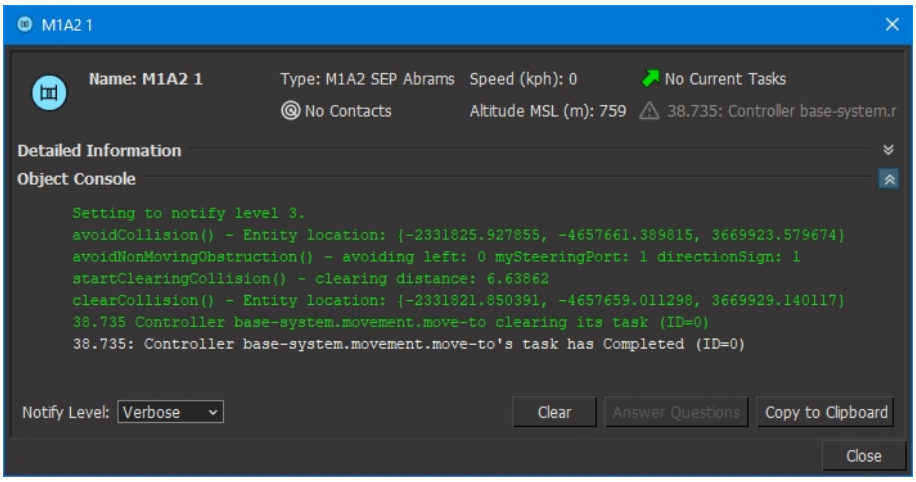
The default notification level is 2.

The notification level is set individually for each object. You can set the notification level in the object's Information dialog box, with the Notify Level set data request (for simulation objects), or with the **objectConsoleNotifyLevel** parameter in *vrfSim.mtl*. (For the set data request procedure, see "Notify Level".)

➤ **To set the notification level for an object's console in the Information dialog box:**

1. Select the simulation object.
2. Choose **Objects > Information**. The Information dialog box opens.
3. Expand the Object Console section (Figure 45).

**Figure 45. Information dialog box, Object Console**



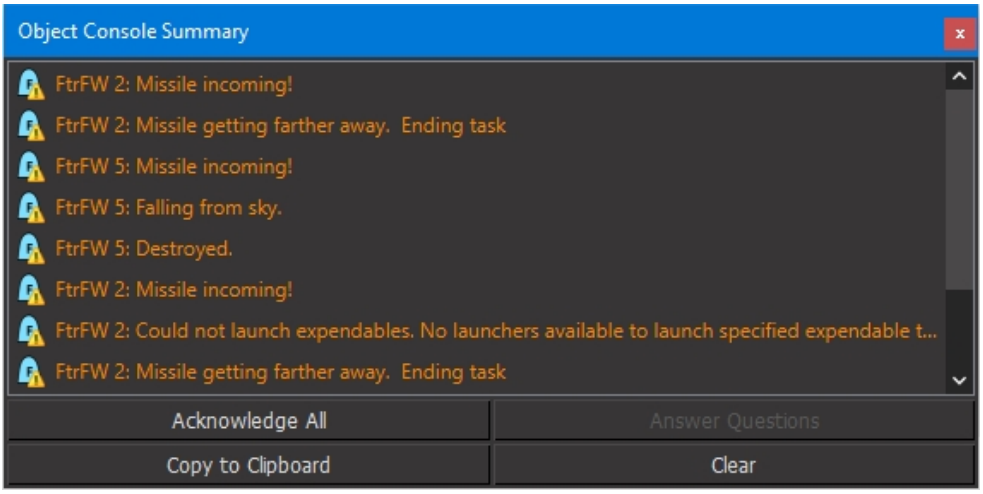

4. Select a notification level from the Notify Level list.

## 5.9.2 Viewing All Console Messages

You can view all object console messages in one place – the Object Console Summary Panel. Messages are displayed based on the notification levels of the individual simulation objects in the scenario. You can set the notification level in a simulation object's Information dialog box or with the Notify Level set data request.

When VR-Forces sends a message to a simulation object's console, it displays an exclamation point next to the entity until you view the message (unless you disable this feature). The messages in the Object Console Summary Panel also display an exclamation point until you open the Information dialog box for that entity or click Acknowledge All.

**Figure 46. Object Console Summary Panel**



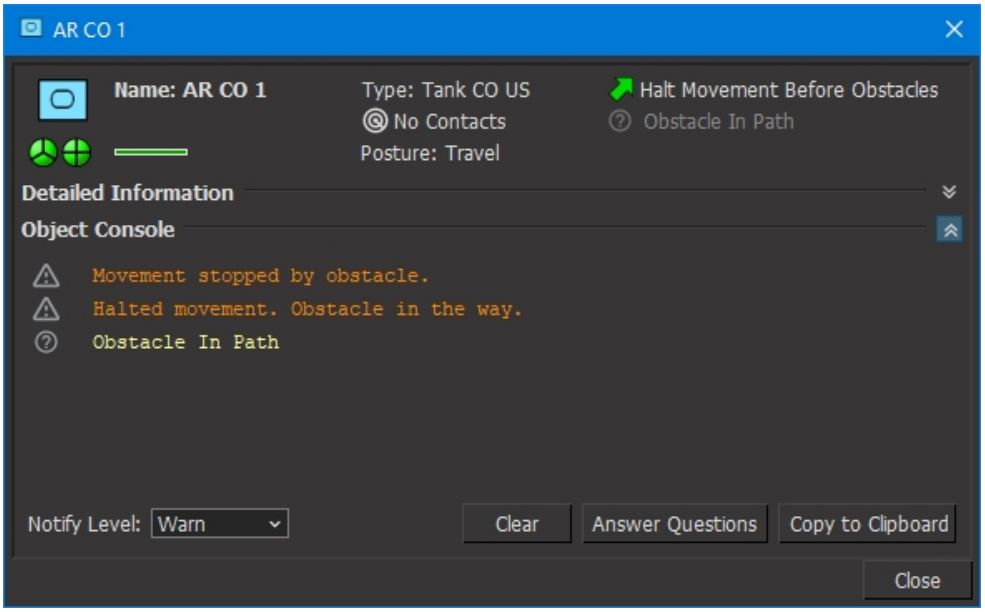

➤ To display the Object Console Summary Panel, choose **View > Object Console Summary Panel**.

➤ To remove the exclamation point from messages in the Object Console Summary Panel, click Acknowledge All.

## 5.9.3 Answering Questions from Scripted Tasks

Scripted tasks can send questions to the object console. Questions are indicated by a question mark next to the message (Figure 47). The script offers a set of responses as buttons, option buttons, or a list. You can answer questions from the Object Console and the Object Console Summary Panel.

**Figure 47. Question in console**




➤ **To answer a question from a scripted task:**

1. On the Object Console or Object Console Summary Panel, click Answer Questions. A dialog box with the possible answers displays.
2. Choose an option.
3. Click OK.

## 5.9.4 Saving Console Messages

If you want to save the messages that are sent to the console, you can copy them to the clipboard and then paste them into a file.

➤ **To save console messages to the clipboard:**

1. Select the simulation object.
2. Choose **Objects > Information**. The Information dialog box opens (Figure 45 on page 79).
3. If the Object Console is collapsed, expand it.
4. To save all messages in the console, click Copy to Clipboard. To save specific messages, select them and click Copy to Clipboard.
5. Paste the text into the text editor or word processor of your choice.

## 5.9.5 Displaying an Object Console Warning Icon

VR-Forces can display a notification icon when an object receives a message on its object console (Figure 48). This option is enabled by default, however the default notification level is Warn and objects do not typically get messages at this level. The more likely use for this icon is to set the notification level to Info or lower and then check the console when a notification is received. For details about how to set the notification level, see "5.9.1 Setting the Notification Level for Console Messages" on page 79.

**Figure 48. Object console warning icon**


Icons are displayed next to the entity icon in the display window, next to the entity's listing in the Objects List Panel, and, if the entity has recently been selected, on the Last Selected Object Panel.

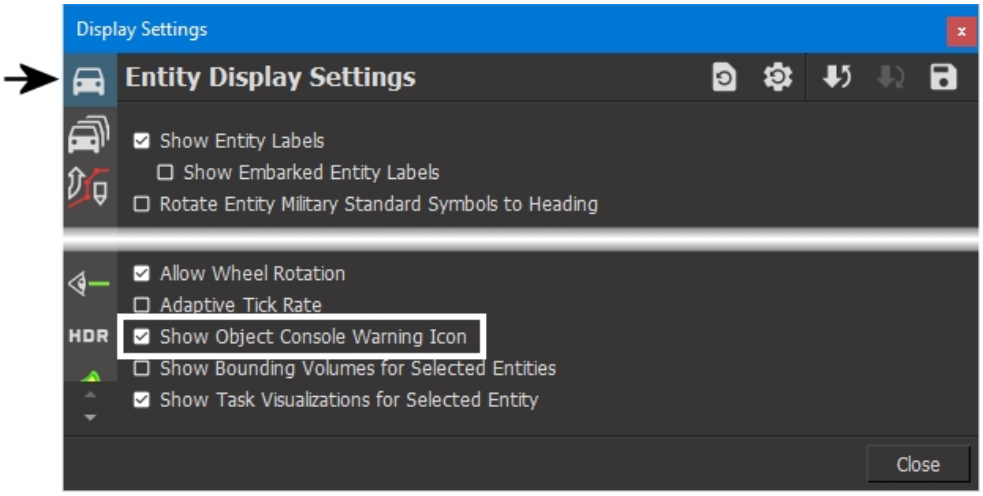
The icon is not displayed if the Information dialog box is open. If an icon is displayed for an object, when you open its Information dialog box, the icon is removed.

If you disable the warning icon, it no longer displays in the display window or the Objects List Panel. It still displays on the Last Select Object Panel, if the entity was the last selected object.

➤ **To enable or disable the console warning icon:**

1. Choose **Settings > Display**. The Display Settings dialog box opens.
2. Select the Entity Display Settings page.
3. Select or clear the Show Object Console Warning Icon check box.

**Figure 49. Entity Display Settings, Show Object Console Warning Icon option**



### 5.9.6 Clearing the Object Console

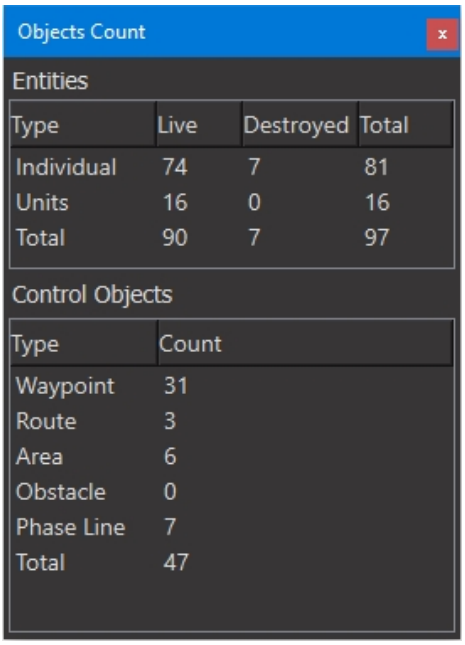
➤ To delete messages from the object console, click Clear.

## 5.10. 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 50).

**Figure 50. Objects Count Panel**



# 6. Embarked and Embedded Entities

This chapter explains how to embark entities on other entities. It also explains how to manage embedded entities, which is another way to embark entities.

# 6.1. Embarking and Disembarking Simulation Objects

Embarkation places simulation objects into or onto other simulation objects, for example, dismounted infantry in a truck, trucks on an LCAC, or aircraft on an aircraft carrier. While embarked, the simulation objects travel with the simulation object on which they are embarked. Embarked simulation objects are not visible in the 2D view. They are visible in the 3D view if they are embarked on the outside of the parent, such as on the deck of an aircraft carrier, or inside a parent that has interior geometry, such as some helicopters.

Embarked entities can shoot while they are embarked if they are embarked on an entity that has object geometry enabled.

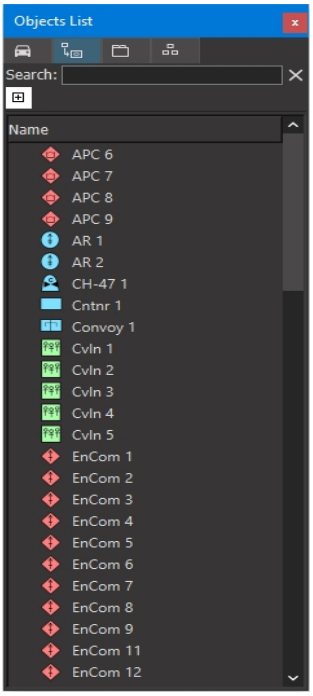
> ⚠ If you are using HLA and want to use the embarkation feature, you must use RPR FOM 2, draft 17 or later. VR-Forces includes configurations in the Simulation Connections Configuration dialog box that start VR-Forces using the correct FED file and FOM Mapper for HLA Evolved using RPR FOM 2.

You can attach control objects to simulation objects so that embarked simulation objects can use them to move about on the parent simulation object. For example, an aircraft could taxi towards a waypoint on the flight deck of an aircraft carrier.

> ⓘ Attached objects are shown in the Embarkation View on the Objects List Panel.

The embarkation status of simulation objects is displayed in the Embarkation View of the Objects List Panel (Figure 51).

**Figure 51. Embarkation View**



You can embark and disembark simulation objects instantly (described in this section), or you can have them simulate embarkation and disembarkation using the Embark and Disembark tasks, described in "Embarkation, Wait, Radio, and Other Tasks".

When you embark a simulation object using the Embark task, the simulation object can embark only on simulation objects that are configured to accept that type of simulation object. For example, LCACs are configured to accept several types of ground vehicles. You cannot give a DI an Embark task to embark on an LCAC. However, when you use the immediate embark method to embark a simulation object, you can use preconfigured slots (spaces) for simulation objects or specify an override. When you specify an override you can embark a simulation object on any other simulation object at the coordinates you enter. However, it is your responsibility to know what the proper coordinates are.

> ⓘ Embarked entities controlled by joystick cannot disembark. For example, if you use a joystick to control a fixed-wing taking off from a carrier, the aircraft will take off, but it will remain in an embarked state.

## 6.1.1 Embarking and Disembarking Simulation Objects Instantly

You can embark and disembark simulation objects instantly by using:

- The Embark On command.
- Set data requests.
- The Embarkation View.
- Automatic embarkation.

Using set data requests lets you include immediate embarkation in plans. This may be particularly useful in a global plan, in which you may want to quickly create and embark simulation objects without paying attention to the realism of the action. For details about using set data requests for embarkation, see "Embarked" and "Disembarked".

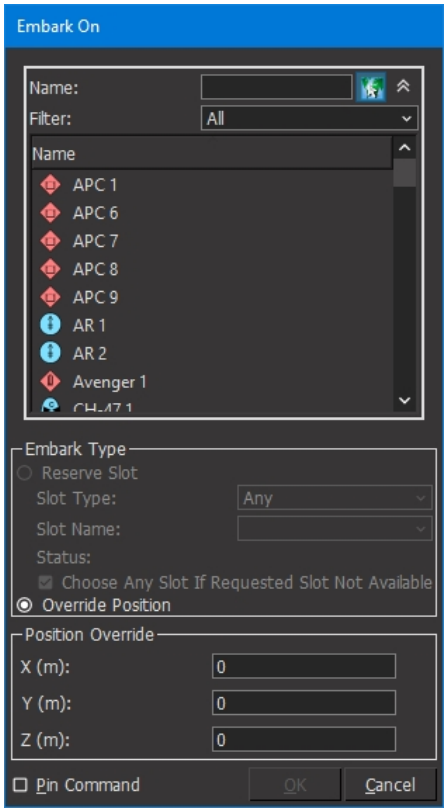
### Embarking Simulation Objects Instantly Using the Embark On Command

The Embark On command lets you embark simulation objects using preconfigured embarkation spaces or using specific coordinates. If the embarkation spaces on the parent simulation object are full, VR-Forces sends a message to the sim. engine console and the simulation object is not embarked. To embark simulation objects instantly from a plan, use the Embarked set data request.

➤ **To embark a simulation object instantly using the Embark On command:**

1. Select the simulation object that you want to embark.
2. Choose **Objects > Embarkation > Embark On**. The Embark On dialog box opens.

**Figure 52. Embark On dialog box**




3. Optionally, filter the list of potential parent simulation objects.
4. Select the parent simulation object.
5. To embark in a configured space, select the Reserve Slot option. To specify an embarkation point, select Override Position.
6. If you chose the Override Position option, specify the coordinates on the parent, in body coordinates, for the embarkation point.
7. Click OK.

### Embarking Simulation Objects in the Embarkation View

➤ **To embark an object in the Embarkation View:**

1. On the Objects List Panel, select the Embarkation View.
2. Drag the simulation object you want to embark onto the parent simulation object. The Embarked dialog box opens.
3. To specify an embarkation location, specify X, Y, and Z values. If you do not specify a location, the simulation object embarks at the center of the parent simulation object.
4. Click OK.

### Embarking Simulation Objects Using Automatic Embarkation

Automatic embarkation lets you embark simulation objects as you create them. It has these conditions:

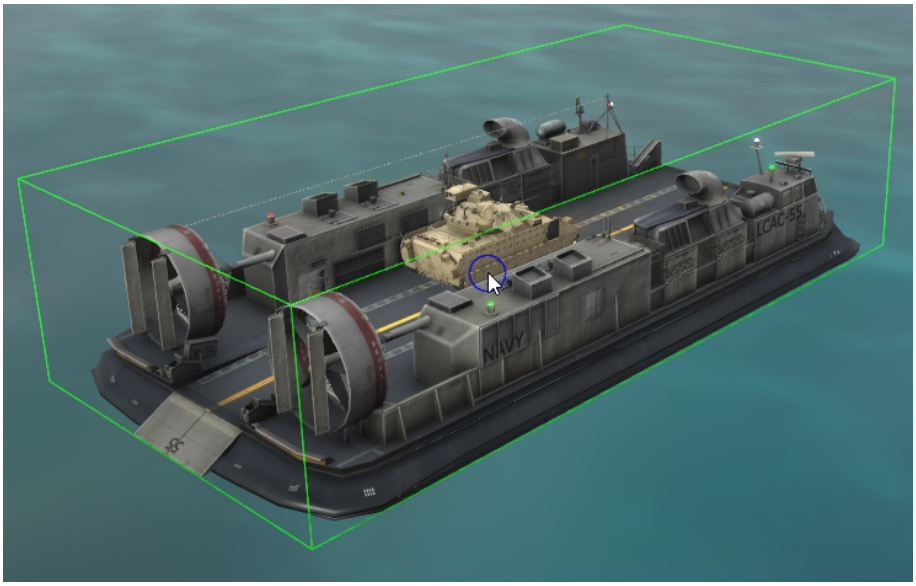
- It is only available in 3D observer modes.
- The simulation object you want to embark must be capable of embarking (as configured in the OPD).
- The simulation object you want to embark on must allow embarkation.
- There is no option to override embarkation slots by specifying a particular location.
- If slots are configured on the parent simulation object and you select the option to use slots, embarked simulation objects are placed in slots until the slots run out. Then they are placed on the parent simulation object where you click.

> ⓘ You can attach tactical graphics to simulation objects using this method. For more information, see "Attaching a Tactical Graphic Automatically".

➤ **To embark a simulation object using automatic embarkation:**

1. Create the entity on which you want to embark another entity (if it is not already created.)
2. In the Create Object Palette, select the entity that you want to embark.
3. Optionally, configure auto-embarkation:
    a. If the Create Entity dialog box is not open, click its tab to open it.
    b. To enable or disable automatic embarkation, select or clear the Enable Automatic Embarkation check box.
    c. To embark the entity in a slot if one is available, select the Embark in Slot if Available check box.
4. Move the cursor over the entity you want the new entity to embark on. If it supports embarkation and automatic embarkation is enabled, a green box is displayed (Figure 53). (If the entity does not support automatic embarkation, no green box is displayed.) If the create dialog box is open, the Not Embarked label changes to Embark On *entity_name*.

**Figure 53. Auto embarkation**




5. Click to place the entity.

## 6.1.2 Disembarking Simulation Objects Using the Disembark Command

Like the Embark On command, the Disembark command is typically used as part of scenario development. To simulate disembarkation, you would use the Disembark task. To provide instant disembarkation in a plan, you would use the Disembarked set data request.

ⓘ Embarked entities controlled by joystick cannot disembark. For example, if you use a joystick to control a fixed-wing taking off from a carrier, the aircraft will take off, but it will remain in an embarked state.

➤ **To disembark a simulation object instantly:**

1. Select the simulation object that you want to disembark.
2. Choose **Objects > Embarkation > Disembark**. If the simulation object was embarked using the default embarkation point, it is disembarked next to the former parent simulation object. If the simulation object was embarked using a position override, it is disembarked in the same location as the former parent.

➤ **To disembark a simulation object in the Embarkation View:**

1. Open the Embarkation View.
2. Drag the simulation object you want to disembark from the parent simulation object to the force level.

### Disembarking All Simulation Objects

When you disembark all simulation objects from a parent, the placement of the simulation objects depends on how they were embarked. If the simulation objects were embarked with the Embark On command or the Embark task, they are disembarked to separate

locations near the parent. If they were embarked with the Embarked set data request, they are all disembarked in the same location in the center of the parent simulation object. Since this behavior is usually not desirable, you should understand how you embarked the simulation objects before you use this command.

➤ **To disembark all simulation objects from a parent entity:**

1. Select the parent simulation object.
2. Choose **Objects > Embarkation > Disembark All**.

## 6.2. Embedded Entities

Embedded entities are a convenient way to manage entities that are typically embarked on other entities.

Ships, planes, and vehicles often carry other platforms or personnel. You can simulate this relationship by embarking entities on other entities. However, this involves creating the embarking entities, giving them plans, embarking them, disembarking them, and otherwise managing the relationship. This can be inconvenient if you need to simulate large numbers of embarked entities, such as the airplanes and helicopters on an aircraft carrier or multiple personnel on a personnel carrier.

The embedded entity feature lets you configure entities with a set of embedded entities. You can then deploy the embedded entities and give them tasks or plans. Once deployed, there is no difference between an embedded entity and any other entity, except that a parent entity can recover a deployed entity. You do not need to manage the initial process of creating the relationship between the entities via embarkation and disembarkation.

For details about deploying and recovering embedded entities, see "Embedded Entity Tasks".

### 6.2.1 Configuring an Entity to Support Embedded Entities

Embedded entities are configured in systems, which you then add to an entity in the Simulation Object Editor. An entity can have only one embedded entity system. Therefore, if you need different configurations of embedded entities for different parent entities, you must create a system for each configuration.

Within a configuration, you must specify each embedded entity. In other words, you cannot specify a system as containing, for example, 5 helicopters and 40 F/A-18 jets. In this case the system would have to individually specify each helicopter and each jet.

A system definition file for embedded entities specifies the entities and the metadata needed to edit properties in the Simulation Object Editor. You must give each entity a unique name.

An entity is formatted as follows (from *./data/simulationModelSets/EntityLevel /vrfSim/systems/other/ddg-embedded-support-craft.sysdef*):

```
(embedded-entities
  (ee-1
    (entity-name  "SH-60 Seahawk Alpha")
    (entity-type  1 (1 2 225 22 3 1 0))
    (deployment-time 0.0)
    (recovery-distance 35.0)
```

```
      (relative-position $rel-pos-lamps-a)
      (relative-heading $rel-heading-lamps-a)
    )
    ...
    (ee-2
      (entity-name  "RHIB Alpha")
      (entity-type  1 (1 3 225 63 8 0 0))
      (deployment-time 15)
      (recovery-distance 50.0)
      (relative-position $rel-pos-rhib-a)
      (relative-heading $rel-heading-rhib-a)
    )
    ...
  )
```

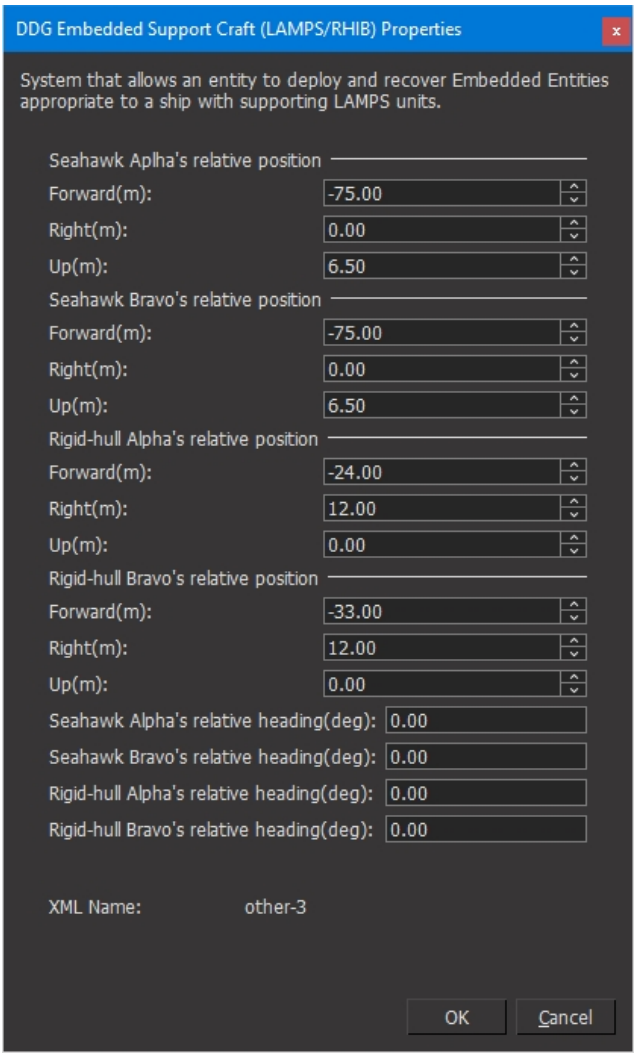
The **relative-position** and **relative-heading** parameters are variables that link to the metadata:

```
  (parameter-data-list
    (vector-parameter-data
      (parameter-name  "rel-pos-lamps-a")
      (variable-type  "DtRwVector")
      (display-label  "Seahawk Alpha's relative position")
      (display-units  "meters")
      (source-units  "meters")
      (default-value  -75.000000 0.000000 -6.500000)
      (relative-to  "")
    )
    (vector-parameter-data
      (parameter-name  "rel-pos-rhib-a")
      (variable-type  "DtRwVector")
      (display-label  "Rigid-hull Alpha's relative position")
      (display-units  "meters")
      (source-units  "meters")
      (default-value  -75.000000 10.000000 -6.500000)
      (relative-to  "")
    )
    ...
    (real-parameter-data
      (parameter-name  "rel-heading-lamps-a")
      (variable-type  "DtRwReal")
      (display-label  "Seahawk Alpha's relative heading")
      (display-units  "degrees")
      (source-units  "degrees")
      (default-value  0.000000)
    )
    (real-parameter-data
      (parameter-name  "rel-heading-rhib-a")
      variable-type  "DtRwReal")
      (display-label  "Rigid-hull Alpha's relative heading")
      (display-units  "degrees")
```

```
    (source-units  "degrees")
    (default-value  90.000000)
  )
  ...
)
```

Figure 54 illustrates the Additional Systems properties dialog box created for the system shown in the previous paragraphs.

**Figure 54. Embedded Entity Properties dialog box**




## 6.2.2 Assigning Tasks and Plans to Embedded Entities

Once an embedded entity is deployed, it is just like any other entity in the scenario. You can give it independent tasks or write a plan for it. Follow the standard procedures for these activities.

### 6.2.3 Restoring an Entity that has Embedded Entities

If you deploy an embedded entity and it is destroyed, it is removed from the scenario and is not available to be deployed again. However, if you restore a parent entity and it had deployed entities that were subsequently destroyed, it is restored to the full complement of embedded entities. Entities that are deployed at the time the parent is restored are not affected. They are not duplicated in the parent's inventory and their relationship is not affected.

# 7. Generating Traffic (Pattern of Life)

This chapter explains how to generate pattern of life simulation object traffic and how to create and task pedestrian crowds.

# 7.1. Generating Traffic

You can make your scenario more realistic by including ground vehicles, pedestrians, surface watercraft, and aircraft that appear to be going about their own business. VR-Forces offers:

- A general-purpose spawn-and-sink traffic feature that can generate both vehicular traffic (including aircraft, watercraft, and ground vehicles) and pedestrian traffic (including humans and animals).
- One method for generating air traffic.
- One method for generating background watercraft.

Most of this chapter explains spawn-and-sink traffic generation. For a discussion of choosing a method of air traffic generation and instructions on using the air-traffic-only methods, see "7.3.1 Aircraft Clutter Area" on page 108. For information on creating a watercraft pattern of life based on historical data, see "7.3.2 Ship Clutter Area" on page 109.

# 7.2. Spawn and Sink Traffic Generation

The VR-Forces spawn and sink traffic feature (one of the pattern of life features) allows you to automatically generate background traffic for your scenario.

Automatically generated (spawned) simulation objects are created at spawn points. The default behavior for a spawned simulation object is to choose a destination, called a sink point, and move towards that point. If there are multiple sink points that it can move to, it randomly selects one. When the simulation object arrives at the sink point, the it is removed from the scenario. This creates the effect of purposeful activity in an area of the terrain without the need to create and task simulation objects individually.

As an alternative to the default behavior, you can write a plan that is assigned to each simulation object created at a particular type of spawn point. This plan can include any tasks and set data requests that are appropriate for the object type.

The behavior of a spawned simulation object, as well as other details such as how frequently they are spawned and how many spawned simulation objects can exist at one time, is configured in a spawn pattern. VR-Forces comes with two spawn patterns. You can edit spawn patterns and create as many new ones as you wish. Spawn points and sink points for new and edited spawn patterns are dynamically added to the Create menu and Create Object Palette so that it is easy to create new spawn and sink points for all of the available patterns.

The SubwayPatternOfLlife example scenario has spawn patterns that demonstrate continuous simulation object creation, burst traffic, and use of a custom plan.

You can find links to videos that demonstrate spawn patterns at https://www.mak.com/support/tutorials, or click the camera icon next to the procedure whose video you want to view.

> Spawn and sink points are automatically added to the Create menu. However, they are not flagged as Favorites on the Create Object Palette. If you flag a spawn or sink point as a Favorite, another entry is added to the Create menu of the specified Force.

## 7.2.1 Creating Spawn Points

You can create spawn points from the Create menu, the Create Object Palette, or from the Spawn Pattern Manager. Figure 55 illustrates a spawn point.

**Figure 55. Spawn point**


If the plan for a spawn point specifies Use Roads and a spawn point is placed too far away from the road network to use roads, a message is sent to the spawn point's console. If warning icons are enabled, a warning icon displays. For details about the Use Roads option, see "Creating a New Spawn Pattern" on page 102. For details about configuring console messages and warning icons, see "5.9. Configuring Object Console Messages" on page 79.

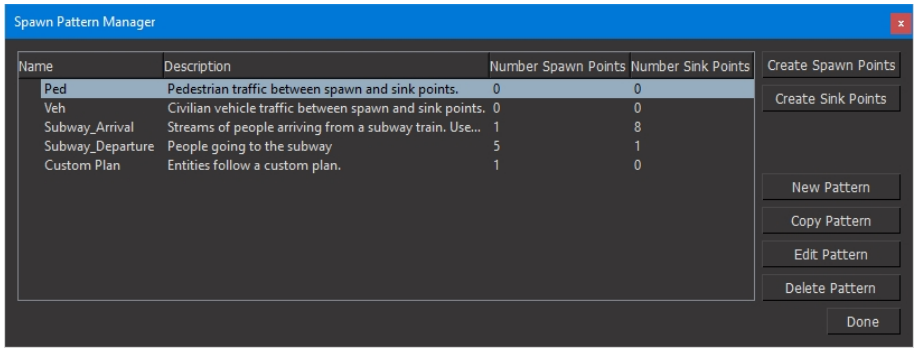
➤ **To create a spawn point from the Create menu:**

1. Choose **Create > Clutter >** *type*, where *type* is a spawn point for one of the spawn patterns. The cursor changes to the spawn point symbol.
2. Click on the terrain where you want to place the spawn point. You can continue clicking to create additional spawn points.
3. Right-click to exit create mode.

➤ **To create a spawn point from the Spawn Pattern Manager:**

1. Choose **Simulation > Spawn Pattern Manager**. The Spawn Pattern Manager opens (Figure 56).

**Figure 56. Spawn Pattern Manager**




2. Select the spawn pattern for which you want to create spawn points.
3. Click Create Spawn Points. The cursor changes to the spawn point symbol.

4. Click on the terrain where you want to place the spawn point. You can continue clicking to create additional spawn points.
5. Right-click to exit create mode.

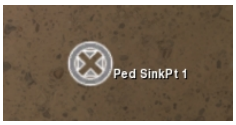
### Activating and Deactivating Spawn Points

You can activate and deactivate spawn points. To do so, use the Activation set data request. For details, see "Activation".

## 7.2.2 Creating Sink Points

If you are using the default spawned simulation object behavior – pick a sink point and move to it – then you have to create one or more sink points for the simulation objects to move to. Entities created at a given type of spawn point automatically choose among sink points of the same type. (You can configure spawned simulation objects to move to any kind of point by editing the spawn pattern. For details, see "Creating a New Spawn Pattern" on page 102.) You can create sink points from the Create menu, the Create Object Palette, or from the Spawn Pattern Manager. Figure 57 illustrates a sink point.

**Figure 57. Sink point**




If a sink point is more than 10 meters from a road and simulation objects move towards it with the Use Roads criteria, they move to the closest point on the road to the sink point and are then removed from the scenario.

➤ **To create a sink point from the Create menu:**

1. Choose **Create > Clutter >** *type*, where *type* is a sink point for one of the spawn patterns. The cursor changes to the sink point symbol.
2. Click on the terrain where you want to place the sink point. You can continue clicking to create additional sink points.
3. Right-click to exit create mode.

➤ **To create a sink point from the Spawn Pattern Manager:**

1. Choose **Simulation > Spawn Pattern Manager**. The Spawn Pattern Manager opens (Figure 56 on the previous page).
2. Select the spawn pattern for which you want to create sink points.
3. Click Create Sink Points. The cursor changes to the sink point symbol.
4. Click on the terrain where you want to place the sink point. You can continue clicking to create additional sink points.
5. Right-click to exit create mode.

## 7.2.3 Spawn Patterns

A spawn pattern configures these aspects of spawned simulation object creation and behavior:

- The name and description of the spawn pattern.
- The timing of simulation object creation. Entities can be spawned continuously or in bursts.
- The type of simulation object created.
- The spawned simulation object's plan. Entities can follow a default plan or a custom plan.

A spawn pattern's name identifies it on the Create menu and is used to build the label for individual spawn points, for example Ped SpawnPt 1, Ped SpawnPt 2, and so on.

A scenario can have multiple spawn patterns and each spawn pattern can spawn one or more object types. Spawning multiple simulation object types in one pattern is useful when you want heterogeneous traffic. For example, in a stream of pedestrians, you might want to include men, women, and children. In other cases, you might want a specific simulation object type engaging in specific behaviors and would only spawn one simulation object type in that pattern.

If you use the default plan, you can configure simulation object movement:

- Maximum and minimum speed. VR-Forces randomly assigns a speed between the maximum and minimum settings. It will not accept a speed that is greater than the maximum speed for the simulation object in the OPD file.
- Use Roads. Enables or disables road-following behavior for vehicles. Road following behavior is described in "2.8. Collision, Obstacle, and Feature Avoidance" on page 11 and "Entity Movement On Roads and Off Roads".

When you use the default plan, the Create Spawn Pattern dialog box lists the sink points that this pattern will use.

The SubwayPatternOfLife example scenario has spawn patterns that demonstrate spawning simulation objects continuously and in bursts. It also demonstrates use of a custom plan. The following sections describe these different ways to set up spawn patterns.

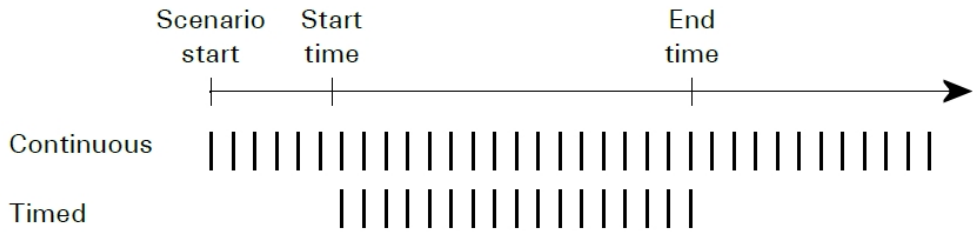
### How Spawn Events are Scheduled

Spawn patterns control the time period during which simulation objects are spawned and the rate at which they are spawned during that time period. A spawn pattern can use one of these options:

- Continuously. Entities are spawned as soon as the scenario starts and throughout its life.
- Time of Day. Entities are spawned between a starting time of day and ending time of day.
- Simulation Time. Entities are spawned between a specific starting and ending simulation time.

Figure 58 illustrates the difference between continuous and timed spawning.

**Figure 58. Continuous spawning compared to time-based**

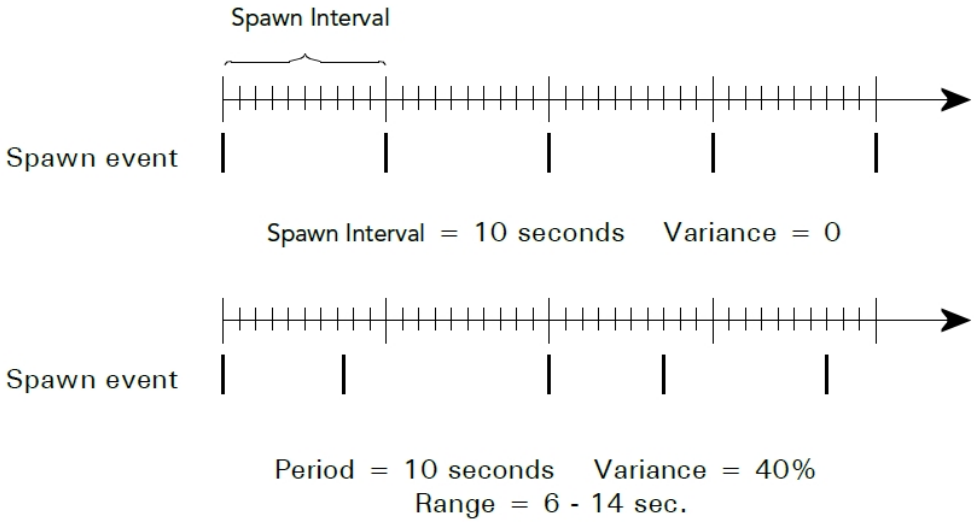



After you specify a time option, you must specify a spawn interval and a spawn interval variance. The spawn interval determines how frequently simulation objects are spawned. The spawn interval variance introduces some variability to the interval so that there is a more random and naturalistic flow of simulation object creation.

For example, if the spawn interval is 10 seconds, a simulation object is spawned every 10 seconds, plus or minus the variance. The variance is a percentage of the spawn interval. If the variance is 10%, then given a spawn interval of 10 seconds, an simulation object is spawned every 9 to 11 seconds. The spawn intervals are not fixed to an absolute time scale. A new spawn interval starts each time a spawn event occurs.

Figure 59 illustrates a series of spawn events with a spawn interval of 10 seconds and a spawn interval variance of 0% and a series of hypothetical events with a spawn interval of 10 seconds and a spawn interval variance of 40%. In the second case, each spawn event can take place anywhere between 6 and 14 seconds after the previous event. Notice that a new spawn interval starts at each spawn event.

**Figure 59. Spawn period and variance**




ⓘ The frequency with which a spawn point spawns simulation objects can be affected by the total number of simulation objects that it has spawned. If the maximum number of simultaneous simulation objects for a spawn point is reached, no more simulation objects are spawned from that point until simulation objects from that spawn point are removed from the scenario.

### Spawning Entities in Bursts

You can introduce additional variation to the spawn frequency by specifying that simulation objects be spawned in bursts. A burst pattern controls spawning behavior within a fixed period of time. For example, you could use a burst pattern to represent a cycle of train disembarkations or rush hour traffic.

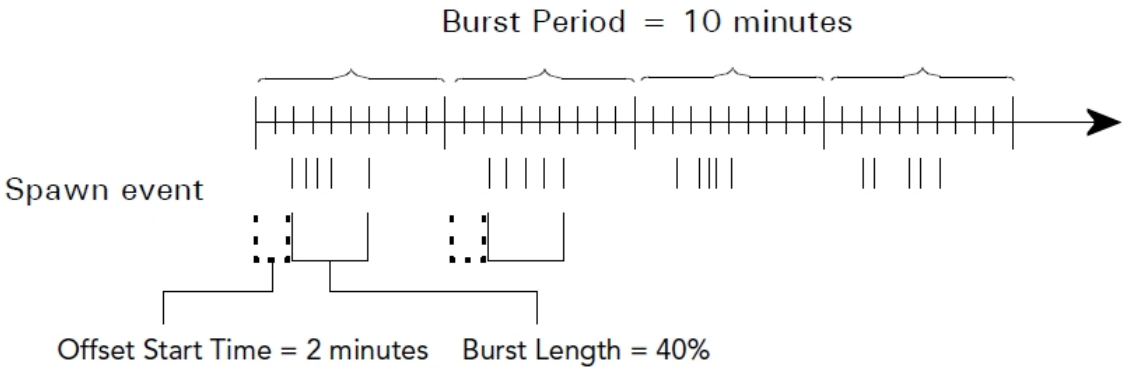
Burst patterns have these parameters:

- Burst Period. The period of time during which a burst can take place.
- Burst Length. The percentage of the burst period in which to spawn simulation objects.
- Offset Start Time. An offset from the start of the burst period at which to start spawning simulation objects.

Within a burst, simulation objects are created at the rate determined by the spawn interval and spawn interval variance values.

Figure 60 illustrates a series of bursts. The burst period is 10 minutes. The offset start time is two minutes, so each burst starts two minutes after the start of the burst period. The burst lasts for 40% of the length of the burst period, in this case four minutes. Within each burst, the frequency of spawn events is controlled by the spawn interval and variance values.

**Figure 60. Burst periods**




### Using a Custom Plan

As noted in "7.2. Spawn and Sink Traffic Generation" on page 96, the default behavior for a spawned simulation object is to choose a sink point, move to the sink point, and be removed from the scenario. However, you can introduce more complex behaviors for spawned simulation objects by writing a custom plan for a spawn pattern. The custom plan enables the simulation objects to engage in a complex set of behaviors, but you still have the benefit of automatic generation. An example of using a custom plan would be to generate a steady stream of jets taxiing to a runway and taking off from an airport.

Custom plans can include all of the tasks, sets, conditions, and global commands that a regular individual plan has. The plan is treated as an individual plan and applied to each simulation object as it is created. When you write a custom plan, simulation objects do not

move to sink points. They remain in the scenario unless their plan removes them. You can delete simulation objects without needing to specify the specific simulation object with the Delete Self global command.

To enable IFF (Identification, Friend or Foe) in aircraft generated by a spawn point, write a custom plan that includes a Set to turn on IFF. Note, however, that in spawn points IFF does not include unique IFF identification codes or realistic flight IDs.

#### Default Spawn Patterns and Scenario-Specific Patterns

The VR-Forces includes default spawn patterns for civilian vehicles and civilian lifeforms. When you create a scenario, these patterns are available in the scenario. When you save a scenario, the spawn patterns are saved with the scenario in the scenario extras file (*.xtr*). (This includes the default spawn patterns if you do not edit them, the edited versions if you have edited them, and any new spawn patterns you have created.) They are now scenario-specific. The next time you load the scenario, it uses the spawn patterns in the scenario extras file, not the default spawn patterns.

If you want to have additional default spawn patterns for your scenarios, you can add entries to *./appData/settings/vrfSim/defaultSpawnTemplates.mtl*. Copy one of the patterns in the file, then change the name and parameter values to suit your needs. Or, create a spawn pattern in a scenario and copy it from the *.xtr* file to *defaultSpawnTemplates.mtl*.

> ⓘ The civilian vehicle spawn pattern included with VR-Forces is set to user roads, therefore it requires a nearby road network to successfully spawn vehicles.

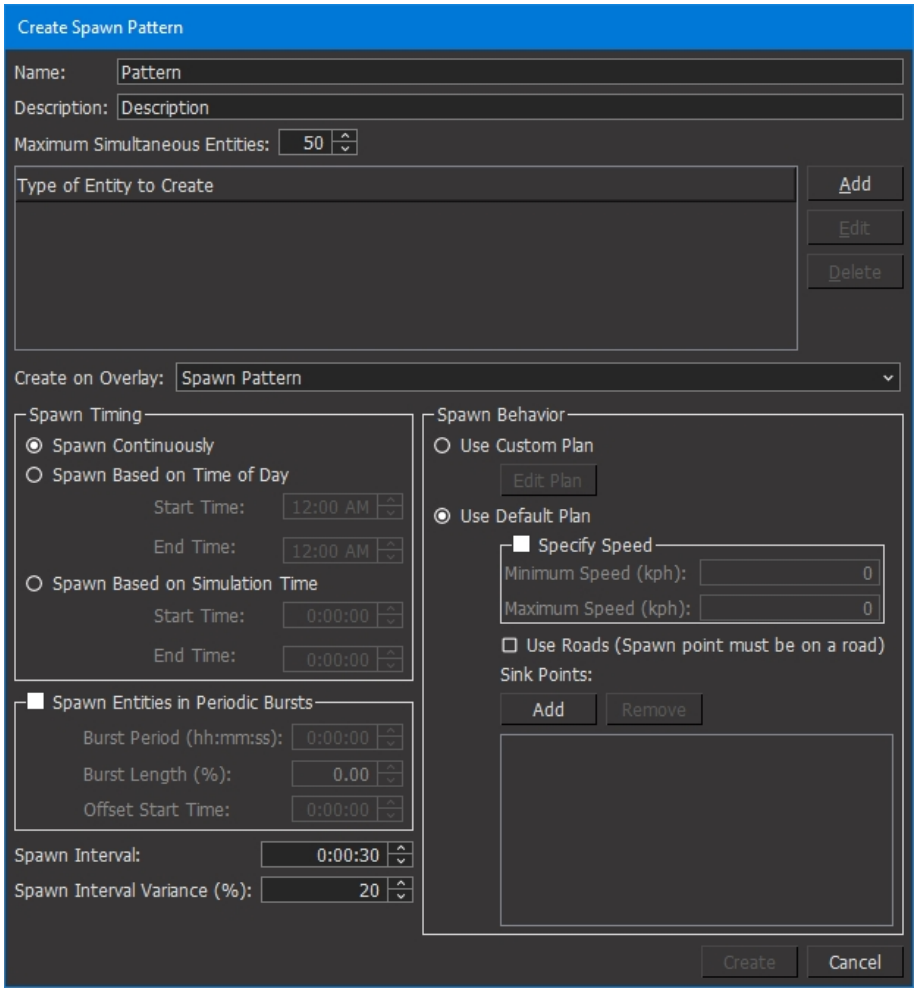
### 7.2.4 Spawn Pattern Management

#### Creating a New Spawn Pattern

➤ **To create a spawn pattern:**

1. Choose **Simulation > Spawn Pattern Manager**. The Spawn Pattern Manager opens (Figure 56 on page 97).
2. Click New Pattern. The Create Spawn Pattern dialog box opens (Figure 61).

**Figure 61. Create Spawn Pattern dialog box**




3. In the Name box, type a name for the spawn pattern. This name is used for the label for spawn points and sink points created from this pattern. It is a required field.
4. In the Description box, type a description for the pattern. This description is listed in the Spawn Pattern Manager to help identify the purpose of the pattern. This is a required field.
5. Specify the maximum number of simulation objects that can be spawned from a spawn point using this pattern and exist in the scenario at the same time. When the maximum is reached for a given spawn point, no more simulation objects are spawned from that point until a simulation object that was spawned from that point is removed from the scenario.

   For example, if the maximum number of simultaneous simulation objects for a spawn pattern is 200 and you have 10 spawn points of that type, you could have as many as 2000 spawned simulation objects. None of those spawn points will individually spawn more than 200 simulation objects at one time.

> ⚠ A high maximum number of simultaneous simulation objects along with a small spawn interval can quickly generate tens to hundreds of simulation objects. This can affect performance.

6. Specify the simulation object types to create:
    a. Click the Add button. The Object Type dialog box opens.
    b. Optionally, filter the object list. (For information about how to filter an objects list using tags, see "3.2.3 Filtering the Create Object Palette" on page 20.)
    c. Select the simulation object type that you want to add. The simulation object is added to the Type of Entity to Create list.
    d. Add additional simulation object types as desired.

> ⓘ Using the Object Type dialog box, you can specify all types of simulation objects, cultural features, and control objects, some of which do not necessarily make any sense to spawn. It is up to you to select types of simulation objects that make sense.

7. In the Create on Overlay list, select the overlay on which you want spawned simulation objects to be created.
8. Choose an option in the Spawn Timing group box:
    — Spawn Continuously. Entities are spawned throughout the simulation.
    — Spawn Based on Time of Day. Specify the starting time of day and ending time of day during which you want simulation objects to be spawned.
    — Spawn Based on Simulation Time. Specify the starting simulation time and ending simulation time during which you want simulation objects to be spawned.
9. Optionally, select Spawn Entities in Periodic Bursts. (For an explanation of periodic bursts, see "Spawning Entities in Bursts" on page 101.) If you select it, configure these values:
    — Burst Period. The time period for each burst.
    — Burst Length. The length of time, as a percentage of the burst period, during which simulation objects are spawned.
    — Offset Start Time. An offset from the start of the burst period to start spawning simulation objects.
10. Specify the spawn interval. This specifies the amount of time between each spawn event, modified by the spawn interval variance.
11. Specify the spawn interval variance as a percentage of the spawn interval.

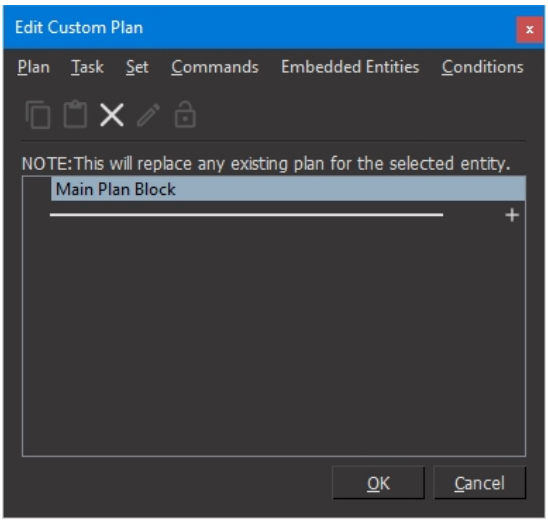
12. In the Spawn Behavior group box, select a behavior option:

    To use Custom Plan:

    a. Click Edit Plan. The Edit Custom Plan dialog box opens.

    b. Write a plan that each simulation object created from this spawn pattern will use. Unless you want the simulation objects to remain in the scenario permanently, end the plan with a Delete Self command. (For details about how to write plans, see "Writing Plans".)

**Figure 62. Edit Custom Plan dialog box**



To use a Default Plan:

a. Optionally, select the Specify Speed check box and specify the minimum and maximum speeds for spawned simulation objects. If you do not select this option, spawned simulation objects move at the default ordered speed in their OPD file. If you select it, they move at randomly chosen speeds between the minimum and maximum values. This option introduces some variance in behavior. Although the GUI allows you to set any maximum speed, simulation objects will not move faster than the maximum speed in the OPD file.

b. To have a simulation object move on roads, select the Use Roads check box. If this option is cleared, simulation objects move directly towards sink points without regard to vector road networks.

> ⚠ Use Roads is valid only for ground vehicles. Do not select this option for any other simulation object type.
>
> If Use Roads is selected, spawn points must be within 10 meters of the road. If a spawn point is farther away, the simulation objects are removed from the scenario as soon as they are created.
>
> If a sink point is more than 10 meters from a road, simulation objects that are moving towards that sink point move to the point on the road that is closest to it and are then removed from the scenario.

c. Optionally, use the Add button to edit the Sink Points list. A new spawn pattern will not have any sink points listed because until you create the spawn pattern, you cannot create any sink points for that pattern. Entities will automatically use sink points that match the spawn pattern, so after you create them, you do not have to add them to the list. However, they can also move to waypoints. If you want spawned simulation objects to move to waypoints, you must add them to the list.

13. Click Create.

### Copying a Spawn Pattern

If you want to create a new spawn pattern that will be similar to an existing spawn pattern, you can copy the existing spawn pattern to a new name.

➤ **To copy a spawn pattern:**

1. Choose **Simulation > Spawn Pattern Manager**. The Spawn Pattern Manager opens (Figure 56 on page 97).
2. Select the spawn pattern that you want to copy.
3. Click Copy Pattern. The Create Spawn Pattern dialog box opens. It has a default name based on the original pattern name.
4. Edit the spawn pattern. The meaning of each option is described in "Creating a New Spawn Pattern" on page 102.
5. Click Create.

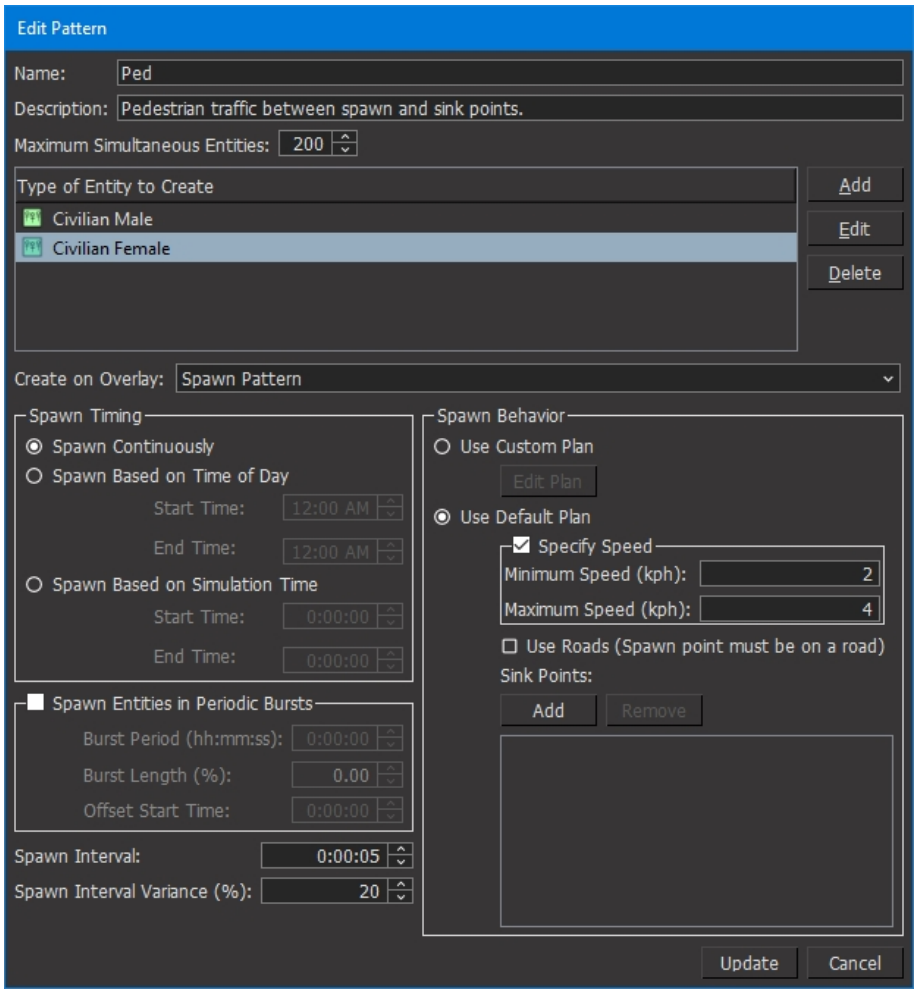
### Editing a Spawn Pattern

You can change any of the parameters of a spawn pattern. The changes do not affect any simulation object that has already been spawned.

➤ **To edit a spawn pattern:**

1. Choose **Simulation > Spawn Pattern Manager**. The Spawn Pattern Manager opens (Figure 56 on page 97).
2. Select the pattern that you want to edit.
3. Click Edit Pattern. The Edit Spawn Pattern dialog box opens (Figure 63).

**Figure 63. Edit Spawn Pattern dialog box**




4. Change any value as desired. The meaning of each option is described in "Creating a New Spawn Pattern" on page 102.
5. Click Update.

### Deleting a Spawn Pattern

If you delete a spawn pattern, all spawn points of that type are deleted from the scenario. Sink points are not deleted. The spawn and sink points of this type are removed from the Create menu and Create Object Palette. If the scenario is running, any simulation objects

that were spawned from these points continue executing their plans.

➤ **To delete a spawn pattern:**

1. Choose **Simulation > Spawn Pattern Manager**. The Spawn Pattern Manager opens (Figure 56 on page 97).
2. Select the spawn pattern that you want to delete.
3. Click Delete Pattern. You are prompted to confirm the deletion.
4. Click Yes.

## 7.3. Clutter Areas

Clutter areas allow you to use automatic identification system (AIS) or automatic dependent surveillance-broadcast (ADS-B) data included with VR-Forces to create water or air traffic as "clutter objects". This provides you with realistic background clutter, which is useful when you are simulating the real world and you need realistic "noise" on the water or in the air. The clutter objects are all of the same force as the area.

⚠
- The prerecorded AIS and ADS-B data were acquired and then preprocessed by MAK. You cannot use your own data.
- There are forty-eight hours' worth of data in the files. If your scenario runs longer than two days, you might experience problems with the clutter objects. If you require more than two days of simulation time, please contact your MAK salesperson or info@mak.com.

Clutter objects are intended to be background entities and display in the Clutter Objects overlay by default. They are not "full-fledged" entities: they are simpler and therefore have a lower impact on performance. When generating clutter objects from AIS or ADS-B data, VR-Forces uses the timestamp in the file to place the clutter objects where they were at the same time of day as the scenario.

Objects move according to the data in the file. If they move out of the clutter area, the object is removed from the scenario. Likewise, if there is data of a ship or fixed-wing aircraft moving into the area, it is added to the scenario.

You cannot assign tasks to clutter objects; they simply move according to the information in the AIS or ADS-B data. If you want to task a clutter object to do something other than what is in the data, use the Promote From Clutter set data request to promote them to a full-fledged entity. After promoting them, you can task the entity as desired. See "Promote From Clutter".

ⓘ You can customize the clutter entities. See "Clutter Entities" in the *Adding Content to MAK ONE Applications Reference Manual*.

### 7.3.1 Aircraft Clutter Area

When you create an Aircraft Clutter Area, VR-Forces populates it with fixed-wing aircraft objects based on two days' worth of ADS-B data provided by MAK. VR-Forces uses the data to name the clutter aircraft for the fixed-wings they represent, select a model that

most closely matches the type of aircraft, and determine their movement. The aircraft clutter area only creates background fixed-wing aircraft. It does not create rotary wing or other types of aircraft.

The aircraft clutter area does not simulate takeoffs and landings. It only displays fixed-wing aircraft objects that are in the air. When the data has an aircraft take off, the object is added to the scenario. Likewise, when an aircraft lands, the object is removed from the scenario.

> Each entity's IFF data displays in the IFF page of the Information dialog box and is published to the network.

➤ **To create aircraft using a clutter area:**

1. Open the Create Object Palette.
2. Optionally, click the Clutter filter ().
3. Select Aircraft Clutter Area.
4. Draw an area on the terrain as you would any other areal tactical graphic. When the area is completed, VR-Forces creates objects to populate the area.

### 7.3.2 Ship Clutter Area

When you create a Ship Clutter Area, VR-Forces populates it with watercraft objects based on two days' worth of AIS data provided by MAK. VR-Forces uses the data to name the clutter ships for the watercraft they represent, select a model that most closely matches the type of vessel, and determine their movement.

> Each entity's AIS data displays in the AIS page of the Information dialog box.

➤ **To create ships using a clutter area:**

1. Open the Create Object Palette.
2. Optionally, click the Clutter filter ().
3. Select Ship Clutter Area.
4. Draw an area on the terrain as you would any other areal tactical graphic. When the area is completed, VR-Forces creates objects to populate the area.

# 8. Crowds and Multiple Simulation Objects

This chapter explains how to create many simulation objects at one time, how to create and task pedestrian crowds, and how to create animal herds.

# 8.1. Creating Crowds, Herds, and Multiple Simulation Objects

VR-Forces can quickly create multiple simulation objects within an area. This allows you to create crowds, traffic, or other multi-object groups without having to place each simulation object by hand. You can use any of the following methods to create multiple objects:

- Crowd unit. Create a crowd unit from the Create Object Palette or aggregate individual humans into a crowd unit. Assign crowd-specific tasks to the crowd.
- Herd units. Create a herd unit of cows, goats, or sheep from the Create Object Palette. They graze and wander around the central point where you placed the unit.
- Pedestrian area. A pedestrian area is populated by a random number of male and female civilians. They execute a wander task.
- Herd areas. An area is populated by a random number of cows, goats, or sheep. The herd animals graze and wander around the area.
- Multiple simulation objects. You can specify the number of simulation objects to create, the type of objects to create, and you can assign tasks and set data requests to them.

> Lifeforms require a navigation area to perform movements such as the Wander task. Two-legged lifeforms (humans and chickens) require a lifeform nav. area. Four-legged lifeforms (cows, sheep, goats, and so on) require a ground platform nav. area. See "Generating Navigation Data" to learn more.

# 8.2.  Crowd and Herd Units

You can create a crowd or herd by placing a unit of the type you need. VR-Forces then creates entities of the particular type.

Based on the entity file, the entities may have random appearances. You cannot control the entities' appearances when you create the unit, but you can set the available appearances for the entity in the Simulation Object Editor. For more information, see "Editing a Lifeform's Visual Model and Animation".

## 8.2.1 Creating Crowd Units

VR-Forces has six preconfigured crowd units: small, medium, and large crowds of western or middle eastern civilians. They create a unit of civilian males and females. You can also create crowd units by aggregating individual human entities. However, some of the animations used in crowd tasks are specific to the civilian characters used in the supplied units.

When you place a crowd, if the terrain has pedestrian paths or other pedestrian-specific locations, the members of the crowd are placed in these locations.

Crowd members are created within a configurable radius of the location you specify, with a configurable density. By default, crowds are created using the parameters in Table 6. You can change the values in the Simulation Object Editor.

**Table 6. Crowd units**

| Unit | Radius (m) | Density | Approximate Size* |
|---|---|---|---|
| Civilian Crowd (Small) | 15 | 2 | 50 |
| Civilian Crowd (Medium) | 30 | 2 | 100 |
| Civilian Crowd (Large) | 50 | 2 | 175 |
| Civilian Mideast Crowd (Small) | 15 | 2 | 50 |
| Civilian Mideast Crowd (Medium) | 30 | 2 | 100 |
| Civilian Mideast Crowd (Large) | 50 | 2 | 175 |
| *If there are not enough valid locations to place entities, the size of the unit may be smaller than the theoretical maximum. | | | |

Crowd units are supported only in entity-level scenarios.

> ⚠ You can create preconfigured crowds only in navigation areas and crowd tasks can be executed only in navigation areas. Although you can aggregate individual entities into a crowd outside of a navigation area, they will not be able to execute any crowd tasks.

➤ **To create a crowd:**

1. Open the Create Object Palette.
2. Click the Entity filter (  ).
3. Clear any tag filters (click the X button on the Filters line).
4. Optionally, click the Human top-level tag, then the Civilian tag, and then click the Crowd tag.
5. Select the crowd unit you want.
6. Click on the terrain to place the unit.

## 8.2.2 Creating Herd Units

VR-Forces has three preconfigured herd units: cows, goats, and sheep. Each unit generates thirty-six animals that graze and wander within an approximate 20-meter radius around the point where you placed the unit. You can also create herd units by aggregating individual animals, but they will not wander automatically.

➤ **To create a herd:**

1. Open the Create Object Palette.
2. Click the Entity filter (  ).
3. Clear any tag filters (click the X button on the Filters line).

4. Optionally, click the Animal top-level tag and then click the Crowd tag.
5. Select the herd unit you want.

# 8.3.  Crowd and Herd Areas

You can draw areas to create crowds of people or herds of animals. VR-Forces then creates entities of the particular type.

Based on the entity file, the entities may have random appearances. You cannot control the entities' appearances when you create the area, but you can set the available appearances for the entity in the Simulation Object Editor. For more information, see "Editing a Lifeform's Visual Model and Animation".

## 8.3.1 Creating Crowds Using a Pedestrian Area

When you create a pedestrian area, VR-Forces populates it with a random number of male and female civilian entities. They are created as a ground unit. They are all assigned a wander task. You cannot specify the number of entities to create or the mix of entities to create when you create the area.

> ⚠ If you delete a pedestrian area, all of the entities that were created for it are also deleted.
>
> You can create pedestrian areas only in navigation areas.

➤ **To create a crowd using a pedestrian area:**

1. Open the Create Object Palette.
2. Optionally, click the Clutter filter (  ).
3. Select Pedestrian Area.
4. Draw an area on the terrain as you would any other areal tactical graphic. When the area is completed, VR-Forces creates entities to populate the area.

## 8.3.2 Creating Animal Herds Using Areas

You can create herds of cows, goats, or sheep using the appropriate area type. Once you create the area, VR-Forces populates it with a random number of that type of animal and they will graze and wander within the area. You cannot specify the number of animals to create at first.

➤ **To create a herd using an area.**

1. Open the Create Object Palette.
2. Optionally, click the Clutter filter (  ).
3. Select the area for the type of herd you want to create.
4. Draw an area on the terrain as you would any other areal tactical graphic. When the area is completed, VR-Forces creates animals of the selected type to populate the area.

### 8.3.3 Adding Entities to a Pedestrian or Herd Area

After you create a pedestrian or herd area, you can add additional entities to the area. See "Create More Entities".

## 8.4. Crowd Tasks

Crowds can execute some of the tasks available to other ground units. Crowds created as a unit or in an area can perform these tasks. They have the following movement tasks:

- Circle around a location
- Crowd along a line
- Crowd around a location
- Crowd around an object
- Crowd around waypoints
- Gather in front of an entity
- Wander
- Flee from entities
- Protest along a line
- Protest around a location
- Riot
- Disperse crowd

When a crowd gathers around a location or object, VR-Forces calculates a target destination for each member of the crowd and each member of the crowd tries to move to that destination. If you send two or more crowds to the same location or object, these target destinations may overlap. Individual entities may give up on their movement tasks or may end up very close to members of the other crowd.

For details about each task, see "Human and Crowd Tasks".

## 8.5. Creating Multiple Simulation Objects

You can add multiple simulation objects to a scenario in one operation and assign them a common task. The simulation objects are added to the scenario as members of a selection group. (For information about selection groups, see "4.3. Using Selection Groups" on page 53.) Creation of multiple simulation objects has constraints:

- The simulation objects are added within a specified area. If an appropriate area does not already exist, you can create one as part of the simulation object creation process.
- If the area in which you want to place simulation objects overlaps navigation data, the simulation objects are placed only in the portion of the area that has navigation data, as illustrated in Figure 64.

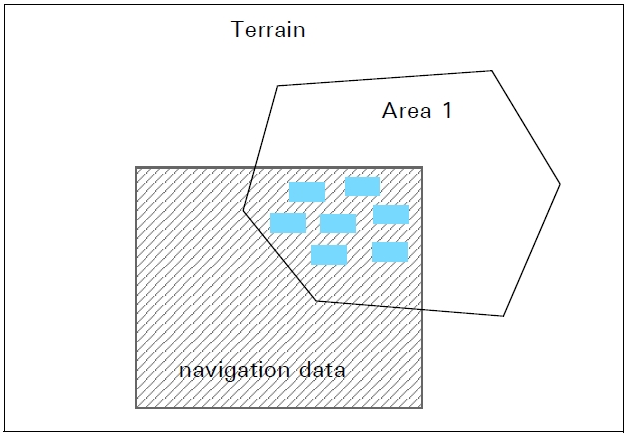
- If the area you choose for placing simulation objects is too small to contain the number of simulation objects that you specify, some simulation objects might not be created.

When you create multiple simulation objects, VR-Forces automatically puts them in a selection group named group_*number*, where *number* starts at 0 and is incremented for each group created.

In Figure 64, Area 1 overlaps the navigation data. If, for example, you try to place 25 entities in this area, they are placed only in the portion of Area 1 that overlaps navigation data, and only the number of simulation objects that fit in this subset of Area 1 are created.

- If VR-Forces cannot create the number of simulation objects specified (due to one of the reasons mentioned in this section), there is no notification. It is your responsibility to check to see how many simulation objects were created.
- Creating multiple simulation objects as described in this section is not the same thing as pattern of life traffic generation or creating simulation object groups. It is just a way to create a lot of simulation objects at one time and give them a task. And unlike creating crowds, which only contain civilians, you can create any type of simulation object.

**Figure 64. Placement of multiple simulation objects**




When you create multiple simulation objects, you can ask VR-Forces to run an accessibility check before it creates each simulation object. VR-Forces tries to ensure that the simulation object being created can reach a specified waypoint from a candidate location in the specified area. This check is primarily designed to avoid putting simulation objects on top of buildings that are unreachable and would not allow the simulation object to leave the building.

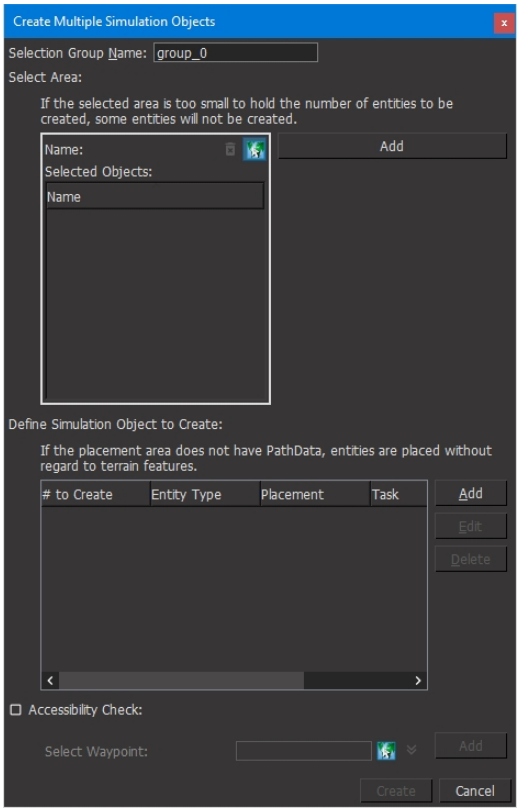
- If the terrain does not have navigation data, the accessibility check is disregarded and the simulation objects are created.

- If the terrain has navigation data and if the candidate location cannot use navigation data to reach the waypoint, VR-Forces does not create the simulation object at that location. It tries to find another location at which to place the simulation object. In a case where, after multiple attempts to place a simulation object, the accessibility check still fails:
  - If this is the first simulation object that VR-Forces has tried to create, VR-Forces disregards the accessibility check and creates the simulation objects (within the constraints previously mentioned).
  - If the accessibility check fails for any simulation object after the first simulation object was created, VR-Forces assumes there is no more space in which to create simulation objects and stops trying to create them.

➤ **To add multiple simulation objects to a scenario:**

1. Choose **Create > Multiple Simulation Objects**. The Create Multiple Simulation Objects dialog box opens (Figure 65).

**Figure 65. Create Multiple Simulation Objects dialog box**




2. Optionally, change the name of the selection group.
3. In the Select Area group box, type the name of the area in which you want to create the simulation objects, or select the area from the list or on the map. If the area in which you want to create the simulation objects does not exist, add a new area as described in "8.5.1 Adding an Area for Multiple Simulation Object Creation" on the next page.

4. In the Define Entities To Create group box, add the simulation objects that you want to create. For details about this part of the procedure, see "8.5.2 Specifying the Simulation Objects to Create" below.
5. Optionally, edit or remove any of the entity sets listed in the Define Entities To Create list.
6. Optionally, select the Accessibility Check check box. If you select this check box, select a waypoint. You can create a waypoint from this dialog box if necessary.
7. Click Create. The entities are created in the specified area.

## 8.5.1 Adding an Area for Multiple Simulation Object Creation

➤ **To add an area from the Create Multiple Entities dialog box:**

1. In the Select Area group box, click Add. A Create Area tab is added to the window.
2. Create an area following the standard VR-Forces procedure for creating areas. For details, see "3.3. Placing Objects" on page 26.

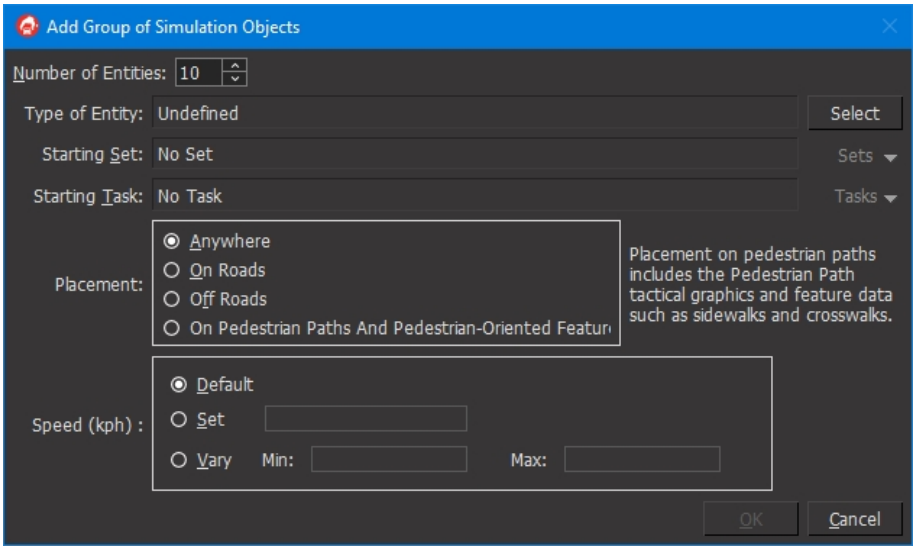
## 8.5.2 Specifying the Simulation Objects to Create

When you add multiple simulation objects to a scenario, you can specify any of the supported object types and assign a common task or set data request to a specific set of simulation objects.

➤ **To specify the simulation objects to add in the Create Multiple Entities dialog box:**

1. In the Define Entities to Create group box, click Add. The Add Group of Entities dialog box opens (Figure 66).

**Figure 66. Add Group of Entities dialog box**




2. In the Number of Entities box, specify the number of simulation objects you want to add for this set.

3. Click the Select button that is next to the Type of Entity text box. The Entity Type dialog box opens. The list of object types available is the same list of simulation objects provided on the Create Object Palette.
4. Select the object type you want to add. The dialog box closes and the object type is listed in the Type of Entity box.
5. Optionally, specify a starting set data request by clicking the Sets button and selecting a set data request. The list is not context sensitive. You must ensure that the selected set data request is appropriate to the object type that you are adding.
6. Optionally, specify a starting task by clicking the Tasks button and selecting a task. The list is not context sensitive. You must ensure that the selected task is appropriate to the object type that you are adding.
7. Optionally, select an option in the Placement group box. VR-Forces will try to place the simulation objects in the desired location.
8. Optionally, set a speed and variance range.
9. Click OK. The object set is added to the list in the Define Entities To Create group box (Figure 65 on page 117).
10. Optionally, add additional object sets.

## 8.6. Pedestrian-Oriented Tactical Graphics

VR-Forces has tactical graphics that are particular to pedestrians. These objects only work in navigation areas.

- Pedestrian path. When humans plan paths in navigation areas that have navigation data, if a pedestrian path is present that can take them to their destination, they will use it. You can think of a pedestrian path as a route that a human automatically chooses to follow instead of being told to follow it. Pedestrian paths are supported only for entity-level scenarios.
- Civilian visit points. Waypoints that attract the attention of civilians.
- Crosswalks. Crosswalks are similar to pedestrian paths. Pedestrians prefer them when path planning. The unique aspect of crosswalks is that if an entity is configured with a human crosswalk controller, it checks to see if it is safe to cross the street before proceeding. (Crosswalks are an experimental feature. Please contact support (https://www.mak.com/mak-one/support/portal) if they do not work as you expect.)
- Pedestrian areas. Pedestrian areas automatically generate pedestrians when you create them. For details, see "8.3.1 Creating Crowds Using a Pedestrian Area" on page 114 .

### 8.6.1 Using Civilian Visit Points

Civilian visit points are waypoints that attract the attention of civilian entities that are walking nearby. The civilians can be members of a crowd or individual entities. When a civilian comes within a configurable distance of a civilian visit point, it stops and faces the point a configurable percentage of the time. If it stops, it waits a few seconds and then continues its task.

This behavior is controlled by the Visit Interest Points reactive task. You can edit the parameters that control an entity's behavior in the Manage Reactive Tasks dialog box. For more information about reactive tasks, see "Reactive Tasks".

# 9. Creating and Using an Order of Battle

An Order of Battle is a description of the simulation objects and simulation object hierarchy that you can create in a scenario.

# 9.1. Introduction to the Order of Battle

In the abstract, an Order of Battle (OOB) describes the hierarchical structure of the personnel and resources that are available to a military commander. The term can also be used to refer to the hierarchical structure of the personnel and resources deployed in a particular engagement.

VR-Forces implicitly supports the second sense of the term by listing all simulation objects in a scenario in the Objects List Panel. It also supports the abstract sense of the term by allowing you to construct an Order of Battle that represents a hierarchy of simulation objects that can be instantiated (created) in a scenario.

You create an Order of Battle in the context of a scenario. However, once you create an OOB, you can export it and then import it into other scenarios. This lets you quickly create new scenarios that use the same OOB for training or scenario development. You can configure VR-Forces so that users do not have access to the Create Object Palette and can only add simulation objects to a scenario if they are in the OOB.

An Order of Battle can include any simulation object that you can create from the Create Object Palette. However, unlike the Create Object Palette where each object is essentially a template for creating multiple instances of that object type, each member of an Order of Battle has its own unique ID (UUID). When you instantiate a member of the OOB, you only create that one instance of the object type.

In VR-Forces, you can:

- Create an Order of Battle, which involves:
  - Adding members.
  - Removing members.
  - Changing the name or object type of members.
  - Changing the order of members.
  - Organizing members into units.
- Instantiate members of the OOB in a scenario.
- Import and export OOBs.

When you create an OOB, it is saved with the scenario.

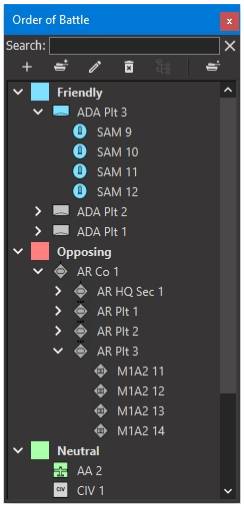
## 9.1.1 The Order of Battle Panel

You can view and manage the Order of Battle on the Order of Battle Panel (Figure 67). The Order of Battle Panel lists the members of the OOB by force.

Each object in the OOB list indicates its status in the scenario:

- Gray icon. A member of the OOB that is not instantiated in the scenario.
- Colored icon. A member of the OOB that is instantiated in the scenario.

**Figure 67. Order of Battle Panel**




➤ To open the Order of Battle Panel, choose **View > Order of Battle Panel**.

## 9.2. Creating an Order of Battle

You create an Order of Battle by adding members. You can add any simulation object that is available on the Create Object Palette.

You can add objects to the OOB by either:

- Adding objects to the Order of Battle without adding them to the scenario.
- Adding objects that exist in the scenario, but are not currently part of the OOB, to the OOB.

### 9.2.1 Adding Objects Directly to the Order of Battle

You can add a simulation object to the Order of Battle, without creating it in the scenario.

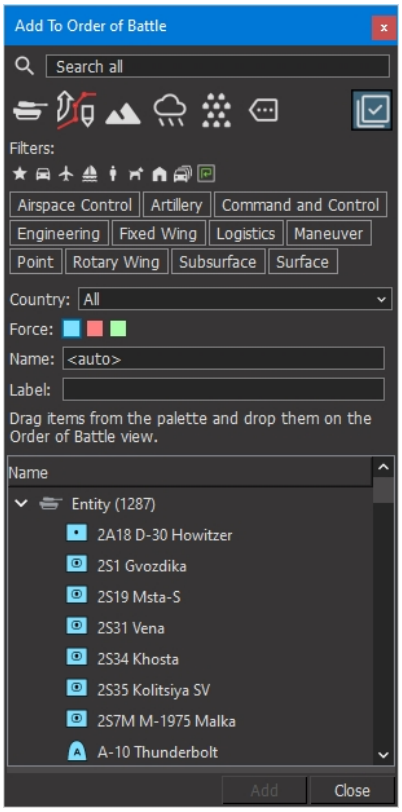
When using this method, if you want to have more than one instance of an object in the OOB, you must add a discrete member for each one. For example, if you want to have 10 HMMWVs in the OOB, you must add 10 HMMWVs, one at a time.

If you want to add many instances of the same simulation object to the Order of Battle, it may be quicker to create them in the scenario from the Create Object Palette and then add them to the OOB.

➤ **To add simulation objects directly to an Order of Battle:**

1. Open the Order of Battle Panel.
2. Click the Add button (+). The Add to Order of Battle dialog box opens (Figure 68). It is similar to the Create Object Palette, except that you can also specify a name and label for the object.

**Figure 68. Add to Order of Battle dialog box**




3. On the Set Force line, select the force to which you want to add this object.
4. Optionally, filter the list.
5. In the list of simulation objects, select the object you want to add to the OOB.
6. Optionally, specify a name for the object.
7. Optionally, add a label for the object.
8. In the Order of Battle Panel, select the force or unit under which you want to add the object. The following rules apply:
   - If you add to a force, it becomes a top level subordinate of the force.
   - If you add to a unit, it becomes a subordinate of that unit.
   - If you add it to an entity, it becomes a sibling of the entity. If the entity is part of a unit, the new member becomes part of the unit.
9. In the Add to Order of Battle dialog box, click Add.

ⓘ You can quickly add objects to the OOB by dragging them from the Add to Order of Battle dialog box to the Order of Battle Panel. The same rules for adding members apply. Also, regardless of the force selected on the Set Force line, when you drag an object onto the Order of Battle Panel, it becomes part of the force you drag it to.

### 9.2.2 Adding Existing Objects to the Order of Battle

You can create objects from the Create Object Palette and then add them to the OOB. You can also use this method to add objects from existing scenarios to an OOB. You can add objects individually and you can add all objects at once.

When you add an existing object to the OOB, it is added to the same force and its place in the force hierarchy is preserved. If you add a member of a unit, the entire unit is added to the OOB.

➤ **To add one or more existing objects to the OOB:**

1. Select the objects that you want to add to the OOB.
2. Choose **Simulation > Order of Battle > Add to Order of Battle**.

➤ To add all objects in the scenario to the OOB, choose **Simulation > Order of Battle > Add All Simulation Objects to Order of Battle**.

## 9.3. Creating Units in the Order of Battle Panel

You can aggregate objects listed in the Order of Battle Panel into units. The process and criteria for creating units is the same as described in "Creating a Unit by Combining Existing Simulation Objects".

To aggregate members of an Order of Battle into units:

1. Select the objects that you want to aggregate.
2. Choose **Objects > Unit > Aggregate As**, or click the Aggregate As button (  ) on the Order of Battle Panel. The Aggregate As dialog box opens.
3. Select the unit type you want from the list.
4. Optionally, change the order of the unit members.
5. Click OK.

## 9.4. Instantiating Members of an Order of Battle in a Scenario

If an Order of Battle includes simulation objects that have not been instantiated (created) in the scenario, you can add them to a scenario. Since each member of the OOB represents a specific simulation object with its own unique ID (UUID), when you add a simulation object from the OOB, you only add that specific object. You cannot add multiple instances of the object type as you would from the Create Object Palette.

➤ **To add a simulation object to a scenario from the Order of Battle Panel:**

1. In the Order of Battle Panel, select the simulation object you want to add to the scenario. (You can only add objects that are not already in the scenario. That is, objects with gray icons.)
2. Click the Add Object button (  ).
3. Click on the terrain where you want to place the object. The icon for the object in the Order of Battle Panel changes from gray to its force color.

### 9.4.1 The Order of Battle Palette

There may be cases where you want to set up a training exercise in which participants can only add simulation objects from the Order of Battle. VR-Forces has an Order of Battle Palette that contains the Order of Battle as listed in the Order of Battle Panel, but without any of the controls for adding, removing, or editing the OOB. Users can add simulation objects in the same way they would from the Create Object Palette. The Order of Battle Palette is hidden by default, but you can enable it from the View menu.

Using the window layout feature, you can save a layout in which the Create Object Palette is hidden and the Order of Battle Palette is visible. Taking things a step further, you could edit the GUI configuration file so that the Create Object Palette and the Order of Battle Panel are not even available in the GUI. For information about saving window layouts, see "Saving Window Layouts". For information about configuring the menu options in the GUI, see "Configuring Toolbars, Dialog Boxes, and Menus".

## 9.5. Removing Objects from an Order of Battle

You can remove objects from the OOB. When you remove an object from the OOB, if it is instantiated in the scenario, it stays in the scenario. If you delete an object from a scenario that is a member of the OOB, it is not removed from the OOB. The object's entry in the Order of Battle Panel changes from its force color to gray.

If you remove a unit, all of its subordinates are removed. If you remove a member of a unit, the rest of the unit is not affected.

➤ **To remove an object from the Order of Battle:**

1. Select the object in the Order of Battle Panel or on the map.
2. Click the Delete button (🗑) on the Order of Battle Panel or press the **Delete** key.

## 9.6. Editing the Objects in an Order of Battle

You can edit the members of an Order of Battle:

- If an object is created in the scenario, editing it opens up the standard Edit dialog box for a simulation object. For details, see "3.10. Editing Simulation Objects" on page 36.
- If an object is not created in the scenario, you can change its name, object type, and force.

➤ **To edit a member of the Order of Battle:**

1. Select an object in the Order of Battle Panel.
2. Click the Edit Object button (✎).

   If the object has been created, a standard Edit dialog box opens.

   a. Make changes as desired.

   b. Click Update.

If the object has not been created, the Edit Name / Object Type dialog box opens.

a. To change the name, type a different name in the Name box.

b. To change the label, type a different name in the Label box.

c. To change the object type, select a different object in the list.

d. Click OK.

## 9.7. Changing the Order of Objects in an Order of Battle

You can reorganize the members of the OOB. This allows you to populate the OOB in the most convenient manner and then organize it into the desired hierarchy. You can move members from units into other units, from units to the force level, and from the force level to units. You cannot change the order in which sibling objects or units are listed.

➤ To reorganize an object in the Order of Battle, in the Order of Battle Panel, drag the object to its new location.

## 9.8. Exporting an Order of Battle

You can export an Order of Battle so that you can import it into other scenarios.

➤ **To export an Order of Battle:**

1. Choose **Simulation > Order of Battle > Export Order of Battle**. The Export Order of Battle dialog box opens.
2. Type a name for the Order of Battle.
3. Click Save.

## 9.9. Importing an Order of Battle

You can import an Order of Battle into a scenario. When you import an OOB, you can merge it with the existing OOB, if any, or you can selectively import members of the OOB.

➤ **To import an Order of Battle:**

1. Choose **Simulation > Order of Battle > Import Order of Battle**. The Import Order of Battle dialog box opens.
2. Select the OOB you want to import.
3. Click Open. The Order of Battle Selector dialog box opens.
4. To import all members of the selected OOB, click Import All.

   To Import selected members of the OOB, drag them from the Order of Battle Selector to the Order of Battle Panel.
5. If necessary, close the Order of Battle Selector.