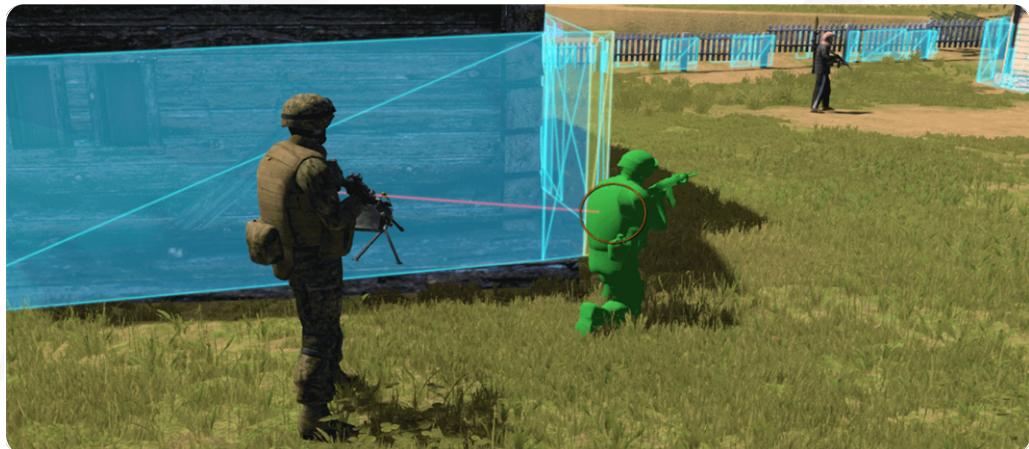


# VBS Control AI



VBS4 24.1.1



## Documentation Legal Notice

This Documentation, including any embedded help systems and electronically distributed materials, (hereinafter referred to as the "Documentation") is for your informational purposes only and is subject to change or withdrawal by Bohemia Interactive Simulations (BISim) at any time. This Documentation and its contents are proprietary information of BISim, also protected by copyright, and may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of BISim.

If you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all BISim copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to BISim that all copies and partial copies of the Documentation have been returned to BISim or destroyed.

BISim has made every reasonable effort to ensure the accuracy of all the information contained in the Documentation. However, product specifications are subject to change without notice, and BISim makes no representations or warranties regarding the accuracy, completeness, or suitability of information contained in the Documentation. To the maximum extent permitted by law, BISim disclaims any and all liability for any loss, damage (direct or indirect) or other consequence which may arise from the use of or reliance upon any information contained in the Documentation.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

Copyright © 2024 - Bohemia Interactive Simulations. All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

## Customer Support

The Bohemia Interactive Simulations Support page can be found at:

- <http://www.bisimulations.com/support>

For any type of assistance with Bohemia Interactive Simulations products, use the following support email and we will respond to your query with urgency.

- [support@bisimulations.com](mailto:support@bisimulations.com)

Our website contains a range of media and handouts relating to Bohemia Interactive Simulations products:

- <http://www.bisimulations.com/>

The BISim Wiki is the primary resource on VBS4 scripting:

<https://sqf.bisimulations.com/display/SQF/VBS+Scripting+Reference>

## PhysX

VBS4 uses the PhysX physics engine. For more information on PhysX visit the Nvidia site.

<https://gameworksdocs.nvidia.com/simulation.html>



# Contents

<b>VBS Control AI</b>	1
<b>1. VBS Control AI Overview</b>	10
1.1 Disabled and Incompatible Features	11
<b>2. Control AI UI Overview</b>	17
2.1 Control AI Editor Objects	17
2.2 (F3) Waypoints Editor Object	18
2.3 Custom Behaviors	20
2.4 Control AI - Civilian Editor Object	20
2.5 Control AI - Activity Editor Object	20
2.6 Control AI - Area Editor Object	21
2.7 Control AI Menu	21
<b>3. Waypoints</b>	23
3.1 Linking Existing Waypoints to Other Editor Objects	25
3.2 Branching Waypoints	25
3.3 Waypoint Synchronization	26
3.4 Waypoint Considerations	27
3.5 Waypoint Functions and Parameters	28
<b>4. AI on Rails</b>	36
4.1 Individual Move	37
4.2 Individual Fire At	39
4.3 Delayed Order Execution	42
4.4 Branching Orders	44
4.5 AI on Rails Orders	46
4.5.1 Individual - Fire At Order	47
4.5.2 Individual - Move Order	50
<b>5. Military AI</b>	53
5.1 Infantry Tutorial	54
5.1.1 Creating the Infantry Groups	54
5.1.2 Creating an Advance Order	54

5.1.3 Creating an Assault Order .....	55
5.1.4 Creating a Suppress Order .....	56
5.2 Vehicle Tutorial .....	58
5.2.1 Creating the Mechanized Infantry Group .....	58
5.2.2 Creating a Mount Order .....	59
5.2.3 Creating an Advance Order .....	60
5.2.4 Creating a Dismount Order .....	61
5.3 Interoperability Tutorial .....	62
5.3.1 Creating the Groups .....	62
5.3.2 Assigning Orders .....	63
5.3.3 SQF Commands / Functions Interoperability .....	66
5.4 Military AI Orders and Behaviors .....	67
5.4.1 Assault Order .....	70
5.4.2 Advance Order .....	72
5.4.3 Pursue Order .....	74
5.4.4 Suppress Order .....	77
5.4.5 Defend Order .....	80
5.4.6 Mount Order .....	82
5.4.7 Dismount Order .....	86
5.4.8 Return to Formation Order .....	89
5.4.9 Tactical Move Order .....	90
<b>6. Convoy AI .....</b>	<b>93</b>
6.1 Fully Autonomous Convoy .....	94
6.2 Pause / Resume Convoy .....	96
6.3 Player Units in Convoy .....	101
6.4 Convoy SQF Functions .....	103
6.4.1 fn_vbsCon_pauseConvoy .....	103
6.4.2 fn_vbsCon_resumeConvoy .....	104
6.5 Convoy Order .....	105
6.5.1 Troubleshooting .....	111
6.5.2 Limitations .....	111

<b>7. Bridge Laying Convoy AI .....</b>	<b>112</b>
7.1 Deploy DSB Order .....	116
7.1.1 Waypoint Completion Settings .....	117
7.1.2 Advanced Settings .....	117
<b>8. Aircraft AI .....</b>	<b>119</b>
8.1 Aircraft Tutorial .....	120
8.1.1 Creating the Entities .....	120
8.1.2 Creating the Flight Path .....	121
8.2 Aircraft AI Orders and Behaviors .....	123
8.2.1 Fly Order .....	124
8.2.2 Land Order .....	127
8.2.3 Loiter Order .....	129
<b>9. Civilian AI .....</b>	<b>132</b>
9.1 Define Populations .....	133
9.1.1 Specify Civilian Settings .....	134
9.1.2 Specify Civilian Populace .....	136
9.1.3 Specify Vehicle Traffic .....	138
9.1.4 Edit Populations .....	140
9.2 Define Pedestrian Flows .....	141
9.2.1 Create Activities .....	142
9.2.2 Create Areas .....	144
9.2.3 Edit Activities .....	146
9.2.4 Edit Areas .....	147
9.3 Define Civilian Riot .....	148
9.4 Define Traffic Flows .....	151
9.4.1 Create Traffic Spawn / Despawn Points .....	152
9.4.2 Create Traffic Areas .....	153
9.4.3 Traffic Signs and Lights .....	154
9.4.4 Traffic Debug Visualization .....	159
9.5 Defining Responsive Behavior .....	163
9.5.1 Interrupt Activity Sub-Types .....	163

9.5.2 Resetting Panic .....	164
9.5.3 Enabling / Disabling the Population .....	165
9.5.4 Initiating Activities .....	166
9.6 Debug Visualization .....	168
9.6.1 Display Narrow Roads .....	169
9.6.2 Display Problematic Turns .....	169
9.6.3 Display Problematic Intersections .....	170
9.6.4 Display Objects Interfering with Road Traffic .....	170
9.6.5 Display Junctions for Crosswalks and Crossroads .....	171
9.7 Basic Example Scenario .....	172
9.7.1 Define Smart Objects .....	173
9.7.2 Define Departing Passengers .....	175
9.7.3 Define Arriving Passengers .....	179
9.7.4 Define the City Population .....	179
9.7.5 Define the Threat .....	181
9.7.6 Define the Safe Zone .....	184
9.7.7 Define the Restricted Area .....	186
9.7.8 Run the Scenario as Administrator .....	187
9.8 Civilian AI Editor Objects .....	188
9.8.1 Population Editor Object .....	189
9.8.2 Activity Editor Object .....	192
9.8.3 Area Editor Object .....	199
9.8.4 Smart Objects .....	200
<b>10. Animal AI .....</b>	<b>202</b>
10.1 Creating a Herd .....	202
10.2 Reaction to Threats .....	204
<b>11. Navigation Meshes .....</b>	<b>205</b>
11.1 Updating the Navigation Mesh at Runtime .....	208
<b>12. Control AI Visualization .....</b>	<b>210</b>
12.1 Movement Mode .....	211
12.1.1 Show .....	212

12.2 Path-Planning .....	212
12.2.1 Show .....	213
12.2.2 Show Path Polys .....	214
12.2.3 Path Polys Vertical Offset .....	214
12.2.4 Flatten Path Polys .....	214
12.3 Navigation Mesh .....	214
12.3.1 Visible Nav Mesh .....	215
12.3.2 Visible Nav Mesh Artifact .....	217
12.3.3 Generate Nav Mesh .....	218
12.3.4 Show Roads .....	218
12.3.5 Generate Roads .....	219
12.3.6 Vertical Offset .....	219
12.3.7 Display Radius .....	219
12.3.8 Polygon Type ID .....	219
12.4 Threats .....	220
12.4.1 Show Threats .....	220
12.4.2 Show in Radius .....	221
12.5 Cover .....	221
12.5.1 Show Cover .....	222
12.5.2 Generate Cover .....	222
12.5.3 Cover Draw Radius .....	223
12.6 Async Task Queue .....	223
12.7 Obstacle Avoidance .....	223
12.7.1 Show .....	224
12.7.2 Vertical Offset .....	224
12.7.3 Scale .....	224
12.7.4 Show Braking Penalty .....	224
12.7.5 Show Velocity Penalty .....	225
12.7.6 Highlight Minimal Penalty .....	225
12.8 Actor Velocity .....	225
12.8.1 Show .....	229

12.8.2 Vertical Offset .....	229
12.8.3 Scale .....	229
12.9 Polygon Settings .....	229
12.9.1 Polygon Line Width .....	229
12.9.2 Polygon Fill Opacity .....	229
<b>13. Example Content .....</b>	<b>230</b>
13.1 Additional Example Content .....	230
<b>14. Known Issues .....</b>	<b>231</b>
14.1 Military .....	231
14.2 Convoy .....	232
14.3 Bridge Laying Convoy .....	233
14.4 Aircraft .....	234
14.5 Civilian .....	234
14.6 Civilian Traffic .....	235
14.7 Animal .....	236
14.8 General .....	237

# 1. VBS Control AI Overview

VBS Control AI provides a set of AI behaviors with Computer Generated Forces (CGF)-like capabilities. It offers a familiar VBS Editor based user interface.

Control AI uses [Waypoints \(on page 23\)](#) and has the following capabilities:

Capabilities	AI Sub-Type	Additional Aspects
Convoy Training, Driver Training, Enemy Ambush, OPFOR Tactics, or any vehicle focused capabilities.  For more information on convoy capabilities, see Convoys in the Introduction to VBS4 Guide.	<a href="#">Convoy AI (on page 93)</a>	The Control AI convoy has the following additional aspects: <ul style="list-style-type: none"> <li>The convoy has behavior that can be paused / continued.</li> <li>The convoy has a leader, which is dynamically selectable and can be replaced.</li> <li>The convoy dynamically determines the order of convoy succession, based on the convoy vehicle placement.</li> <li>The convoy can drive both on and off the road.</li> <li>The convoy reactions to enemy contact or fire can be set.</li> </ul>
Enemy Ambush, OPFOR Tactics.  For more information on these capabilities, see Enemy Ambush in the Introduction to VBS4 Guide.	<a href="#">AI on Rails (on page 36)</a>	<ul style="list-style-type: none"> <li>Fine-tune enemy placement to prepare an ambush.</li> <li>Use Triggers to create dynamic situations.</li> <li>Branch Waypoints to achieve different outcomes on each Scenario run.</li> </ul>
Dry Support Bridge (DSB) Training.  For more information about DSB laying capabilities, see HX45M Bridge Laying (Land 155) in the Introduction to VBS4 Guide.	<a href="#">Bridge Laying Convoy AI (on page 112)</a>	<ul style="list-style-type: none"> <li>Place DSBs of varying lengths at the required places on the terrain.</li> </ul>
Large-scale battles with a primary focus on fighting and maneuvers.	VBS Plan (see VBS Plan Overview in the VBS Plan Manual)	<ul style="list-style-type: none"> <li>Place large amounts of military entities easily, using VBS Plan.</li> <li>Have the entities maneuver and attack the enemy on their way.</li> <li>Place tanks or mechanized platoons.</li> </ul>
Military infantry and vehicle fighting and maneuvers.	<a href="#">Military AI (on page 53)</a>	<ul style="list-style-type: none"> <li>Have infantry and vehicle AI perform general-purpose military orders.</li> </ul>

Capabilities	AI Sub-Type	Additional Aspects
Rotary-wing aircraft maneuvering.	Aircraft AI (on page 119)	<ul style="list-style-type: none"> <li>Have rotary-wing aircraft AI fly to the specified location.</li> </ul>
Civilian pattern of life.	Civilian AI (on page 132)	<ul style="list-style-type: none"> <li>Create a civilian pattern of life that consists of pedestrians and vehicles.</li> <li>Create various activities for pedestrians.</li> </ul>
Animal-herd behaviors.	Animal AI (on page 202)	<ul style="list-style-type: none"> <li>Add animal-herd movement.</li> </ul>

For an overview of the Control AI UI and its Editor Objects, see [Control AI UI Overview \(on page 17\)](#).

### Reference:

- [Navigation Meshes \(on page 205\)](#) - Describes how to create and update a map navigation mesh.
- [Control AI Visualization \(on page 210\)](#) - Describes how to use debug visualization for Control AI entities.
- [Example Content \(on page 230\)](#) - Lists the Control AI example missions, available with VBS Control AI.
- [Known Issues \(on page 231\)](#) - Lists the current VBS Control AI issues / limitations.

## 1.1 Disabled and Incompatible Features

The legacy Game AI was fully substituted by Control AI in VBS4 24.1.0.

### Automatic Battlespace Conversion

When loading Battlespaces, VBS4 automatically converts any legacy Game AI waypoints to their closest equivalent Control AI waypoints. For more information, see [Waypoints \(on page 23\)](#).

Confirm that the Battlespace still works as expected and save it to make the changes permanent.

If your Battlespace makes use of scripting for AI, check the commands used against the list of commands supported by Control AI (see [Commands and Functions Compatible with Control AI \(https://sqf.bisimulations.com/display/SQF/Control+AI#ControlAI-CommandsandFunctionsCompatiblewithControlAI\)](#) and [Commands and Functions Not Compatible with Control AI \(https://sqf.bisimulations.com/display/SQF/Control+AI#ControlAI-CommandsandFunctionsNotCompatiblewithControlAI\)](#)). In the event that you use scripting commands that are not supported by Control AI, verify the functionality of your Battlespace, and update the scripting as necessary.

The following legacy Game AI VBS4 features have been permanently removed from VBS4:

## Artificial Intelligence Simulation Settings

The following Artificial Intelligence Simulation Settings (see the VBS4 Administrator Manual) were removed:

- AI Voices
- Auto Report
- BLUFOR Surrender
- INDEPENDENT Surrender
- Long Term Morale
- Morale Simulation
- OPFOR Surrender
- Strong Morale
- Super AI

These settings will be replaced by other features in future releases of VBS4.

## Game AI Waypoints

All Game AI waypoints were removed, and replaced by Control AI ones - see [Waypoints \(on page 23\)](#).

The following table specifies the Game AI waypoint and the matching Control AI Order replacement (see the respective Orders):

Game AI Waypoint	Replacement	Comment
Cycle	Create Cycle	See <a href="#">Waypoints (on page 23)</a> .
Destroy	Pursue Order (on page 74)	Partial support, full support in future VBS4 releases.
Dismissed	None	Removed
Get In	Mount Order (on page 82)	
Get In Nearest	Mount Order (on page 82)	
Get Out	Dismount Order (on page 86)	
Guard	Defend Order (on page 80)	
Hold	Defend Order (on page 80) (Guard)	Partial support, full support in future VBS4 releases.
Join	Return to Formation Order (on page 89)	The <b>Return to Formation Order</b> Control AI waypoint is different from the <b>Join</b> Game AI waypoint, and may require additional Battlespace modifications, depending on the scenario needs.

Game AI Waypoint	Replacement	Comment
<b>Join and Lead</b>	Return to Formation Order (on page 89)	The <b>Return to Formation Order</b> Control AI waypoint is different from the <b>Join and Lead</b> Game AI waypoint, and may require additional Battlespace modifications, depending on the scenario needs.
<b>Land</b>	Land Order (on page 127)	Partial support, full support in future VBS4 releases.
<b>Load</b>	Mount Order (on page 82)	Partial support, full support in future VBS4 releases.
<b>Loiter</b>	Loiter Order (on page 129)	Partial support, full support in future VBS4 releases.
<b>Move</b>	Advance Order (on page 72) Fly Order (on page 124)	
<b>Retreat</b>	Advance Order (on page 72)	Partial support, full support in future VBS4 releases.
<b>Scripted</b>	None	Removed
<b>Seek and Destroy</b>	Pursue Order (on page 74)	
<b>Sentry</b>	Defend Order (on page 80) (Guard)	Partial support, full support in future VBS4 releases.
<b>Support</b>	Suppress Order (on page 77)	
<b>Talk</b>	None	Removed
<b>Transport</b> <b>Unload</b>	Dismount Order (on page 86)	Partial support, full support in future VBS4 releases.
<b>Unload</b>	Dismount Order (on page 86)	Partial support, full support in future VBS4 releases.

## VBS4 Features

The following VBS4 features, previously compatible with Game AI, are either currently / temporarily disabled or partially supported for Control AI, and will either be re-enabled or equivalent functionality will be added in future versions of VBS4:

Feature	Description
Active Protection System (APS) Editor Object	Turret manipulation by a Control AI vehicle crew is unsupported on APS-enabled vehicles. For more information, see Active Protection System in the VBS4 Editor Manual.
AI Rules of Engagement Editor Object	The AI Rules of Engagement Editor Object <b>Move Options</b> only support the <b>Full</b> option for Control AI. Also, the <b>Stance</b> options are not supported for Control AI. For more information, see AI Rules of Engagement in the VBS4 Editor Manual.
Airstrike Editor Object	The Airstrike Editor Object is disabled for Control AI.
Bomb Detection Dog (MWD)	The Bomb Detection Dog / Military Working Dog (MWD) can only be controlled by players, but not Control AI. For more information, see Bomb Detection Dog (MWD) in the VBS4 Trainee Manual.
Camera Editor Object	It is not possible to record the Camera Editor Object waypoints path using the <b>Record to Disk?</b> option. For more information, see Camera Editor Object in the VBS4 Editor Manual.
Close Air Support (CAS) Editor Object	Using helicopters for CAS may currently result in unexpected behavior of the aircraft, especially over uneven terrain. For more information, see Close Air Support in the VBS4 Editor Manual.
Command Menu	Not all the First / Third Person Command Menu options are supported for Control AI. For the currently available Command Menu options, see Commanding Subordinates in the VBS4 Trainee Manual.
Complex Animations Editor Object	The Complex Animations Editor Object is disabled for Control AI. Some animations may be achieved by using the Control AI - Activity Editor Object - see <a href="#">Defining Responsive Behavior (on page 163)</a> , or by using custom behaviors - see <a href="#">Custom Behaviors (on page 20)</a> .
Concussion Effects Editor Object	It is not possible to use the Concussion Effects Editor Object with waypoints. For more information, see Concussion Effects in the VBS4 Editor Manual.
Fast Rope	The Fast Rope Editor Object is disabled for Control AI. For more information, see Fast Rope in the VBS4 Editor Manual.
Fire Support	The entities providing fire support cannot follow the waypoints after the fire support is completed. For more information, Fire Support in the VBS4 Instructor Manual.
Flee Point Editor Object	The Flee Point Editor Object is disabled for Control AI. Instead, use the Control AI - Activity Editor Object - see <a href="#">Defining Responsive Behavior (on page 163)</a> .
Formation Editor	The real-time (Preview / Execute mode) Formation Editor is currently disabled for Control AI. Use the ORBAT Formations Editor (see the VBS4 Editor Manual) in Prepare mode instead.

Feature	Description
Game AI Waypoints	<p>Game AI waypoints currently with no Control AI equivalents:</p> <ul style="list-style-type: none"><li>• Repair</li><li>• Taxi</li></ul> <p>Game AI waypoint settings currently not present in Control AI waypoints:</p> <ul style="list-style-type: none"><li>• Timeout</li></ul> <p>For more information, see <a href="#">Waypoints (on page 23)</a>.</p>
LCAC	The LCAC Transport Hovercraft cannot be used by Control AI. For more information, see LCAC Transport Hovercraft in the VBS4 Trainee Manual.
MEDEVAC	Only possible with rotary-wing aircraft - see Request MEDEVAC / CASEVAC in the VBS4 Trainee Manual.
Mission Briefing	The Mission Briefing (see Mission Briefings in the VBS4 Editor Manual) does not display waypoints.
Observe Sector	Commanding units to observe a sector cannot be used with Control AI. For more information, see Suppress Area in the VBS4 Trainee Manual.
Sandstorm Editor Object	It is not possible to use the Sandstorm Editor Object with waypoints. For more information, see Sandstorm in the VBS4 Editor Manual.
Skydiving	Control AI units cannot use parachutes. For more information, see Skydiving in the VBS4 Trainee Manual.
Surrender Editor Object	The Surrender Editor Object has some limitations for Control AI. For more information, see Surrender in the VBS4 Editor Manual.
Unit Psychological Settings	<p>The following unit psychological settings (see Adding Units in the VBS4 Editor Manual) are unavailable to Control AI:</p> <ul style="list-style-type: none"><li>• Training</li><li>• Navigation</li><li>• Experience</li><li>• Leadership</li><li>• Fleeing Chance</li></ul>
Unit Swimming	Control AI units cannot swim. For more information, see Underwater Diving in the VBS4 Trainee Manual.

Feature	Description
Weapon Simulation	<p>The following weapon simulation is partially supported / unsupported for Control AI usage:</p> <ul style="list-style-type: none"><li>• AT Missiles (Guided) (partially supported)</li><li>• SAM (Man-Portable) (unsupported)</li><li>• SAM (Vehicle) (unsupported)</li><li>• Air To Ground Systems (partially supported)</li><li>• Laser Designator (Control AI units cannot engage laser-designated targets - see Laser Designator in the VBS4 Trainee Manual)</li></ul>

## 2. Control AI UI Overview

The Control AI functionality in VBS4 consists of the following:

- [Control AI Editor Objects \(below\)](#) - Control AI Editor Objects, available in VBS Editor (Prepare / Execute Mode).
- [Control AI Menu \(on page 21\)](#) - Available in VBS Editor (Prepare / Execute Mode), allows you to reload Control AI behaviors and contains the Control AI visualizations for Control AI Editor Objects.

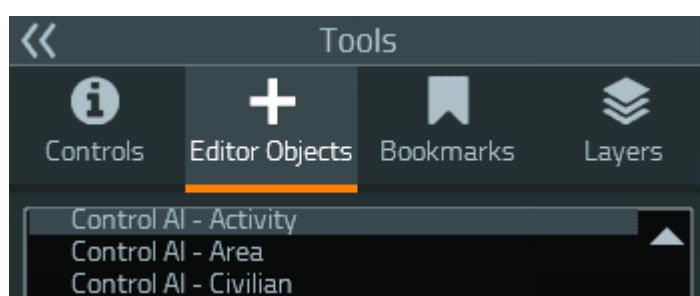
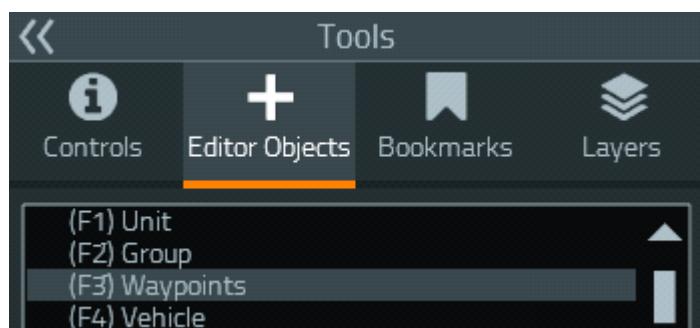
### 2.1 Control AI Editor Objects

Predefined Control AI has the following Editor Objects associated with it:

- [\(F3\) Waypoints Editor Object \(on the next page\)](#)
- [Control AI - Civilian Editor Object \(on page 20\)](#)
- [Control AI - Activity Editor Object \(on page 20\)](#)
- [Control AI - Area Editor Object \(on page 21\)](#)

You can also use a Custom Control AI behaviors - see [Custom Behaviors \(on page 20\)](#).

**Image-1: Control AI Editor Objects in VBS Editor**



## 2.2 (F3) Waypoints Editor Object

The (F3) Waypoints Editor Object in VBS Editor (Prepare / Execute Mode) is used to run various infantry and vehicle military behaviors, civilian riot behavior (protest marches and on-site demonstrations), and animal herd movement.

For how-tos on using this Editor Object with existing / predefined Control AI behaviors, see:

### [AI on Rails \(on page 36\)](#)

Provide an infantry entity with a sequence of orders that can be visualized in 3D.

AI on Rails uses the following orders:

- [Individual - Fire At Order \(on page 47\)](#) - The entity assumes the specified stance and speed, moves to target position, orients itself according to the specified heading, and fires at the specified target.
- [Individual - Move Order \(on page 50\)](#) - The entity assumes the specified stance and speed, moves to target position, and orients itself according to the specified heading.

### [Military AI \(on page 53\)](#)

Use the Editor Object to control general-purpose group infantry and vehicle AI.

The following orders / behaviors are available:

- [Assault Order \(on page 70\)](#)

The group (infantry / vehicles / both) moves in formation to the given position (the location of the [\(F3\) Waypoints Editor Object \(above\)](#) on the map), and assaults the enemy.

Infantry groups split into fireteams, form a line, and perform bounding to the given position to assault the enemy.

- [Advance Order \(on page 72\)](#)

The group (infantry / vehicle / both) advances in formation to the given position (the location of the [\(F3\) Waypoints Editor Object \(above\)](#) on the map), choosing the fastest path to the destination, while preferring speed of movement over engaging and eliminating the enemy.

- [Pursue Order \(on page 74\)](#)

The infantry team (only fireteam behavior) pursues the enemy, until it is eliminated. Then, the team takes cover at their current position.

- [Suppress Order \(on page 77\)](#)

The infantry and / or vehicle group forms a firing line and starts firing at the given position area (the location of the [\(F3\) Waypoints Editor Object \(above\)](#) on the map).

- [Defend Order \(on page 80\)](#)

The infantry and / or vehicle group moves to the given position (the location of the [\(F3\) Waypoints Editor Object \(on the previous page\)](#) on the map), forms a line and begins to defend it by firing at enemy forces in the position area.

- [Mount Order \(on page 82\)](#)

The infantry team moves to the given position (the location of the [\(F3\) Waypoints Editor Object \(on the previous page\)](#) on the map), and mounts the nearest vehicle at the selected vehicle positions (crew, cargo, or both).

- [Dismount Order \(on page 86\)](#)

The infantry teams dismounts the vehicle from the selected vehicle positions (crew, cargo, or both).

- [Return to Formation Order \(on page 89\)](#)

The infantry / vehicle group returns back under the command of a higher-echelon group.

- [Tactical Move Order \(on page 90\)](#)

The infantry group performs a tactical move to the given position (the location of the [\(F3\) Waypoints Editor Object \(on the previous page\)](#) on the map), using a given stance, formation, and a maximum range to engage enemy forces at.

## Convoy AI (on page 93)

Use the Editor Object to control vehicle convoy AI.

Convoy AI uses the [Convoy Order \(on page 105\)](#).

## Bridge Laying Convoy AI (on page 112)

Use the Editor Object to control bridge-laying vehicle convoy AI.

Bridge-laying convoy AI uses the [Deploy DSB Order \(on page 116\)](#).

## Aircraft AI (on page 119)

Use the Editor Object to control rotary-wing aircraft AI.

Aircraft AI uses the [Fly Order \(on page 124\)](#), [Land Order \(on page 127\)](#), and [Loiter Order \(on page 129\)](#).

## Define Civilian Riot (on page 148)

Use the Editor Object to add riot behavior to civilian groups, to simulate protest marches and on-site demonstrations.

### NOTE

Other aspects of the civilian AI use the [Control AI - Civilian Editor Object \(on the next page\)](#).

## Animal AI (on page 202)

Use the Editor Object to control animal-herd AI.

### [Waypoints \(on page 23\)](#)

Use the (F3) Waypoints Editor Object to add waypoints to create paths.

## 2.3 Custom Behaviors

The Custom behavior option is part of the (F3) Waypoints Editor Object in VBS Editor (Prepare / Execute Mode) and allows you to use your own Behavior Tree. A Custom behavior is any behavior specified by Behavior Trees for an individual entity and / or group.

For more information, see Custom Behaviors in the VBS Control Editor Manual .

Novice and intermediate Control AI developers can also see the following how-tos in the VBS Control Editor Manual, which rely on using the Custom behavior option in VBS4:

- How to Build Basic Behaviors
- How to Build Scripted Behaviors

## 2.4 Control AI - Civilian Editor Object

The Control - Civilian Editor Object in VBS Editor (Prepare / Execute Mode) is used to run civilian AI. It is used as a pattern of life generator, and can simulate a crowd of civilians, engaged in various activities.

For a how-to on using this Editor Object, see:

- [Civilian AI \(on page 132\)](#)

Create a civilian pattern of life.

Civilian AI uses the following Editor Objects:

- [Population Editor Object \(on page 189\)](#) - The primary Editor Object with which you can create a civilian pattern of life, consisting of civilian pedestrians and vehicles.
- [Activity Editor Object \(on page 192\)](#) - Allows you to define civilian pedestrian and vehicle flows.
- [Area Editor Object \(on page 199\)](#) - Allows you to create areas for civilian activities.

## 2.5 Control AI - Activity Editor Object

The Control AI - Activity Editor Object in VBS Editor (Prepare / Execute Mode) is used to set activities for civilian AI.

For a how-to on using this Editor Object, see [Civilian AI \(on page 132\)](#).

For the Editor Object properties reference, see [Activity Editor Object \(on page 192\)](#).

## 2.6 Control AI - Area Editor Object

The Control AI - Area Editor Object in VBS Editor (Prepare / Execute Mode) defines civilian behavior in the specified area.

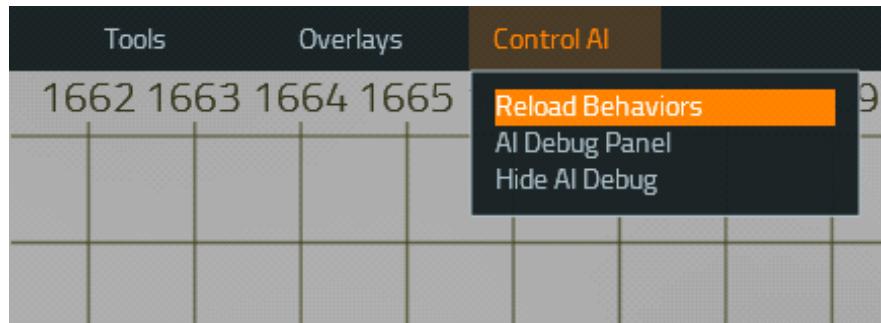
For a how-to on using this Editor Object, see [Civilian AI \(on page 132\)](#).

For the Editor Object properties reference, see [Area Editor Object \(on page 199\)](#).

## 2.7 Control AI Menu

The Control AI menu in VBS Editor (Prepare / Execute Mode) allows you to reload and debug behaviors.

**Image-2: The Control AI menu in VBS Editor (Prepare / Execute Mode)**



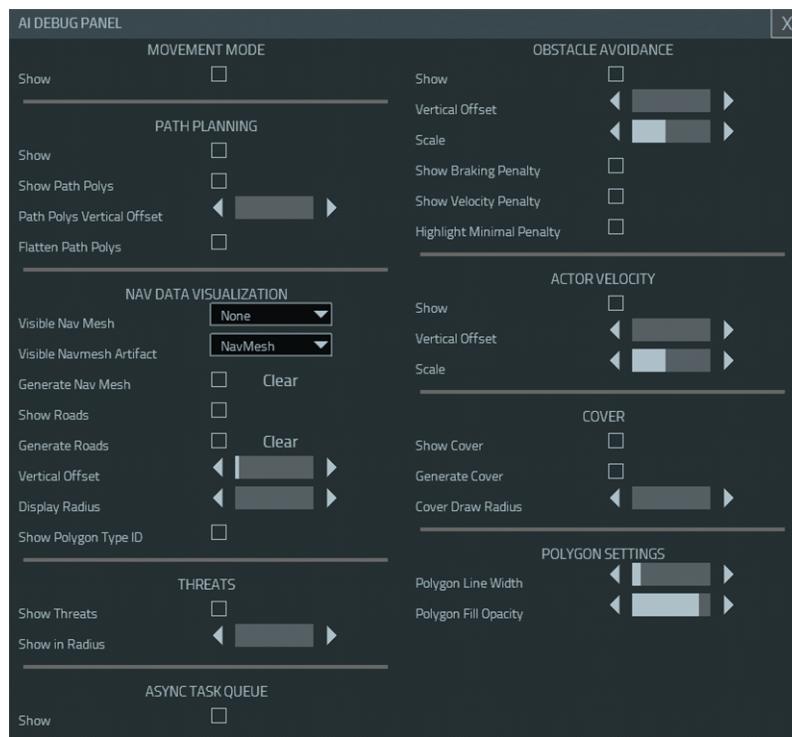
The Control AI menu contains the following menu options:

**Reload Behaviors** - Reloads behaviors. Used when the behaviors are updated and require reloading.

**AI Debug Panel** and **Hide AI Debug** - The AI Debug Panel provides Control AI debugging functionality and contains path-planning and navigation-mesh visualization (in the 3D view of the VBS Editor (Prepare / Execute Mode)).

To open the AI Debug Panel, in VBS Editor (Prepare / Execute Mode), open the menu **Control AI > AI Debug Panel**.

### Image-3: The AI Debug Panel



To hide all the debug visualizations, in VBS Editor (Prepare / Execute Mode) main menu, select **Control AI > Hide AI Debug**. To switch them on again, select **AI Debug Panel**.

For more information about the visualization settings, see [Control AI Visualization \(on page 210\)](#).

## 3. Waypoints

Link an AI entity / group to a [\(F3\) Waypoints Editor Object \(on page 18\)](#) to use it as a waypoint. You can link several waypoints to create a complex path.

### **WARNING**

The [\(F3\) Waypoints Editor Object \(on page 18\)](#) behaviors are not fully optimized for VBS4, and can cause reduced performance, when used in large quantities in the Scenario.

For SQF waypoint functions and their parameters, see [Waypoint Functions and Parameters \(on page 28\)](#).

### **WARNING**

These SQF functions and their parameters are experimental and subject to change in future releases of VBS4.

Create an AI entity / group by adding units or vehicles using **(F1) Unit / (F4) Vehicle / (F2) Group** in the Editor Objects List (to link the units / vehicles to create a group, see Creating and Adding to Groups with Links in the VBS4 Editor Manual).

**To create a waypoint for an entity / group in the Editor (Prepare / Execute Mode), choose one of the following approaches:**

- Using only an entity / group object:
  1. Right-click the entity / group and select **Orders > Assign New Waypoint**, then click a position on the map to create a waypoint.
  2. In the **Behavior** list, select the AI Order (waypoint behavior), set the Order properties, and click **OK**.
- Using two Editor Objects - an entity / group object and a [\(F3\) Waypoints Editor Object](#):
  1. Make sure that **(F3) Waypoints** is selected in the Editor Objects List (you can press **F3** to select it) and place it on the map to indicate the position of the waypoint.
  2. In the **Behavior** list, select the AI Order (waypoint behavior), set the Order properties, and click **OK** to create the waypoint.
  3. Right-click the entity / group and select **Orders > Assign Existing Waypoint**, then click the waypoint.

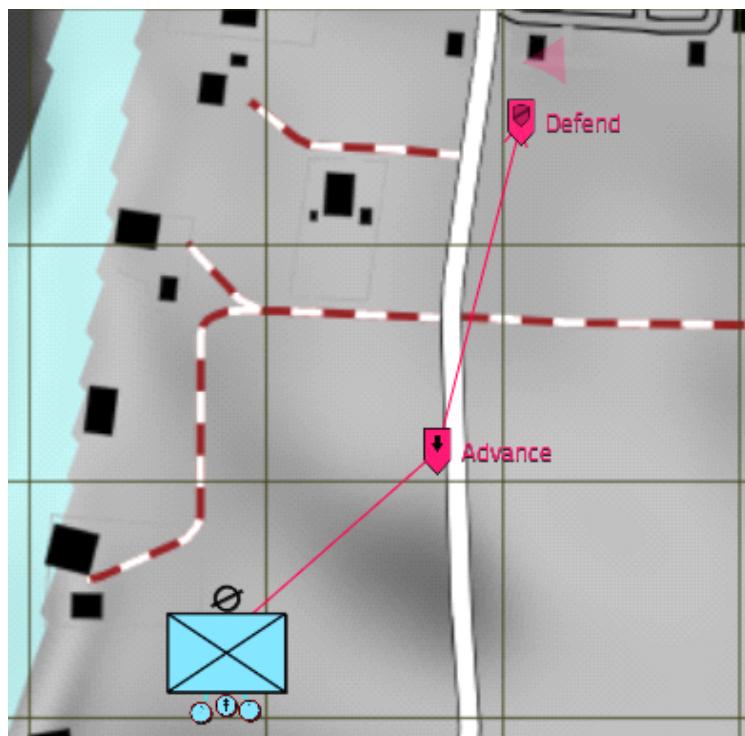
The entity / group has a waypoint.

To create a path / chain that consists of waypoints, choose one of the following approaches:

- Using one waypoint:
  1. Create a waypoint based on the previous procedure.
  2. Right-click the waypoint and select **Assign Next Waypoint**, then click a position on the map for the next waypoint.
- Using two or more waypoints:
  1. Create two or more waypoints based on the previous procedure.
  2. Press **Shift + LMB** on the first waypoint, and click the second waypoint to create a link that defines the order in which the waypoints are completed. Proceed in the same fashion with the remaining waypoints.
  3. (Optional) You can also create a loop / cycle of waypoints: right-click the last waypoint, select **Create Cycle**, then click the first waypoint.

The two or more waypoints are linked to create a more complex path.

**Image-4: An example path of two waypoints**



For information on how to link existing (F3) Waypoints Editor Objects and other Editor Objects, see [Linking Existing Waypoints to Other Editor Objects \(on the next page\)](#).

## 3.1 Linking Existing Waypoints to Other Editor Objects

You can link existing (F3) Waypoints Editor Objects to the following Editor Objects (EOs) in the Editor Objects List:

- **(F1) Unit** - Assigns a waypoint to a unit (see Adding Units in the VBS4 Editor Manual).
- **(F2) Group** - Assigns a waypoint to a group (see Adding Groups in the VBS4 Editor Manual).
- **(F4) Vehicle** - Assigns a waypoint to a vehicle (see Adding Vehicles in the VBS4 Editor Manual).
- **(F7) Trigger** - Synchronizes a waypoint with a trigger (see Triggers in the VBS4 Editor Manual). For waypoint-to-waypoint synchronization, see [Waypoint Synchronization \(on the next page\)](#).

### Follow these steps:

1. Do one of the following:
  - For the **(F1) Unit / (F2) Group / (F4) Vehicle** EOs, right-click the (F3) Waypoints Editor Object, select **Link to Entity**.
  - For the **(F7) Trigger** EO, right-click the (F3) Waypoints Editor Object, select **Link to Trigger**.
2. Click the EO.

The link between the (F3) Waypoints EO and the other EO is established.

## 3.2 Branching Waypoints

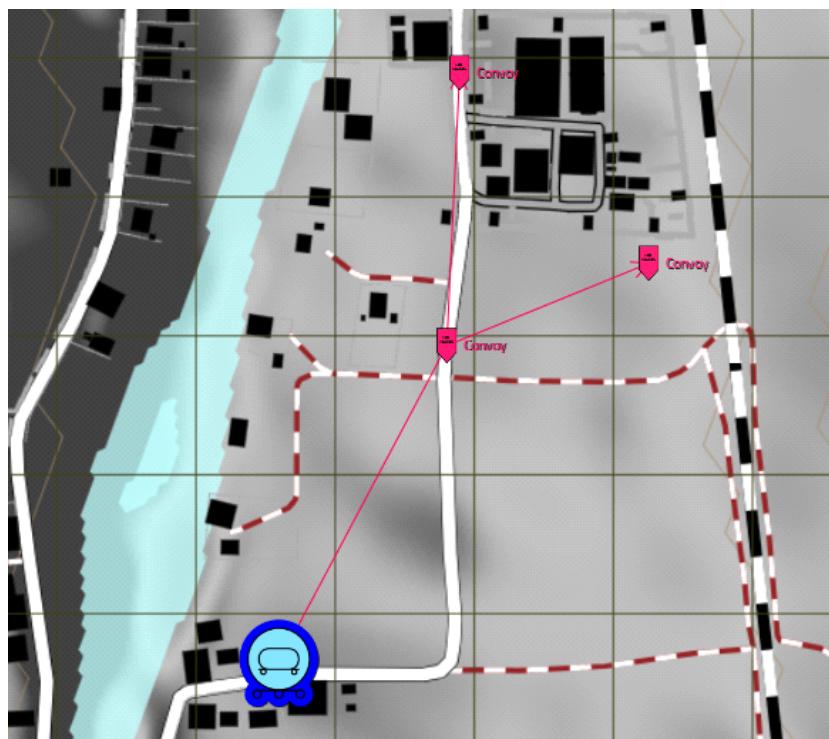
You can create branching waypoints for AI entities / groups to choose from (for example, based on trigger conditions).

### Follow these steps:

1. Create an AI unit (or a vehicle that contains AI units, if creating a convoy) and select it.
2. Right-click the unit and select **Orders > Assign New Waypoint**, and then click the map where you want the branching waypoint to be.
3. In the **Behavior** list, select the AI Order (waypoint behavior), set the Order properties, and click **OK**.
4. Right-click the branching waypoint and select **Assign Next Waypoint**, and click the map, where a branch of the waypoint should be. Update the branch settings as required and click **OK**.
5. Repeat step 4, until you have all the waypoint branches.

The AI now has a branching waypoint. For a more detailed example, see [Branching Orders \(on page 44\)](#).

### Image-5: Convoy branching waypoints



## 3.3 Waypoint Synchronization

You can synchronize the execution of one waypoint / Order with another.

**Follow these steps:**

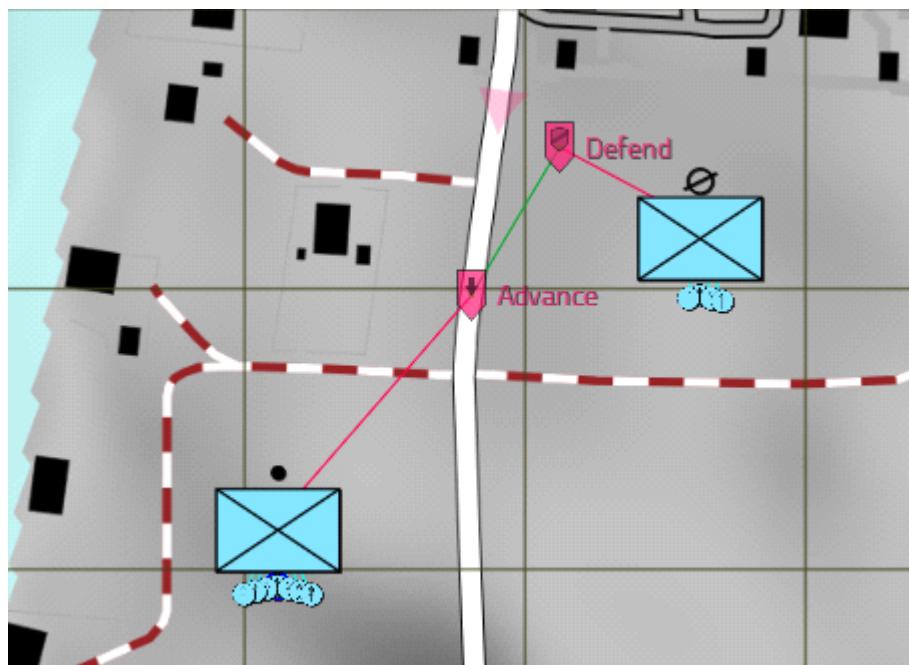
1. Create two waypoints.
2. Right-click one waypoint, and select **Sync to Waypoint**.
3. Click the other waypoint you want to synchronize with the first one.

The two waypoints are synchronized.

**NOTE**

You can synchronize one waypoint to more than one waypoint.

To remove the waypoint synchronization, right-click the waypoint you want to unsynchronize and select **Sync to Waypoint**, then either click the specific waypoint you want to unsynchronize from, or click an empty location on the map to unsynchronize the former waypoint from all the waypoints it is synchronized with.

**Image-6: Synchronized Advance and Defend waypoints**

### 3.4 Waypoint Considerations

The following considerations apply to waypoints:

- It is possible to create chains of waypoints.
- Waypoint chains can contain branching waypoints.
- Deleting a waypoint in a chain automatically links the other waypoints to the next waypoint (if there is any).
- Waypoint name types and any waypoint status information is only displayed, if the waypoint is selected.

## 3.5 Waypoint Functions and Parameters

Waypoints can be created, modified, and assigned using the following SQF functions:

- [fn\\_vbsCon\\_waypointCreate](https://sqf.bisimulations.com/display/SQF/fn_vbsCon_waypointCreate) ([https://sqf.bisimulations.com/display/SQF/fn\\_vbsCon\\_waypointCreate](https://sqf.bisimulations.com/display/SQF/fn_vbsCon_waypointCreate)) - Creates a waypoint.
- [fn\\_vbsCon\\_waypointUpdate](https://sqf.bisimulations.com/display/SQF/fn_vbsCon_waypointUpdate) ([https://sqf.bisimulations.com/display/SQF/fn\\_vbsCon\\_waypointUpdate](https://sqf.bisimulations.com/display/SQF/fn_vbsCon_waypointUpdate)) - Modifies an existing waypoint.
- [fn\\_vbsCon\\_waypointAssign](https://sqf.bisimulations.com/display/SQF/fn_vbsCon_waypointAssign) ([https://sqf.bisimulations.com/display/SQF/fn\\_vbsCon\\_waypointAssign](https://sqf.bisimulations.com/display/SQF/fn_vbsCon_waypointAssign)) - Assigns an existing waypoint to a unit, vehicle, group, or another waypoint.

### **WARNING**

These SQF functions and their parameters are experimental and subject to change in future releases of VBS4.

Each of these SQF functions use the following waypoint parameters, which correspond to different Order behaviors (for the Order behavior class names - see [Order Behavior Class Names \(on page 34\)](#)).

For more information on the data types, see the respective data types in the VBS4 Scripting Manual.

Parameter (Data Type and Specific Values)	VBS4 Name / Description	Order Behaviors
<b>ACCURACY</b> <b>Data Type:</b> Number	Accuracy	Individual - Fire At Order (on page 47) Individual - Move Order (on page 50)
<b>AIRCRAFT_LOITER_DIRECTION</b> <b>Data Type:</b> String <b>Specific Values:</b> "clockwise", "counterClockwise"	Direction	Loiter Order (on page 129)
<b>AIRCRAFT_LOITER_RADIUS</b> <b>Data Type:</b> Number	Loiter Radius	Loiter Order (on page 129)
<b>AZIMUTH</b> <b>Data Type:</b> Number	Azimuth	Defend Order (on page 80)
<b>BEHAVIOR_ENTITY</b> <b>Data Type:</b> String	Behavior Tree (Entity)	Custom (see <a href="#">Custom Behaviors (on page 20)</a> )
<b>BEHAVIOR_GROUP</b> <b>Data Type:</b> String	Behavior Tree (Group)	Custom (see <a href="#">Custom Behaviors (on page 20)</a> )

Parameter (Data Type and Specific Values)	VBS4 Name / Description	Order Behaviors
<b>BEHAVIOR_LINK</b> <b>Data Type:</b> Object	Assign Next Waypoint	All
<b>BEHAVIOR</b> (see <a href="#">Order Behavior Class Names (on page 34)</a> ) <b>Data Type:</b> String	Behavior	All
<b>BTSET</b> <b>Data Type:</b> String	BT Set Name	Custom (see <a href="#">Custom Behaviors (on page 20)</a> )
<b>BTSETPATH</b> <b>Data Type:</b> String	BT Set Path	Custom (see <a href="#">Custom Behaviors (on page 20)</a> )
<b>CODE_ON_COMPLETION</b> <b>Data Type:</b> Code	Code on Completion	All
<b>CONDITION_TO_COMPLETE</b> <b>Data Type:</b> Code	Condition to Complete	All
<b>CONVOYREACTIONTOCONTACT</b> <b>Data Type:</b> String <b>Specific Values:</b> "driveIgnore", "driveFire"	On Visual Contact	<a href="#">Convoy Order (on page 105)</a>
<b>CONVOYREACTIONTOFIRE</b> <b>Data Type:</b> String <b>Specific Values:</b> "driveIgnore", "driveFire", "haltEngage"	On Taking Fire	<a href="#">Convoy Order (on page 105)</a>
<b>CONVOYROUTERESTRICTIONS</b> <b>Data Type:</b> String <b>Specific Values:</b> "StayOnRoad", "RespectLanes", "PreferRoads", "Unrestricted"	Road Usage	Autopilot - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual) <a href="#">Convoy Order (on page 105)</a> <a href="#">Tactical Move Order (on page 90)</a> UGV Control (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual)
<b>CONVOYSPACING</b> <b>Data Type:</b> Number	Vehicle Spacing (m)	<a href="#">Convoy Order (on page 105)</a> UGV Control (see AV Assign Waypoints in the VBS4 Trainee Manual)

Parameter (Data Type and Specific Values)	VBS4 Name / Description	Order Behaviors
<b>CONVOY_SPEED</b> <b>Data Type:</b> Number	Movement Speed (km/h)	Autopilot - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual) <a href="#">Convoy Order (on page 105)</a> <a href="#">Tactical Move Order (on page 90)</a> UGV Control (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual)
<b>CREW_ROLE_CATEGORY</b> <b>Data Type:</b> String <b>Specific Values:</b> "All", "Crew", "Cargo"	Crew Role Category	Dismount Order (on page 86) Mount Order (on page 82)
<b>CYCLE</b> <b>Data Type:</b> Object	Create Cycle	All
<b>DEPLOYDSB_BRIDGE_LENGTH</b> <b>Data Type:</b> Object <b>Specific Values:</b> "L_46", "L_40", "L_34", "L_28", "L_22"	Bridge Length	<a href="#">Deploy DSB Order (on page 116)</a>
<b>DESTINATION_ALTITUDE_MODE</b> <b>Data Type:</b> String <b>Specific Values:</b> "AGL", "ASL"	Altitude Mode	<a href="#">Fly Order (on page 124)</a> <a href="#">Loiter Order (on page 129)</a>
<b>DESTINATION_ALTITUDE</b> <b>Data Type:</b> Number	Destination Altitude (m)	<a href="#">Fly Order (on page 124)</a> <a href="#">Loiter Order (on page 129)</a>
<b>ENTITY_TO_FOLLOW</b> <b>Data Type:</b> Object	Follow Entity	Autopilot - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual) <a href="#">Convoy Order (on page 105)</a> <a href="#">Tactical Move Order (on page 90)</a> UGV Control (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual)
<b>FLIGHT_SPEED_LIMIT</b> <b>Data Type:</b> Number	Maximum Speed (km/h)	<a href="#">Fly Order (on page 124)</a> <a href="#">Loiter Order (on page 129)</a>

Parameter (Data Type and Specific Values)	VBS4 Name / Description	Order Behaviors
<b>INDIVIDUAL_REACTION_DIRECT_FIRE</b> <b>Data Type:</b> String <b>Specific Values:</b> "openFire", "holdFire"	On Direct Fire	Individual - Fire At Order (on page 47) Individual - Move Order (on page 50)
<b>INDIVIDUAL_REACTION_SPOTTED</b> <b>Data Type:</b> String <b>Specific Values:</b> "openFire", "holdFire"	On Enemy Spotted	Individual - Fire At Order (on page 47) Individual - Move Order (on page 50)
<b>INDIVIDUAL_SPEED</b> <b>Data Type:</b> String <b>Specific Values:</b> "slowWalk", "walk", "run"	Speed	Individual - Fire At Order (on page 47) Individual - Move Order (on page 50)
<b>INDIVIDUAL_STANCE</b> <b>Data Type:</b> String <b>Specific Values:</b> "standing", "crouched", "prone"	Stance	Individual - Fire At Order (on page 47) Individual - Move Order (on page 50) Tactical Move Order (on page 90)
<b>INDIVIDUAL_TARGET_SQF</b> <b>Data Type:</b> String	Target (SQF Code)	Individual - Fire At Order (on page 47)
<b>INDIVIDUAL_WAIT</b> <b>Data Type:</b> Number	Wait After Finishing (s)	Individual - Fire At Order (on page 47) Individual - Move Order (on page 50)
<b>INDIVIDUAL_WEAPON_BURST_DELAY_BETWEEN_BURSTS</b> <b>Data Type:</b> Number	Wait Between Bursts (s)	Individual - Fire At Order (on page 47)
<b>INDIVIDUAL_WEAPON_BURST_NUMBER_OF_BURSTS</b> <b>Data Type:</b> Number	Number of Bursts	Individual - Fire At Order (on page 47)
<b>INDIVIDUAL_WEAPON_BURST_ROUNDS_PER_BURST</b> <b>Data Type:</b> Number	Rounds per Burst	Individual - Fire At Order (on page 47)
<b>INDIVIDUAL_WEAPON_CLASS</b> <b>Data Type:</b> String <b>Specific Values:</b> "primary", "secondary"	Weapon	Individual - Fire At Order (on page 47)
<b>INDIVIDUAL_WEAPON_FIRE_RATE</b> <b>Data Type:</b> String <b>Specific Values:</b> "single", "burst"	Firing Mode	Individual - Fire At Order (on page 47)

Parameter (Data Type and Specific Values)	VBS4 Name / Description	Order Behaviors
<code>INDIVIDUAL_WEAPON_SINGLE_DELAY_BETWEEN_ROUNDS</code> <b>Data Type:</b> Number	Wait Between Rounds (s)	<a href="#">Individual - Fire At Order (on page 47)</a>
<code>INDIVIDUAL_WEAPON_SINGLE_NUMBER_OF_ROUNDS</code> <b>Data Type:</b> Number	Rounds	<a href="#">Individual - Fire At Order (on page 47)</a>
<code>INDIVIDUAL_WEAPON_STANCE</code> <b>Data Type:</b> String <b>Specific Values:</b> "lowered", "raised"	Weapon Stance	<a href="#">Individual - Fire At Order (on page 47)</a> <a href="#">Individual - Move Order (on page 50)</a>
<code>LOITER_RADIUS</code> <b>Data Type:</b> Number	Loiter Radius	<a href="#">Animal Herd Movement (see Animal AI (on page 202))</a>
<code>LOITER_TIMEOUT</code> <b>Data Type:</b> Number	Loiter Timeout	<a href="#">Animal Herd Movement (see Animal AI (on page 202))</a>
<code>SIZE_X</code> <b>Data Type:</b> Number	Radius	<a href="#">Suppress Order (on page 77)</a>
<code>SIZE_Y</code> <b>Data Type:</b> Number	Size (Up-Down)	<a href="#">Suppress Order (on page 77)</a>
<code>SYNCED</code> <b>Data Type:</b> Object	Sync to Waypoint	All
<code>TACTICAL_MOVEREACTION_DISTANCE</code> <b>Data Type:</b> Number	Reaction Distance	<a href="#">Tactical Move Order (on page 90)</a>
<code>TARGET_GROUP</code> <b>Data Type:</b> Group	Select Target	<a href="#">Pursue Order (on page 74)</a>
<code>TARGET_VEHICLE</code> <b>Data Type:</b> Object	Select Target	<a href="#">Mount Order (on page 82)</a>
<code>UGV_AVOID_THREATS</code> <b>Data Type:</b> Boolean	Avoid Threats	Autopilot - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual)

Parameter (Data Type and Specific Values)	VBS4 Name / Description	Order Behaviors
<b>UGV_DEFEND</b> <b>Data Type:</b> Boolean	Defend on Completion	Autopilot - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual)
<b>UGV_DELAY</b> <b>Data Type:</b> Number	Delay (s)	Autopilot - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual)
<b>UGV_ENGAGE_ENEMIES</b> <b>Data Type:</b> String <b>Specific Values:</b> "openFire", "holdFire", "returnFire"	Weapon Control Status	Autopilot - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual)
<b>UGV_HALT_ON_DETECT</b> <b>Data Type:</b> Boolean	Halt on Detect	Autopilot - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual)
<b>UGV_WP_REPORT</b> <b>Data Type:</b> Boolean	Send Waypoint Complete Report	Autopilot - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control (see AV Assign Waypoints in the VBS4 Trainee Manual) UGV Control - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual)
<b>VARNAME</b> <b>Data Type:</b> String	Variable Name	All

Parameter (Data Type and Specific Values)	VBS4 Name / Description	Order Behaviors
<p><b>WEAPON_CONTROL_STATUS</b></p> <p><b>Data Type:</b> String</p> <p><b>Specific Values:</b> "noChange", "weaponsFree", "holdFire"</p>	Weapon Control Status	<a href="#">Advance Order (on page 72)</a> <a href="#">Assault Order (on page 70)</a> <a href="#">Defend Order (on page 80)</a> <a href="#">Dismount Order (on page 86)</a> <a href="#">Fly Order (on page 124)</a> <a href="#">Land Order (on page 127)</a> <a href="#">Loiter Order (on page 129)</a> <a href="#">Mount Order (on page 82)</a> <a href="#">Pursue Order (on page 74)</a> <a href="#">Riot (see Define Civilian Riot (on page 148))</a> <a href="#">Suppress Order (on page 77)</a> <a href="#">Tactical Move Order (on page 90)</a>

## Order Behavior Class Names

The following Order behavior class names are available:

Class Name	Order Behavior
Advance	<a href="#">Advance Order (on page 72)</a>
Assault	<a href="#">Assault Order (on page 70)</a>
autopilotDefend	Autopilot - Defend (see AV Assign Waypoints in the VBS4 Trainee Manual)
convoy	<a href="#">Convoy Order (on page 105)</a>
Defend	<a href="#">Defend Order (on page 80)</a>
Fly	<a href="#">Fly Order (on page 124)</a>
Land	<a href="#">Land Order (on page 127)</a>
Loiter	<a href="#">Loiter Order (on page 129)</a>
Riot	Riot (see Define Civilian Riot (on page 148))
Suppress	<a href="#">Suppress Order (on page 77)</a>
TacticalMove	<a href="#">Tactical Move Order (on page 90)</a>
Team_Dismount	<a href="#">Dismount Order (on page 86)</a>
Team_Mount	<a href="#">Mount Order (on page 82)</a>

Class Name	Order Behavior
Team_Pursue	Pursue Order (on page 74)
VBS_Custom	Custom (see <a href="#">Custom Behaviors (on page 20)</a> )
vbsCon_animal	Animal Herd Movement (see <a href="#">Animal AI (on page 202)</a> )
deployDSB	Deploy DSB Order (on page 116)
individual_fire	Individual - Fire At Order (on page 47)
individual_move	Individual - Move Order (on page 50)
ReturnToFormation	Return to Formation Order (on page 89)
umvDefend	UGV Control - Defend (see <a href="#">AV Assign Waypoints in the VBS4 Trainee Manual</a> )
moveAggressive	UGV Control (see <a href="#">AV Assign Waypoints in the VBS4 Trainee Manual</a> )



## EXAMPLE

Creating waypoints:

```
_firstWaypoint = [group player, "Advance", getPosASL2 player
vectorAdd [0,100,0]] call fn_vbsCon_waypointCreate;
_secondWaypoint = [_firstWaypoint, "Advance", getPosASL2 player vectorAdd
[0,200,0]] call fn_vbsCon_waypointCreate;
```

Modifying an existing waypoint:

```
[group player, ["VARNAME", "wp1", "WEAPON_CONTROL_STATUS",
"HoldFire"]] call fn_vbsCon_waypointUpdate;
```

Assigning an existing waypoint:

```
[waypoint1, group player] call fn_vbsCon_waypointAssign;
```

## 4. AI on Rails

You can specify the behavior of a Control AI entity by using the [\(F3\) Waypoints Editor Object \(on page 18\)](#), to give the entity a sequence of orders to perform. These orders can be as simple as movement from point A to point B with no autonomy, or composed together and combined with triggers, to create complex, branching plans.

The following sections describe how to build a progressively more complex behavior:

- [Individual Move \(on the next page\)](#) - Add a sequence of move orders to a single soldier.
- [Individual Fire At \(on page 39\)](#) - Add a sequence of move-and-fire orders to a single soldier.
- [Delayed Order Execution \(on page 42\)](#) - Use triggers to control when further orders should be executed.
- [Branching Orders \(on page 44\)](#) - Allow entities to choose from several orders to execute.
- [AI on Rails Orders \(on page 46\)](#) - AI on Rails orders that can be used. Each order behavior is applied to an entity using the [\(F3\) Waypoints Editor Object \(on page 18\)](#).

For an AI on Rails example mission, see [Example Content \(on page 230\)](#).

## 4.1 Individual Move

You can create and link a move order to an ungrouped entity, for the entity to perform.

The green soldier proxy, displayed in the 3D Editor (Prepare / Execute Mode), reflects the target entity stance, position, and orientation.

### Follow these steps:

1. In the Editor (Prepare Mode), place a single, ungrouped Control AI entity on the map.
2. Shift-click the entity, and then click a position on the map (see [Waypoints \(on page 23\)](#)).
3. In the **Behavior** list, select **Individual - Move**.
4. Set the Order properties (see [Individual - Move Order \(on page 50\)](#)) and click **OK**.
5. Preview the mission.

The entity assumes the specified stance and moves to the destination.

**Image-7: An Individual - Move Order, at the end of which the soldier entity crouches**



You can link multiple move orders by linking multiple [Individual - Move Order \(on page 50\)](#) Editor Objects (see [Waypoints \(on page 23\)](#)). The entity performs these orders in sequence.

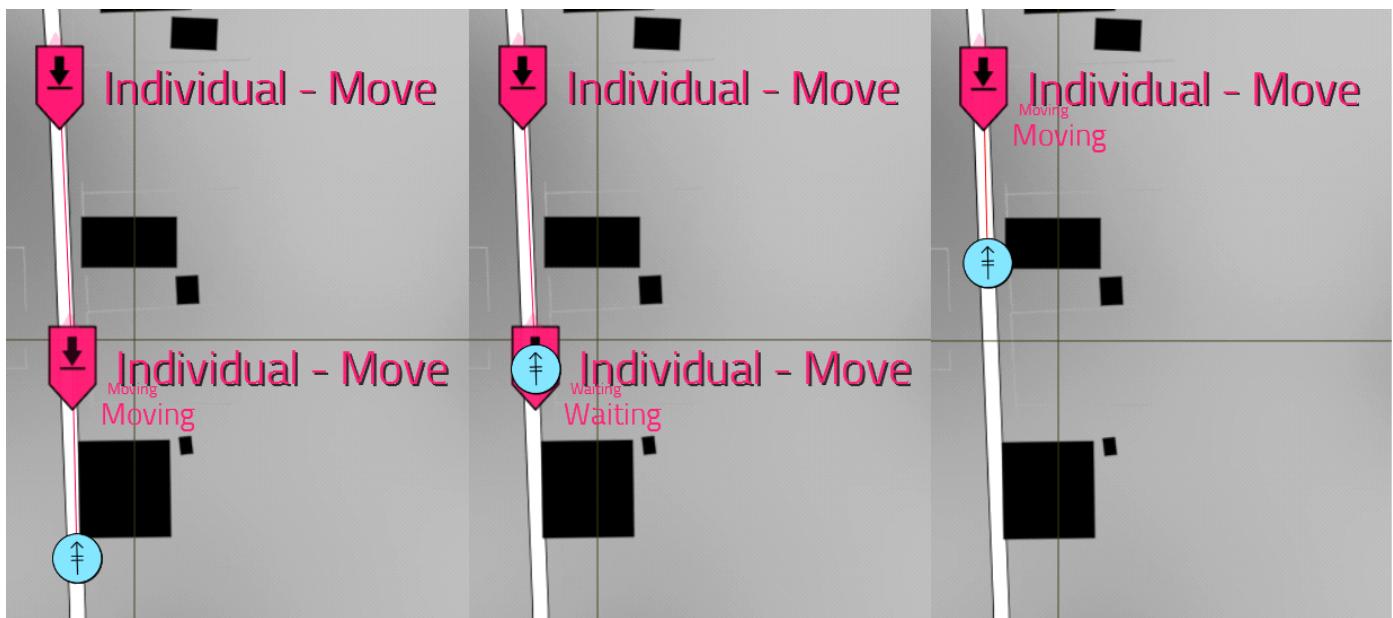
You can also set the entity to wait at its destination for a specified amount of time, before continuing with the next order.

**Follow these steps:**

1. Create an [Individual - Move Order \(on page 50\)](#), based on the previous procedure.
2. In the Order properties, enter the number of seconds for the entity to wait, before continuing with to the next order, in the **Wait After Finishing (s)** field.
3. Shift-click the order, and then click a position on the map for the next order.
4. Set the Order properties, based on the previous procedure.
5. Preview the mission.

The entity moves to the first destination, waits for the specified amount of time, and then continues moving to the second destination.

**Image-8: From left to right: the entity moves to the first waypoint, waits, and continues to the second waypoint**



## 4.2 Individual Fire At

You can create and link a move-and-fire order to an ungrouped entity, for the entity to perform. The green soldier proxy, displayed in the 3D Editor (Prepare / Execute Mode), reflects the entity stance, position, and orientation.

The entity first performs the move (similarly to [Individual Move \(on page 37\)](#)), and then fires at the specified target (the entity automatically reloads). You can fire at either an object, such as an enemy entity, or an ASL2 position on the terrain.

This section demonstrates an example, where a soldier performs the following order sequence:

1. Moves to the first position and fires at a falling target.
2. Moves to the next position and fires at an ASL2 position on the terrain.

### Follow these steps:

1. In the Editor (Prepare Mode), place a single, ungrouped Control AI entity on the map.
2. Place two falling-target objects (any object from [\(F8\) Objects > Targets - Falling](#)) on the map and give them names (for example, `ftarget1` and `ftarget2`).
3. Shift-click the entity, and then click a position on the map (see [Waypoints \(on page 23\)](#)).
4. In the **Behavior** list, select **Individual - Fire At**.
5. In **Target (SQF Code)**, specify the name of the first falling-target object (for example, `ftarget1`).

#### NOTE

It can be any SQF code that returns an Entity or Position (in ASL2 format). The SQF script is evaluated **only after** the move is performed. Therefore, the entity fires at the target **only after** it performs the move. If the target is moving, only the target position at the time of the SQF code evaluation is taken into account.

6. In **Weapon**, select the weapon the entity should use (based on the weapons the entity has) to fire at the target. In **Firing Mode**, select the firing mode, which can be either **Single** (single rounds) or **Burst** (burst rounds). In both firing modes, you can set the number of rounds / bursts to fire in **Rounds**, as well as the waiting period between each round / burst in **Wait Between Rounds (s)**.
7. Set the other Order properties (see [Individual - Fire At Order \(on page 47\)](#)) and click **OK**.
8. Shift-click the Fire At Order, and then click a position on the map, for the next move-and-fire waypoint.
9. In **Target (SQF Code)**, specify the ASL2 position of the second falling-target object (for example, using the SQF script: `ftarget2 modelToWorldASL2 [0,0,1]` - this fires 1m above the ASL2 position of the target, which would be on the ground).

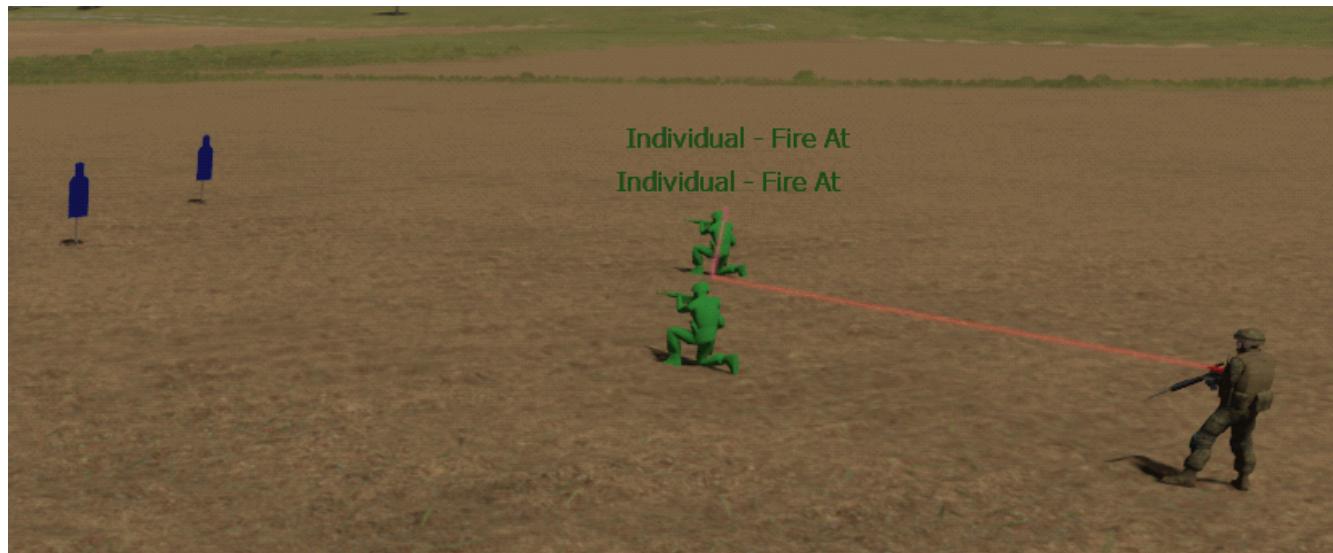
10. Set the other Order properties and click **OK**.

11. Preview the mission.

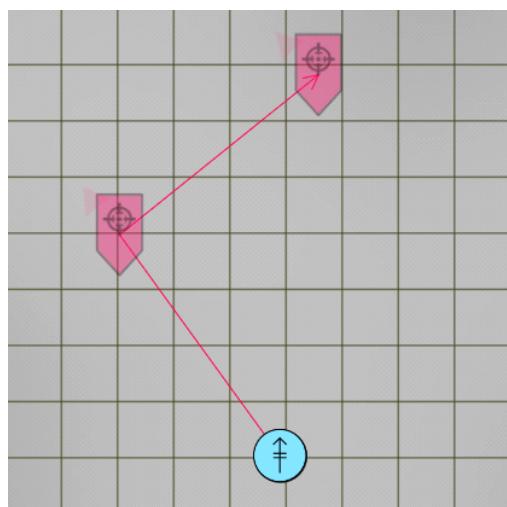
The entity does the following:

1. Moves to the first firing position.
2. Fires at the first target and the target falls down.
3. The entity then moves to the second firing position.
4. Fires 1m above the ASL2 ground position of the second target, which results in hitting the target.

**Image-9: Fire At Order settings and the finished example scenario**



**Image-10: Fire At Orders, connected as waypoints**



**Image-11: The entity aims and fires at a falling target****Considerations:**

- If the entity has a weapon, but no ammunition (which is indicated in the 2D Editor (Execute Mode)), it waits for the administrator to edit the entity loadout, to provide it with usable ammunition.
- If the entity has no weapon of the selected type, it waits at its firing position, until the administrator edits the entity loadout, to provide it with a weapon of this type.
- The entity cannot fire at dead targets (for example, if the target is killed by the first shot, the entity does not fire the remaining rounds).
- Setting the target to `enemySoldier` is not the same as `getPosASL2 enemySoldier`. The former aims at the chest of the target and considers target velocity, while the latter aims at a static position near the feet of the target / ground.
- Aiming can take some time, if the target is too far. If the waiting period between rounds / bursts is set to 0, re-aiming between rounds / bursts can still take more than 0 seconds.
- If the given target is too far, the entity waits at its firing position.

## 4.3 Delayed Order Execution

You can control when an order is executed by linking a trigger to it.

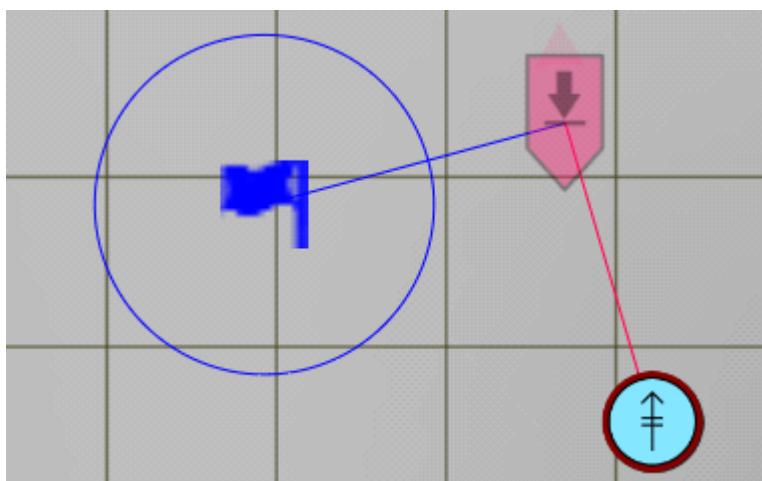
When a Control AI entity finishes an order, and the next order in the sequence has a linked trigger, the execution of the next order is delayed, until the trigger is activated. Similarly, if a trigger is linked to an existing order, and that order is linked to a Control AI entity, it does not start executing the order, until the trigger is activated.

**Follow these steps:**

1. Set up an **Individual - Move Order**, as described in [Individual Move \(on page 37\)](#).
2. Create a trigger, and set its **Activation** field to **Radio Alpha**.
3. Link the order to the trigger.
4. Link the order to an entity.
5. Preview the mission.

The entity does not begin executing the order, until the radio trigger is activated.

**Image-12: An Individual - Move Order linked to an entity and a trigger**



You can link multiple triggers to a single order. The order is then delayed, until any of the triggers is activated. This way, you can specify multiple activation conditions for an order. For example, a Control AI entity can be set to crest a ridge, when either a BLUFOR soldier enters a trigger area, or when a radio trigger is manually activated by an administrator.

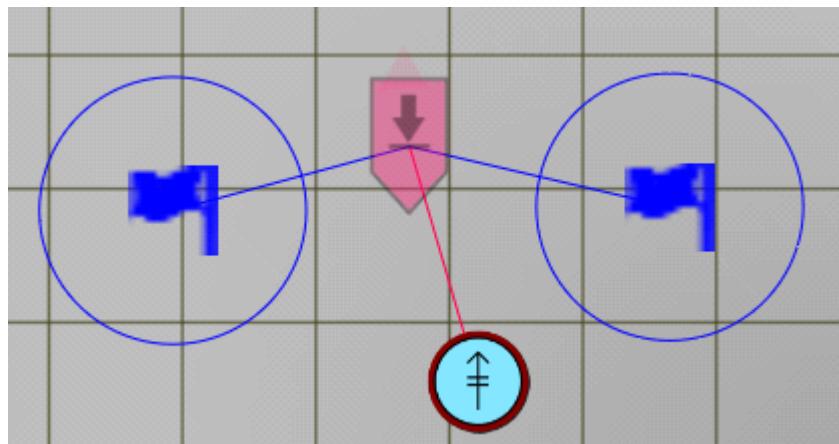
**Follow these steps:**

1. Set up an **Individual - Move Order** with a linked radio trigger, as described in the previous procedure.
2. Create a second trigger, and set its **Activation** field to **BLUFOR**, and its **Activation Type** field to **Present**.

3. Link the order to the second trigger.

4. Preview the mission.

**Image-13: An Individual Move - Order linked to an entity and two triggers**



## 4.4 Branching Orders

You can have an entity select one of several orders to execute.

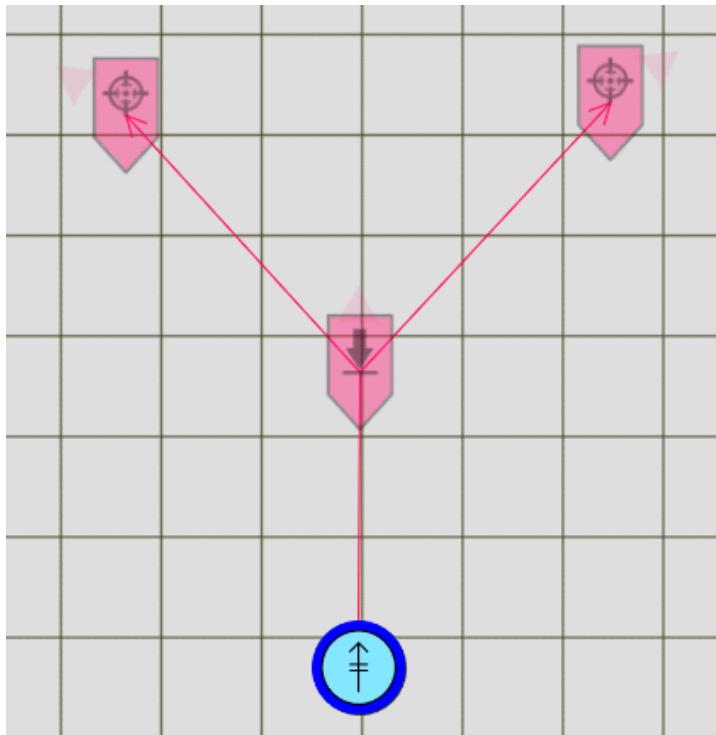
In the most basic case, when the entity finishes an order and there are subsequent orders to select from, the selection is done randomly.

**Follow these steps:**

1. Set up an **Individual - Move Order / Individual - Fire At Order**, as described in [Individual Move \(on page 37\)](#) / [Individual Fire At \(on page 39\)](#). Make sure the order is linked to an entity.
2. Create two additional Individual Orders and link them to the order created in step 1.
3. Preview the mission.

The entity executes the first order, and then randomly chooses one of the two subsequent orders.

**Image-14: Branching orders example**



You can also use branching orders together with triggers, to have an entity select an order based on scenario conditions or manual trigger activation, as described in the example below:

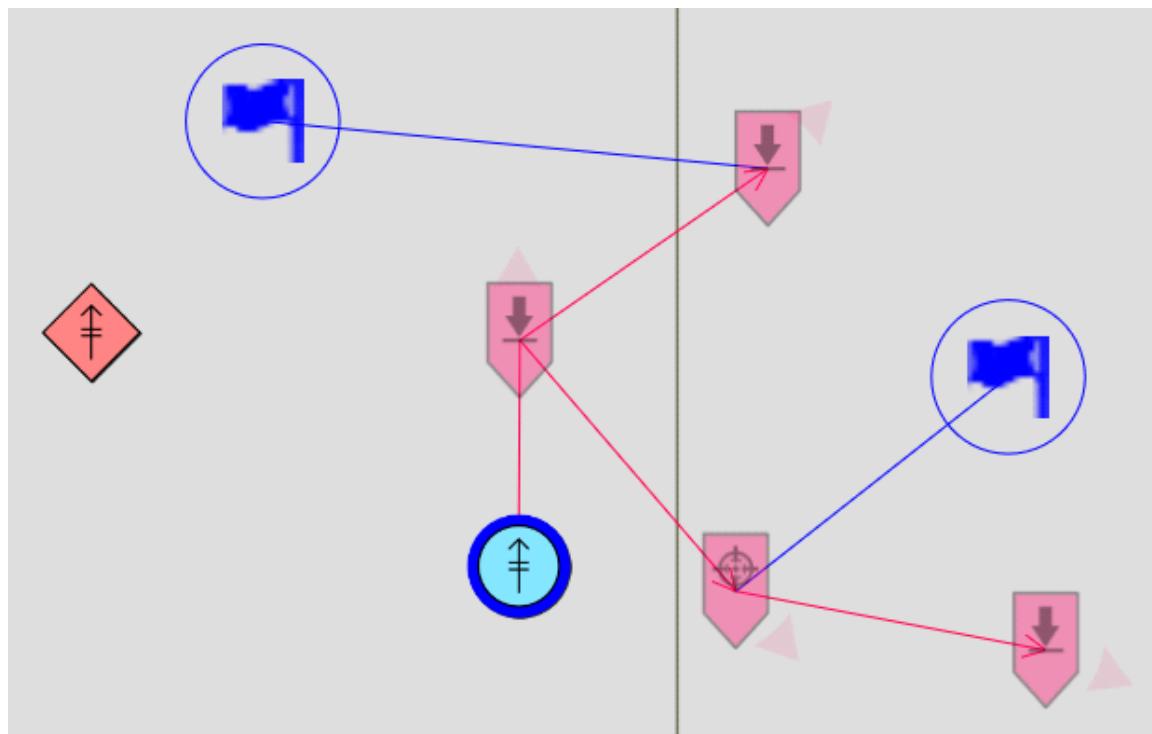
**Follow these steps:**

1. Set up an **Individual - Move Order**. In its properties, set all of its reactions to **Hold Fire**.
2. Create a trigger, and set its **Activation** field to **Radio Alpha**.
3. Create an **Individual - Move Order**. In its properties, set all of its reactions to **Hold Fire**. Link this order to the trigger created in step 2, and then link it to the order created in step 1.

4. Create a second trigger, set its **Activation** field to **OPFOR** and its name to "**Target\_Zone**" (without the quotation marks). Adjust its size as necessary.
5. Create an **Individual - Fire At Order**. In its properties, set all of its reactions to **Hold Fire**, and set the **Target (SQF Code)** field to "**(list Target\_Zone) select 0**" (without the quotation marks). Link this order to the trigger created in step 4, and then link it to the order created in step 1.
6. Create an **Individual - Move Order**. Link this order to the order created in step 5.
7. Link the order created in step 1 to a **BLUFOR** entity.
8. Create an **OPFOR** entity.
9. Preview the mission.

The BLUFOR entity moves to the first order. It waits there, until the OPFOR entity is moved into the trigger area created in step 4. Then it engages that entity and proceeds on. However, if the radio trigger is manually activated by an administrator before the OPFOR entity enters the trigger area, the entity moves away immediately.

**Image-15: Branching orders with triggers example**



## 4.5 AI on Rails Orders

AI on Rails uses the [\(F3\) Waypoints Editor Object \(on page 18\)](#) and consists of the following orders:

 **NOTE**

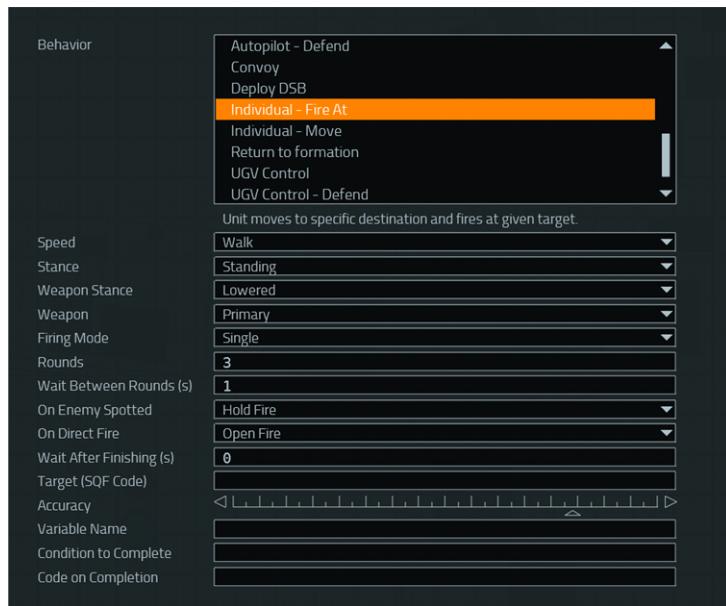
Each order can only be assigned to individual, ungrouped entities.

Order Type	Description
Individual - Fire At Order (on the next page)	The entity assumes the specified stance and speed, moves to target position, orients itself according to the specified heading, and fires at the specified target.
Individual - Move Order (on page 50)	The entity assumes the specified stance and speed, moves to target position, and orients itself according to the specified heading.

## 4.5.1 Individual - Fire At Order

Assigns a waypoint to an individual entity to fire at the given target (the entity automatically reloads). The speed, stance, reactions, and fire target of the entity can be configured. The position is the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map.

**Image-16: Individual Fire At - Order settings**



### Follow these steps:

1. Select the **(F3) Waypoints** Editor Object from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Individual - Fire At**.
3. Set the [Individual Fire At Settings \(below\)](#).
4. Set the [Waypoint Completion Settings \(on the next page\)](#).
5. Click **OK** to confirm.

The Individual - Fire At Order behavior is set up.

### Individual Fire At Settings

Setting	Description
Speed	Desired move speed of the entity. <ul style="list-style-type: none"><li>• Slow Walk</li><li>• Walk</li><li>• Run</li></ul>

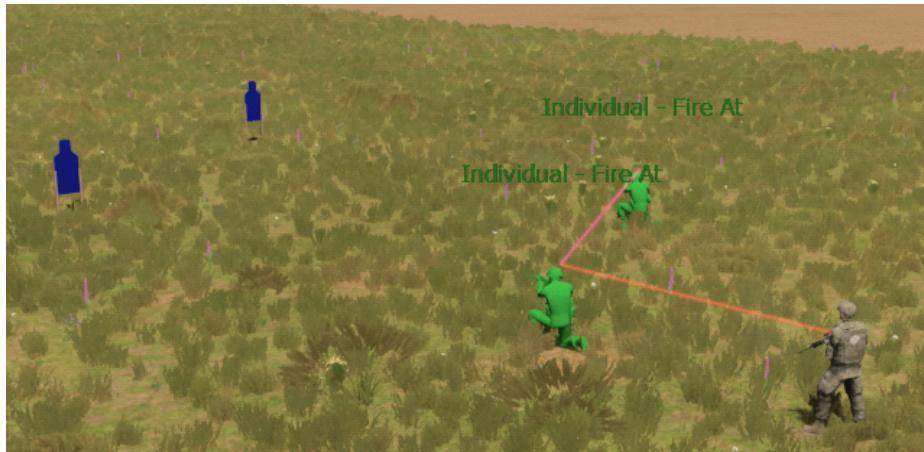
Setting	Description
<b>Stance</b>	Desired stance to assume during the move. <ul style="list-style-type: none"> <li><b>Standing</b> - Entity is upright and walks / runs to the destination.</li> <li><b>Crouched</b> - Entity is crouched and moves in a lowered stance to the destination.</li> <li><b>Prone</b> - Entity lays down and crawls to the destination.</li> </ul>
<b>Weapon Stance</b>	Defines how the weapon should be carried during the move. <ul style="list-style-type: none"> <li><b>Lowered</b> - Weapon is lowered.</li> <li><b>Raised</b> - Weapon is aimed and ready to fire.</li> </ul>
<b>Weapon</b>	The weapon the entity should use to fire at the target. <ul style="list-style-type: none"> <li><b>Primary</b> - The primary weapon of the entity.</li> <li><b>Pistol Weapon</b> - A pistol.</li> </ul>
<b>Firing Mode</b>	The firing mode. <ul style="list-style-type: none"> <li><b>Single</b> - Single rounds.</li> <li><b>Burst</b> - Burst rounds.</li> </ul>
<b>Rounds</b>	The number of rounds to fire at the target.
<b>Wait Between Rounds (s)</b>	The number of seconds to wait between each round / burst.
<b>On Enemy Spotted</b>	Controls how the entity reacts, when spotting the enemy. <ul style="list-style-type: none"> <li><b>Open Fire</b> - Entity stops and opens fire on any visible threats.</li> <li><b>Hold Fire</b> - Entity ignores visible threats.</li> </ul>
<b>On Direct Fire</b>	Controls how the entity reacts to incoming enemy fire (defined by shots impacting or passing around the entity at a short distance). <ul style="list-style-type: none"> <li><b>Open Fire</b> - Entity stops and opens fire on any visible threats.</li> <li><b>Hold Fire</b> - Entity ignores incoming fire.</li> </ul>
<b>Wait After Finishing (s)</b>	Controls how long (in seconds) the entity holds its target position after finishing the order, before continuing to execute further orders.
<b>Target (SQF Code)</b>	Target to fire at. Can be any SQF code that returns an Entity or Position (in ASL2 format).
<b>Accuracy</b>	Firing accuracy of the entity.
<b>Variable Name</b>	In <b>Variable Name</b> , enter the Order waypoint name, which can be used in SQF scripts.

## Waypoint Completion Settings

Set the Order waypoint completion settings:

Option	Description
<b>Condition to Complete</b>	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
<b>Code on Completion</b>	SQF code to execute on waypoint completion.

**Image-17: Individual - Fire At example in the 3D Editor (Prepare Mode)**



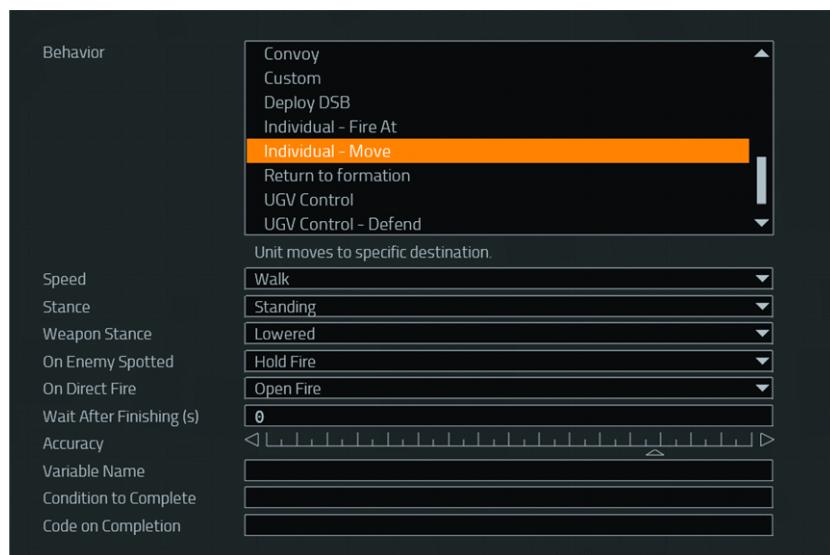
### Considerations:

- If the entity has a weapon, but no ammunition (which is indicated in the 2D Editor (Execute Mode)), it waits for the administrator to edit the entity loadout, to provide it with usable ammunition.
- If the entity has no weapon of the selected type, it waits at its firing position, until the administrator edits the entity loadout, to provide it with a weapon of this type.
- The entity cannot fire at dead targets (for example, if the target is killed by the first shot, the entity does not fire the remaining rounds).
- Setting the target to `enemySoldier` is not the same as `getPosASL2 enemySoldier`. The former aims at the chest of the target and considers target velocity, while the latter aims at a static position near the feet of the target / ground.
- Aiming can take some time, if the target is too far. If the waiting period between rounds / bursts is set to 0, re-aiming between rounds / bursts can still take more than 0 seconds.
- If the given target is too far, the entity waits at its firing position.

## 4.5.2 Individual - Move Order

Assigns a waypoint to an individual entity. The speed, stance, and reactions of the entity can be configured. The position is the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map.

**Image-18: Individual Move - Order settings**



### Follow these steps:

1. Select the [\(F3\) Waypoints](#) Editor Object from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Individual - Move**.
3. Set the [Individual Move Settings](#) (below).
4. Set the [Waypoint Completion Settings](#) (on the next page).
5. Click **OK** to confirm.

The Individual - Move Order behavior is set up.

### Individual Move Settings

Setting	Description
<b>Speed</b>	Desired move speed of the entity. <ul style="list-style-type: none"><li>• <b>Slow Walk</b></li><li>• <b>Walk</b></li><li>• <b>Run</b></li></ul>

Setting	Description
<b>Stance</b>	Desired stance to assume during the move. <ul style="list-style-type: none"> <li><b>Standing</b> - Entity is upright and walks / runs to the destination.</li> <li><b>Crouched</b> - Entity is crouched and moves in a lowered stance to the destination.</li> <li><b>Prone</b> - Entity lays down and crawls to the destination.</li> </ul>
<b>Weapon Stance</b>	Defines how the weapon should be carried during the move. <ul style="list-style-type: none"> <li><b>Lowered</b> - Weapon is lowered.</li> <li><b>Raised</b> - Weapon is aimed and ready to fire.</li> </ul>
<b>On Enemy Spotted</b>	Controls how the entity reacts, when spotting the enemy. <ul style="list-style-type: none"> <li><b>Open Fire</b> - Entity stops and opens fire on any visible threats.</li> <li><b>Hold Fire</b> - Entity ignores visible threats.</li> </ul>
<b>On Direct Fire</b>	Controls how the entity reacts to incoming enemy fire (defined by shots impacting or passing around the entity at a short distance). <ul style="list-style-type: none"> <li><b>Open Fire</b> - Entity stops and opens fire on any visible threats.</li> <li><b>Hold Fire</b> - Entity ignores incoming fire.</li> </ul>
<b>Wait After Finishing (s)</b>	Controls how long (in seconds) the entity holds its target position after finishing the order, before continuing to execute further orders.
<b>Accuracy</b>	Firing accuracy of the entity.
<b>Variable Name</b>	In <b>Variable Name</b> , enter the Order waypoint name, which can be used in SQF scripts.

## Waypoint Completion Settings

Set the Order waypoint completion settings:

Option	Description
<b>Condition to Complete</b>	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
<b>Code on Completion</b>	SQF code to execute on waypoint completion.

**Image-19: Individual - Move example in the 3D Editor (Prepare Mode)**

## 5. Military AI

You can give Control AI general-purpose military infantry and vehicle groups orders to follow.

For vehicles, also see the specific [Convoy AI \(on page 93\)](#) and [Bridge Laying Convoy AI \(on page 112\)](#).

Each order behavior is applied to an infantry and / or vehicle group, using the [\(F3\) Waypoints Editor Object \(on page 18\)](#). The orders create VBS4 waypoints (see [Waypoints \(on page 23\)](#)), and can use branching (see [Branching Waypoints \(on page 25\)](#)).

The following topics are discussed:

- [Infantry Tutorial \(on the next page\)](#) - A tutorial that demonstrates the Control AI military behaviors for infantry.
- [Vehicle Tutorial \(on page 58\)](#) - A tutorial that demonstrates the Control AI military behaviors for vehicles.
- [Interoperability Tutorial \(on page 62\)](#) - A tutorial that demonstrates running group and sub-group Control AI behaviors in parallel.
- [Military AI Orders and Behaviors \(on page 67\)](#) - A list of general-purpose infantry and vehicle AI order behaviors.

## 5.1 Infantry Tutorial

This tutorial demonstrates the use of Control AI military behaviors for infantry.

The workflow of the tutorial is as follows:

1. In Prepare mode, create infantry groups A and B, using the ORBAT Editor - [Creating the Infantry Groups \(below\)](#).
2. Give group A an order to advance to a given location - [Creating an Advance Order \(below\)](#).
3. After group A performs the advance, give it an assault order to assault the enemy forces at the given location - [Creating an Assault Order \(on the next page\)](#).
4. Give group B a suppress order to provide fire support to group A - [Creating a Suppress Order \(on page 56\)](#).

### 5.1.1 Creating the Infantry Groups

First, create two infantry groups - group A and group B.

**Follow these steps:**

1. Open the VBS Editor in Prepare mode, and select (**F9**) ORBAT in the Editor Objects List.
2. Double-click a location on the map to place a US squad, referred to as group A.  
The ORBAT Editor (see in the VBS4 Editor Manual) opens in the Place ORBAT mode.
3. In the **Affiliation** drop-down, select **US**.
4. In the ORBAT Tree, select **Ground Unit > Infantry > Squad**.
5. Click **Place**.  
The US squad is placed on the map.
6. Repeat steps 1 - 5 to place a second US squad, referred to as group B.

Groups A and B are placed on the map.

### 5.1.2 Creating an Advance Order

Give group A an [Advance Order \(on page 72\)](#) to move to a certain position. A group that performs an Advance Order prefers movement speed over engaging and eliminating the enemy.

**Follow these steps:**

1. Right-click group A and select **Orders > Assign New Waypoint**.
2. Click a location on the map where you want group A to advance to.  
The Object Properties dialog opens.

3. In the **Behavior** list, select **Advance**.

4. Click **OK**.

Group A is given an Advance Order waypoint.

5. Click **Preview** to preview the scenario.

Group A moves to the Advance Order waypoint position in a column of fireteams, and after reaching its destination, its members take cover.



### 5.1.3 Creating an Assault Order

If there are enemy forces expected near the location to which group A advances, an [Assault Order](#) (on page 70) is needed as well. The Advance Order is mainly focused on moving fast in a certain direction, whereas an Assault Order allows the group to perform basic bounding, so that while one part of the group advances, another part stays covered and surveys the area.

**Follow these steps:**

1. Right-click the Advance Order waypoint and select **Assign New Waypoint**.

2. Click a location on the map where you want the assault to happen.

The Object Properties dialog opens.

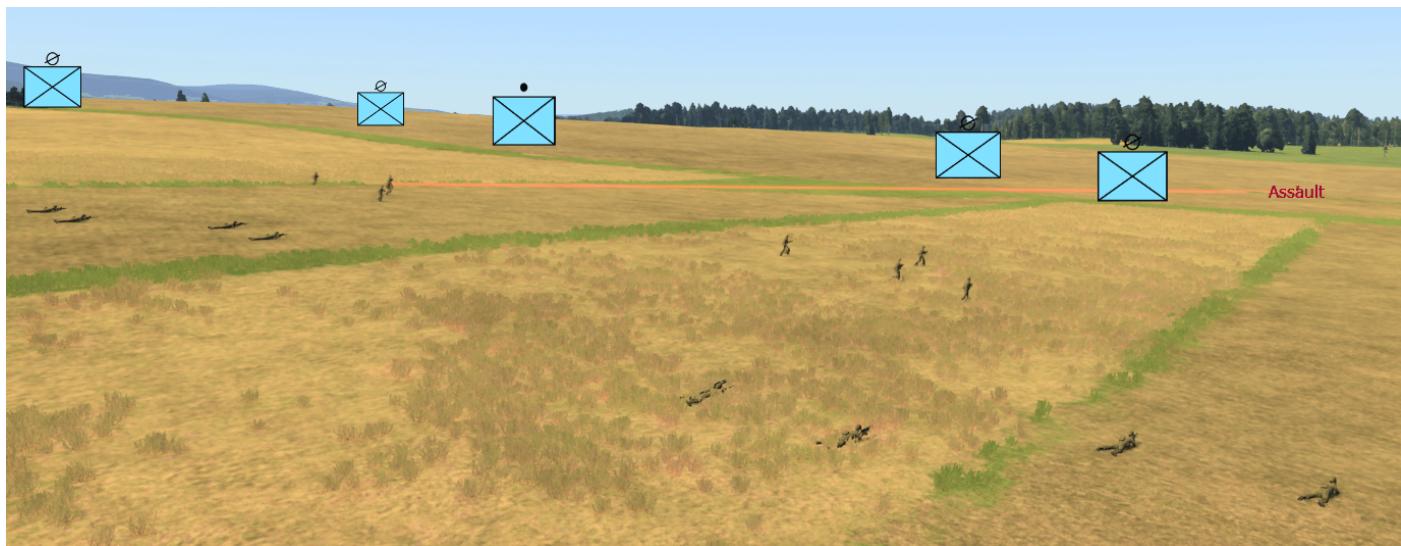
3. In the **Behavior** list, select **Assault**.

4. Click **OK**.

The Advance Order waypoint is followed up by an Assault Order waypoint.

##### 5. Click **Preview** to preview the scenario.

After finishing the Advance Order, group A starts bounding to the Assault Order waypoint position, where one part of the group advances, while another part stays covered and surveys the area.



#### 5.1.4 Creating a Suppress Order

While group A advances and then assaults any expected enemy forces, group B can provide fire support to group A, using the [Suppress Order \(on page 77\)](#).

##### Follow these steps:

1. Right-click group B and select **Orders > Assign New Waypoint**.
2. Click a location on the map where you want the fire support / suppression to happen (somewhere close to the Assault Order waypoint position).

The Object Properties dialog opens.

3. In the **Behavior** list, select **Suppress**.
4. In **Radius**, set the fire support radius (in meters).
5. Click **OK**.

Group B is given a Suppress Order waypoint.

6. Click **Preview** to preview the scenario.

Group B forms a firing line from which they start firing at the Suppress Order waypoint position, supporting group A.

If the suppressing group B should suppress from another position, it can be first ordered to move to a certain position (using an Advance Order), followed by a Suppress Order waypoint as the next one in the waypoint chain.

The whole scenario now contains two infantry squad-sized groups - A and B. Group A first quickly moves to the position of the Advance Order waypoint, and then starts bounding to the position of the Assault Order waypoint. Meanwhile, group B provides fire support.

**Image-20: Group B providing fire support to group A**



## 5.2 Vehicle Tutorial

This tutorial demonstrates the use of Control AI military behaviors for vehicles.

The workflow of the tutorial is as follows:

1. In Prepare mode, create a mechanized infantry group, using the ORBAT Editor and VBS Editor - [Creating the Mechanized Infantry Group \(below\)](#).
2. Order the infantry part of the group to mount a vehicle as cargo - [Creating a Mount Order \(on the next page\)](#).
3. Give the vehicle an order to advance to a given location - [Creating an Advance Order \(on page 60\)](#).
4. Order the infantry part of the group to dismount the vehicle - [Creating a Dismount Order \(on page 61\)](#).

### 5.2.1 Creating the Mechanized Infantry Group

First, create a mechanized infantry group.

**Follow these steps:**

1. Open the VBS Editor in Prepare mode, and select **(F9) ORBAT** in the Editor Objects List.
2. Double-click a location on the map to place a US fireteam, to serve as the infantry part of the mechanized infantry group.

The ORBAT Editor (see in the VBS4 Editor Manual) opens in the Place ORBAT mode.

3. In the **Affiliation** drop-down, select **US**.
4. In the ORBAT Tree, select **Ground Unit > Infantry > Fireteam**.
5. Click **Place**.

The US fireteam is placed on the map.

6. In the VBS Editor, in the Editor Objects List, select **(F4) Vehicle**.
7. Double-click a location on the map where you want to place the vehicle to transfer the fireteam as cargo.

The Object Properties dialog opens.

**NOTE**

Place the vehicle about 200m from the fireteam. See the Measure Distance Tool in the VBS4 Editor Manual.

8. In **Filters**, type **Stryker** and select any of the US Strykers that can fit a US fireteam as cargo.

## 9. Click **OK**.

The Stryker vehicle is placed on the map.

The infantry group (fireteam) and vehicle, which constitute the overall mechanized infantry group, are placed on the map.

### 5.2.2 Creating a Mount Order

Order the fireteam to mount the Stryker as cargo, using the [Mount Order \(on page 82\)](#).

#### Follow these steps:

1. Right-click the infantry part of the group and select **Orders > Assign New Waypoint**.
2. Click a location on the map where you want the fireteam to mount the Stryker.

The Object Properties dialog opens.

#### **WARNING**

Make sure that the location is not farther away than 50m from the vehicle, for the vehicle mounting to happen.

3. In the **Behavior** list, select **Mount**.

4. Click **OK**.

The infantry is given an Mount Order waypoint.

5. Click **Preview** to preview the scenario.

The infantry moves to the Mount Order waypoint position and mounts the Stryker as cargo.



## 5.2.3 Creating an Advance Order

Give the Stryker an order to advance to a given location, using the [Advance Order \(on page 72\)](#).

**Follow these steps:**

1. Right-click the Stryker and select **Orders > Assign New Waypoint**.
2. Click a location on the map within 50m from the Mount Order waypoint.

The Object Properties dialog opens.

3. In the **Behavior** list, select **Advance**.
4. Click **OK**.

The Stryker is given an Advance Order waypoint.

5. Right-click the Advance Order waypoint and select **Sync to Waypoint** (see [Waypoint Synchronization \(on page 26\)](#)).
6. Click the Mount Order waypoint.

This synchronizes the Advance Order with the Mount Order, which means that the Stryker moves to the Advance Order waypoint location, and waits for the fireteam to complete the Mount Order.

7. Right-click the Advance Order waypoint and select **Assign New Waypoint**.
8. Click a location on the map where you want the Stryker to advance to, for the fireteam to dismount.

The Object Properties dialog opens.

9. In the **Behavior** list, select **Advance**.
10. Click **OK**.

The Stryker is given another Advance Order waypoint.

11. Click **Preview** to preview the scenario.

The Stryker drives to the pickup location for the fireteam to mount it, and then drives to another location, where the fireteam is supposed to dismount.



## 5.2.4 Creating a Dismount Order

Order the fireteam to dismount the Stryker, using the [Dismount Order \(on page 86\)](#).

**Follow these steps:**

1. Right-click the Mount Order waypoint and select **Assign New Waypoint**.
2. Click a location on the map, next to the final Advance Order waypoint, where you want the fireteam to dismount.

The Object Properties dialog opens.

**NOTE**

The position of the Dismount Order waypoint has no effect on where the dismount occurs, which is controlled by the final Advance Order waypoint. However, visually, it is easier to place these Order waypoints together.

3. In the **Behavior** list, select **Dismount**.

4. Click **OK**.

The Stryker is given a Dismount Order waypoint.

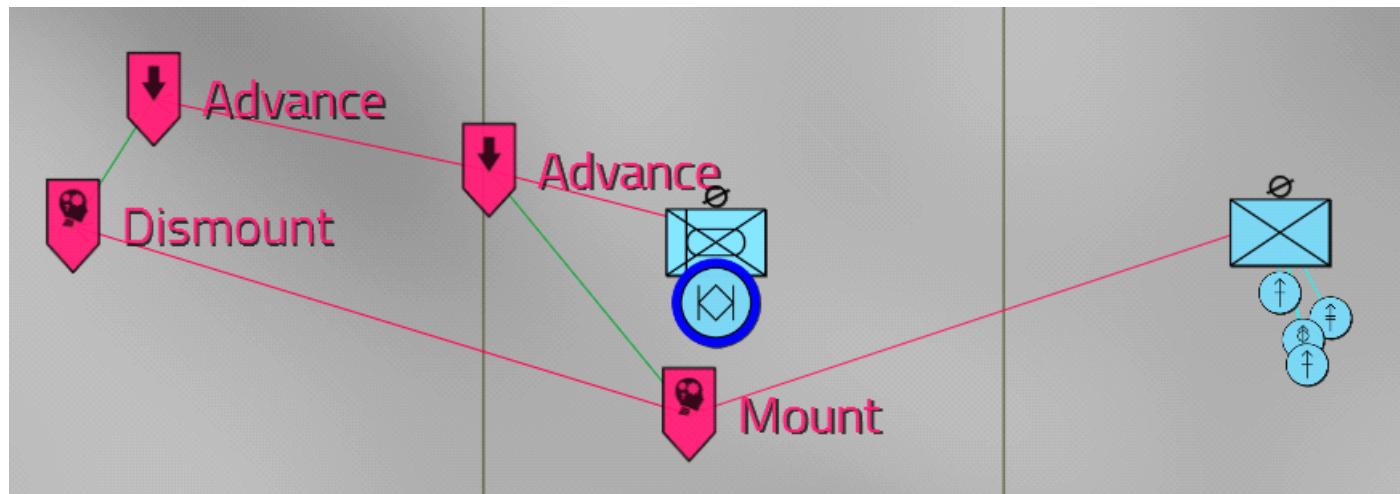
5. Right-click the Dismount Order waypoint and select **Sync to Waypoint**.

6. Click the final Advance Order waypoint.

The Dismount Order is synchronized with the final Advance Order.

7. Click **Preview** to preview the scenario.

After the Stryker reaches the final Advance Order waypoint, the fireteam dismounts.



## 5.3 Interoperability Tutorial

This tutorial demonstrates Control AI interoperability, which means that while a military group executes a primary order, secondary orders can be assigned to sub-groups to be executed in parallel with the primary one. The tutorial builds on the concepts discussed in the [Infantry Tutorial \(on page 54\)](#) and the [Vehicle Tutorial \(on page 58\)](#).

### Scenario Workflow

1. A **US > Armored Brigade Combat Team > Mechanized Infantry Company > Platoon (M2A3)** group starts advancing to the given location.
2. One **M2A3 Bradley** sub-group separates from the **Platoon (M2A3)** group and moves to a pickup location, for a **US > Ground Unit > Infantry > Fireteam** group to mount it as cargo (the **Fireteam** group first advances to the pickup location, and waits for the **M2A3 Bradley** sub-group to arrive).
3. Another **M2A3 Bradley** sub-group separates from the **Platoon (M2A3)** group and also moves to the **M2A3 Bradley** sub-group pickup location, to defend it and the **Fireteam** group.
4. After the **Fireteam** group mounts the first **M2A3 Bradley** sub-group, the latter proceeds to advance to a drop-off location, where the **Fireteam** group dismounts, and the returns to formation and continues advancing with the **Platoon (M2A3)** group. At the same time, the second **M2A3 Bradley** sub-group proceeds to advance from the defend location to return to formation and continues advancing with the **Platoon (M2A3)** group.

The following aspects are discussed:

- [Creating the Groups \(below\)](#)
- [Assigning Orders \(on the next page\)](#)
- [SQF Commands / Functions Interoperability \(on page 66\)](#)

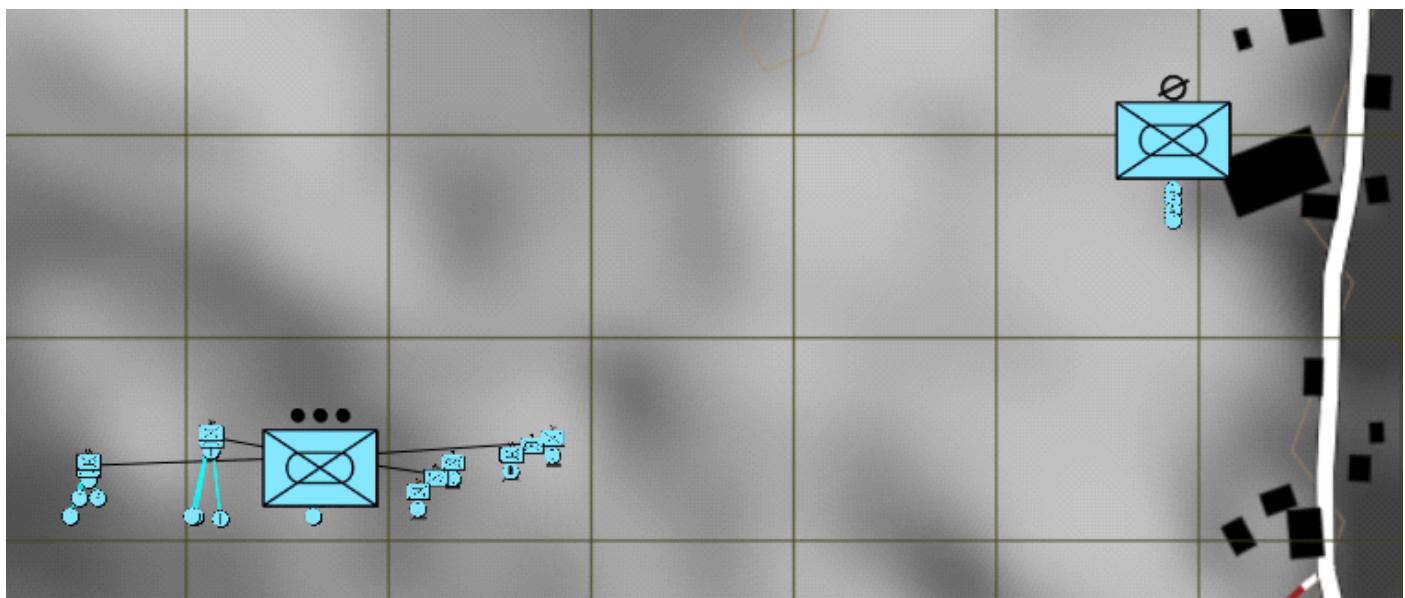
### 5.3.1 Creating the Groups

First, create the groups.

#### Follow these steps:

1. Open the VBS Editor in Prepare mode, and place the following two groups, using the ORBAT Editor (see in the VBS4 Editor Manual):
  - **US > Armored Brigade Combat Team > Mechanized Infantry Company > Platoon (M2A3)**
  - **US > Ground Unit > Infantry > Fireteam**
2. Position the groups, so that the **Platoon (M2A3)** group is situated left of the **Fireteam** group.

The military groups are placed.

**Image-21: Platoon (M2A3) and Fireteam groups on the map**

### 5.3.2 Assigning Orders

Now, assign the orders to the groups and sub-groups.

First, set up one of the **M2A3 Bradley** sub-groups to pick up and drop off the **Fireteam** group.

**Follow these steps:**

1. Right-click the **Platoon (M2A3)** group, and select **Orders > Assign New Waypoint**.
2. Click a map location northeast of the group, and north of the **Fireteam** group.

The Object Properties dialog opens.

3. In the **Behavior** list, select **Advance** and click **OK**.

The **Platoon (M2A3)** group is assigned an [Advance Order \(on page 72\)](#) waypoint, serving as the primary Advance Order.

4. Right-click the **Fireteam** group, and select **Orders > Assign New Waypoint**.

5. Click a map location slightly north of the **Fireteam** group.

The Object Properties dialog opens.

6. In the **Behavior** list, select **Advance** and click **OK**.

The **Fireteam** group is assigned an Advance Order waypoint, to move to the pickup location.

7. Right-click one of the **M2A3 Bradley** sub-groups, and select **Orders > Assign New Waypoint**.

8. Click a map location right next to the **Fireteam** group.

The Object Properties dialog opens.

9. In the **Behavior** list, select **Advance** and click **OK**.

The **M2A3 Bradley** sub-group is assigned an Advance Order waypoint, to move to the pickup location.

10. Right-click the Advance Order waypoint of the **Fireteam** group, and select **Orders > Assign Next Waypoint**.

11. Click a map location right next to the Advance Order waypoint of the **Fireteam** group.

The Object Properties dialog opens.

12. In the **Behavior** list, select **Mount**.

13. In the **Crew Role Category** drop-down, select **Cargo** and click **OK**.

The **Fireteam** group is assigned a [Mount Order \(on page 82\)](#) waypoint, to mount the M2A3 Bradley vehicles in the **M2A3 Bradley** sub-group.

 **WARNING**

Make sure the **M2A3 Bradley** and the **Fireteam** Advance Order waypoints are within 50m of the Mount Order waypoint, for the **Fireteam** to mount the M2A3 Bradley vehicles.

14. Right-click the Mount Order waypoint, and select **Sync to Waypoint**.

15. Click the **M2A3 Bradley** Advance Order waypoint.

This synchronizes the **M2A3 Bradley** Advance Order with the **Fireteam** Mount Order, which means that the M2A3 Bradley vehicles move to the Advance Order waypoint location, and wait for the fireteam to complete the Mount Order.

16. Right-click the **M2A3 Bradley** Advance Order waypoint, and select **Assign Next Waypoint**.

17. Click a map location north of the **M2A3 Bradley** Advance Order waypoint.

The Object Properties dialog opens.

18. In the **Behavior** list, select **Advance** and click **OK**.

The **M2A3 Bradley** sub-group is assigned an Advance Order waypoint, to move to the drop-off location.

19. Right-click the Mount Order waypoint, and select **Assign Next Waypoint**.

20. Click a map location right next to the drop-off Advance Order waypoint.

The Object Properties dialog opens.

21. In the **Behavior** list, select **Dismount**.

22. In the **Crew Role Category** drop-down, select **Cargo** and click **OK**.

The **Fireteam** group is assigned a [Dismount Order \(on page 86\)](#) waypoint at the drop-off location.

23. Right-click the Dismount Order waypoint, and select **Sync to Waypoint**.

24. Click the drop-off Advance Order waypoint.

This synchronizes the drop-off **M2A3 Bradley** Advance Order with the **Fireteam** Dismount Order, for the fireteam to dismount the M2A3 Bradley vehicles.

25. Right-click the drop-off Advance Order waypoint, and select **Assign Next Waypoint**.

26. Click a map location right next or close to the drop-off Advance Order waypoint.

The Object Properties dialog opens.

27. In the **Behavior** list, select **Return to Formation** and click **OK**.

The **M2A3 Bradley** sub-group is assigned a [Return to Formation Order \(on page 89\)](#) waypoint to return to the **Platoon (M2A3)** formation.

Next, set up the second **M2A3 Bradley** sub-group to defend the first **M2A3 Bradley** sub-group and the **Fireteam** group, while the latter mounts the M2A3 Bradley vehicles.

#### Follow these steps:

1. Right-click the second **M2A3 Bradley** sub-group, and select **Orders > Assign New Waypoint**.
2. Click a map location right next to the pickup Advance Order of the first **M2A3 Bradley** sub-group.

The Object Properties dialog opens.

3. In the **Behavior** list, select **Defend** and click **OK**.

The second **M2A3 Bradley** sub-group is assigned a [Defend Order \(on page 80\)](#) waypoint.

4. Right-click the Defend Order waypoint, and select **Sync to Waypoint**.

5. Click the **Fireteam** Mount Order waypoint.

This synchronizes the second **M2A3 Bradley** sub-group Defend Order with the **Fireteam** Mount Order, which means that the defense takes place, until the **Fireteam** finishes mounting the first **M2A3 Bradley** sub-group vehicles.

6. Right-click the Defend Order waypoint, and select **Assign Next Waypoint**.

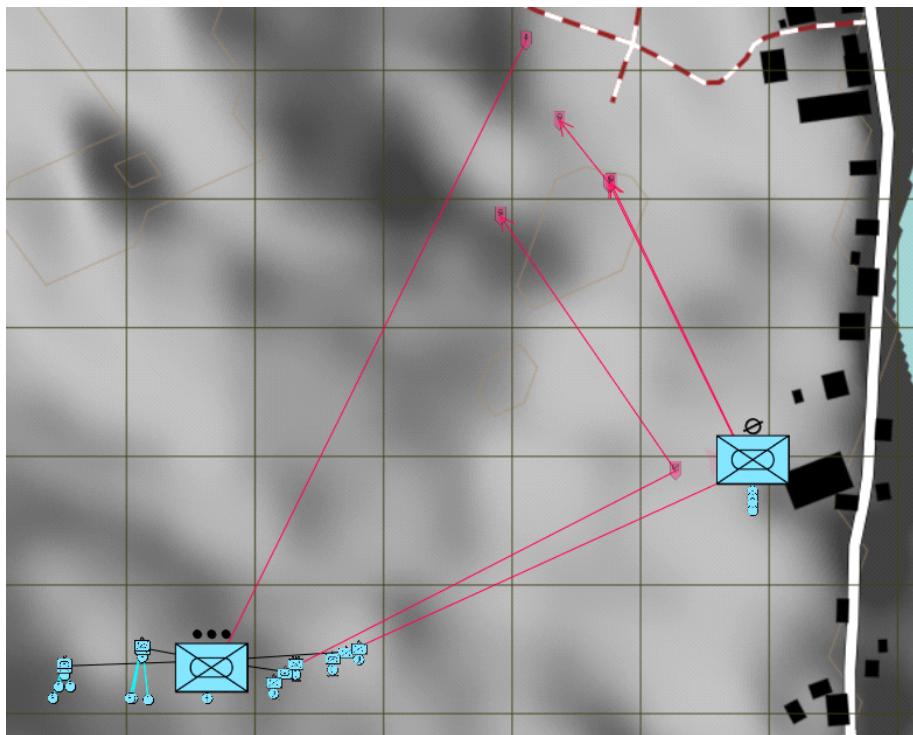
7. Click a map location right next to the Return to Formation Order waypoint of the first **M2A3 Bradley** sub-group.

The Object Properties dialog opens.

8. In the **Behavior** list, select **Return to Formation** and click **OK**.

The second **M2A3 Bradley** sub-group is assigned a Return to Formation Order waypoint to return to the **Platoon (M2A3)** formation.

**Image-22: Final waypoints on the map**



Click **Preview** to preview the scenario and see how the [Scenario Workflow](#) (on page 62) is executed.

### 5.3.3 SQF Commands / Functions Interoperability

Additionally, you can use the SQF commands / functions listed in the [AI Parity](#) (<https://sqf.bisimulations.com/display/SQF/AI+Parity>) section of the VBS Scripting Reference to assign secondary orders to sub-groups to perform in parallel with the primary group orders.

## 5.4 Military AI Orders and Behaviors

Military AI uses the [\(F3\) Waypoints Editor Object \(on page 18\)](#) and consists of the following orders / behaviors:

Order / Behavior Type	Description
Assault Order (on the next page)	<p>The group (infantry / vehicles / both) moves in formation to the given position (the location of the <a href="#">(F3) Waypoints Editor Object (on page 18)</a> on the map), and assaults the enemy.</p> <p>Infantry groups split into fireteams, form a line, and perform bounding to the given position to assault the enemy.</p>
Advance Order (on page 72)	<p>The group (infantry / vehicle / both) advances in formation to the given position (the location of the <a href="#">(F3) Waypoints Editor Object (on page 18)</a> on the map), choosing the fastest path to the destination, while preferring speed of movement over engaging and eliminating the enemy.</p>
Pursue Order (on page 74)	<p>The infantry team (only fireteam behavior) pursues the enemy, until it is eliminated. Then, the team takes cover at their current position.</p>
Suppress Order (on page 77)	<p>The infantry and / or vehicle group forms a firing line and starts firing at the given position area (the location of the <a href="#">(F3) Waypoints Editor Object (on page 18)</a> on the map).</p>
Defend Order (on page 80)	<p>The infantry and / or vehicle group moves to the given position (the location of the <a href="#">(F3) Waypoints Editor Object (on page 18)</a> on the map), forms a line and begins to defend it by firing at enemy forces in the position area.</p>
Mount Order (on page 82)	<p>The infantry team moves to the given position (the location of the <a href="#">(F3) Waypoints Editor Object (on page 18)</a> on the map), and mounts the nearest vehicle at the selected vehicle positions (crew, cargo, or both).</p>
Dismount Order (on page 86)	<p>The infantry teams dismounts the vehicle from the selected vehicle positions (crew, cargo, or both).</p>
Return to Formation Order (on page 89)	<p>The infantry / vehicle group returns back under the command of a higher-echelon group.</p>
Tactical Move Order (on page 90)	<p>The infantry group performs a tactical move to the given position (the location of the <a href="#">(F3) Waypoints Editor Object (on page 18)</a> on the map), using a given stance, formation, and a maximum range to engage enemy forces at.</p>

### NOTE

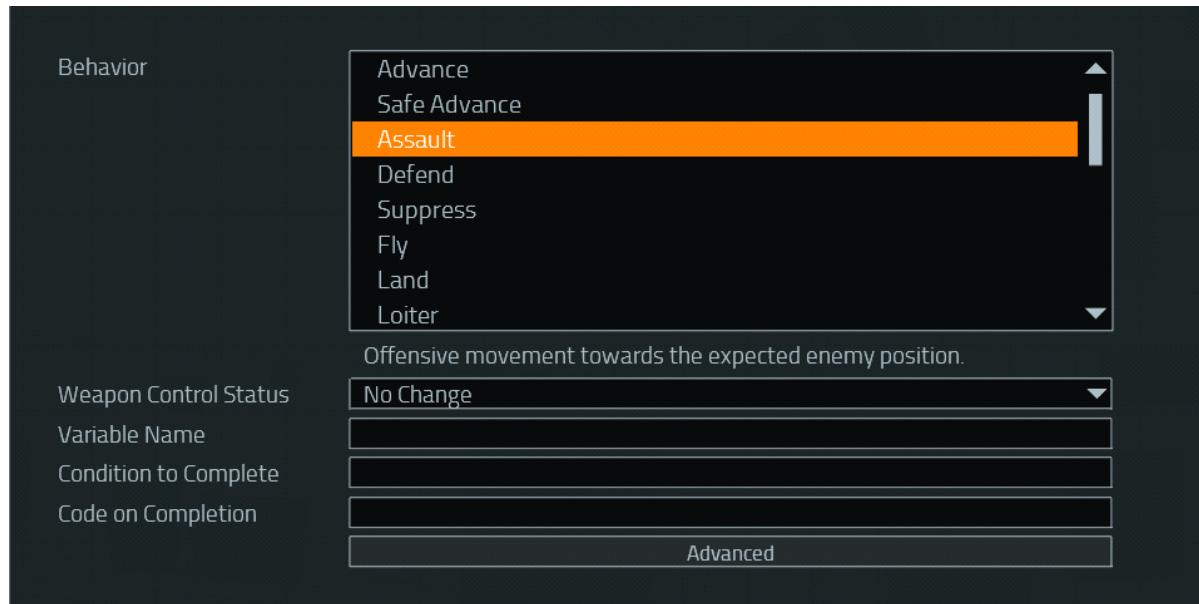
VBS Plan (see VBS Plan Overview in the VBS Plan Manual) relies on the military AI behaviors. For more information on how to use them, see Order Tactical Tools in Tactical Objects in the VBS Plan Manual.

## 5.4.1 Assault Order

The group (infantry / vehicles / both) moves in formation to the given position (the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map), and assaults the enemy.

Infantry groups split into fireteams, form a line, and perform bounding to the given position to assault the enemy.

**Image-23: Assault Order settings**



**Follow these steps:**

1. Select the **(F3) Waypoints** Editor Object from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Assault**.

3. In the **Weapon Control Status** drop-down, select the behavior for engaging enemy forces, when they are encountered:

Option	Description
No Change	No change in the behavior upon encountering enemy forces.
Weapons Free	Fire at enemy forces, when they are encountered.
Hold Fire	Do not fire at enemy forces, when they are encountered.

 **NOTE**

The Weapon Control Status is related to the following SQF commands:

- [combatMode](https://sqf.bisimulations.com/display/SQF/combatMode) (<https://sqf.bisimulations.com/display/SQF/combatMode>)
- [setCombatMode](https://sqf.bisimulations.com/display/SQF/setCombatMode) (<https://sqf.bisimulations.com/display/SQF/setCombatMode>)
- [unitCombatMode](https://sqf.bisimulations.com/display/SQF/unitCombatMode) (<https://sqf.bisimulations.com/display/SQF/unitCombatMode>)
- [setUnitCombatMode](https://sqf.bisimulations.com/display/SQF/setUnitCombatMode)  
(<https://sqf.bisimulations.com/display/SQF/setUnitCombatMode>)

4. In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.

5. Set the Order waypoint completion settings:

Option	Description
Condition to Complete	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
Code on Completion	SQF code to execute on waypoint completion.

6. Click **OK**.

The Assault Order behavior is set up.

 **WARNING**

Bohemia Interactive Simulations advises scenario Administrators not to change the **Advanced** settings for the behavior to work correctly.

 **NOTE**

For some ORBATs, the Assault Order is identical to an [Advance Order](#) (on the next page).

## 5.4.2 Advance Order

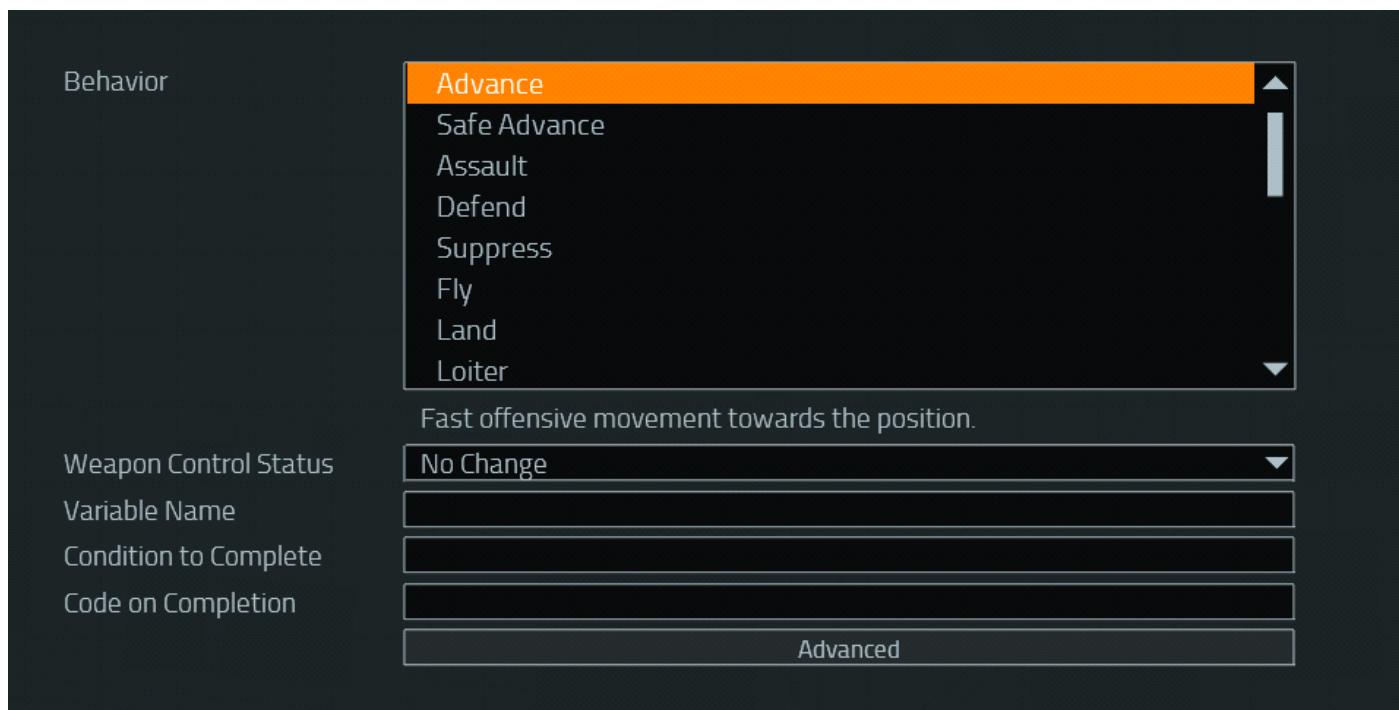
The group (infantry / vehicle / both) advances in formation to the given position (the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map), choosing the fastest path to the destination, while preferring speed of movement over engaging and eliminating the enemy.

### NOTE

The following considerations apply:

- In a chain of (F3) Waypoints Editor Object Orders (see [Waypoints \(on page 23\)](#)), the group occupies a position only if the Advance Order is the last (F3) Waypoints Editor Object.
- The Advance Order applies to infantry, land vehicles, and watercraft, while aircraft use the [Fly Order \(on page 124\)](#), [Loiter Order \(on page 129\)](#), and [Land Order \(on page 127\)](#) to move to the given position.
- Land forces can also use the [Tactical Move Order \(on page 90\)](#).

**Image-24: Advance Order settings**



### Follow these steps:

1. Select the **(F3) Waypoints** Editor Object from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Advance**.

3. In the **Weapon Control Status** drop-down, select the behavior for engaging enemy forces, when they are encountered:

Option	Description
No Change	No change in the behavior upon encountering enemy forces.
Weapons Free	Fire at enemy forces, when they are encountered.
Hold Fire	Do not fire at enemy forces, when they are encountered.

 **NOTE**

The Weapon Control Status is related to the following SQF commands:

- [combatMode](https://sqf.bisimulations.com/display/SQF/combatMode) (<https://sqf.bisimulations.com/display/SQF/combatMode>)
- [setCombatMode](https://sqf.bisimulations.com/display/SQF/setCombatMode) (<https://sqf.bisimulations.com/display/SQF/setCombatMode>)
- [unitCombatMode](https://sqf.bisimulations.com/display/SQF/unitCombatMode) (<https://sqf.bisimulations.com/display/SQF/unitCombatMode>)
- [setUnitCombatMode](https://sqf.bisimulations.com/display/SQF/setUnitCombatMode)  
(<https://sqf.bisimulations.com/display/SQF/setUnitCombatMode>)

4. In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.

5. Set the Order waypoint completion settings:

Option	Description
Condition to Complete	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
Code on Completion	SQF code to execute on waypoint completion.

6. Click **OK**.

The Advance Order behavior is set up.

 **WARNING**

Bohemia Interactive Simulations advises scenario Administrators not to change the **Advanced** settings for the behavior to work correctly.

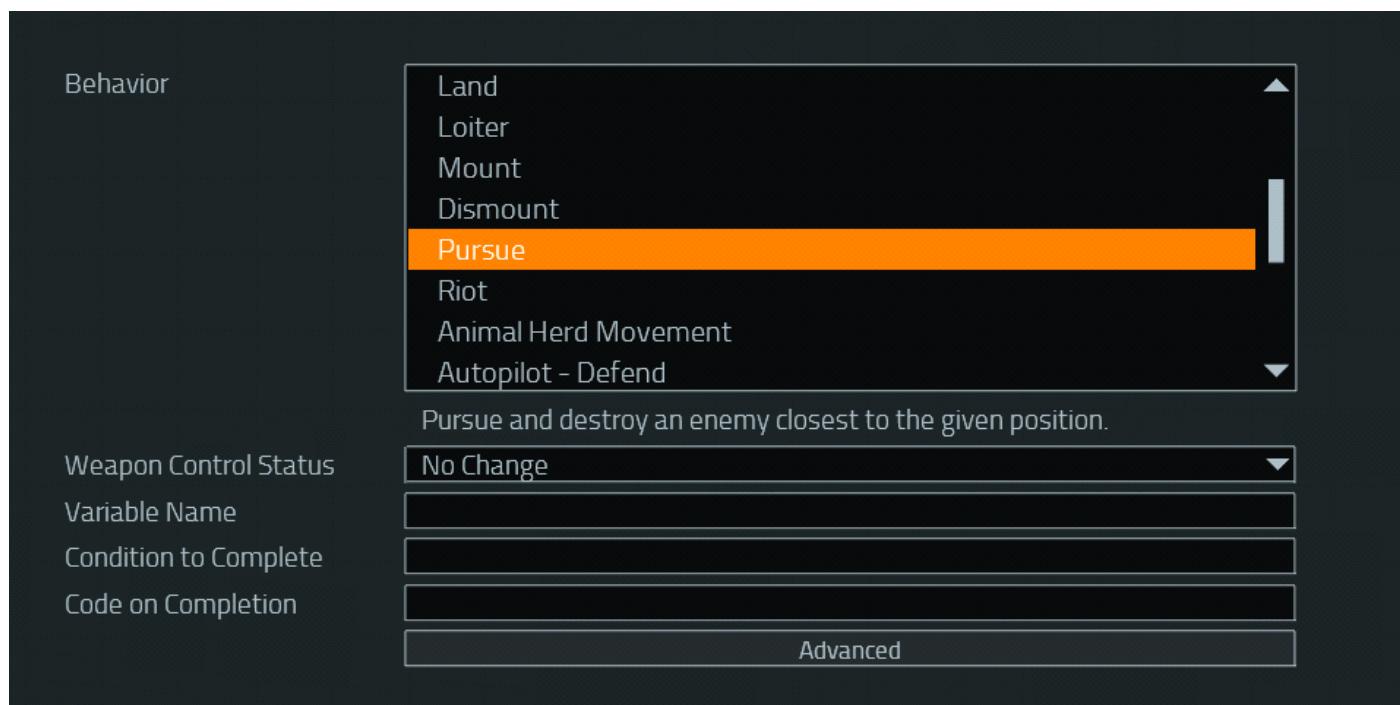
### 5.4.3 Pursue Order

The infantry team (only fireteam behavior) pursues the enemy, until it is eliminated. Then, the team takes cover at their current position.

#### **WARNING**

For the Pursue Order to execute, the enemy needs to be within a 20m radius from the Pursue Order waypoint (the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map). Alternatively, you can explicitly set the target enemy to pursue - see [Target Specification \(on page 76\)](#).

**Image-25: Pursue Order settings**



#### Follow these steps:

1. Select the **(F3) Waypoints** Editor Object from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Pursue**.

3. In the **Weapon Control Status** drop-down, select the behavior for engaging enemy forces, when they are encountered:

Option	Description
No Change	No change in the behavior upon encountering enemy forces.
Weapons Free	Fire at enemy forces, when they are encountered.
Hold Fire	Do not fire at enemy forces, when they are encountered.

 **NOTE**

The Weapon Control Status is related to the following SQF commands:

- [combatMode](https://sqf.bisimulations.com/display/SQF/combatMode) (<https://sqf.bisimulations.com/display/SQF/combatMode>)
- [setCombatMode](https://sqf.bisimulations.com/display/SQF/setCombatMode) (<https://sqf.bisimulations.com/display/SQF/setCombatMode>)
- [unitCombatMode](https://sqf.bisimulations.com/display/SQF/unitCombatMode) (<https://sqf.bisimulations.com/display/SQF/unitCombatMode>)
- [setUnitCombatMode](https://sqf.bisimulations.com/display/SQF/setUnitCombatMode)  
(<https://sqf.bisimulations.com/display/SQF/setUnitCombatMode>)

4. In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.

5. Set the Order waypoint completion settings:

Option	Description
Condition to Complete	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
Code on Completion	SQF code to execute on waypoint completion.

6. Click **OK**.

The Pursue Order behavior is set up.

 **WARNING**

Bohemia Interactive Simulations advises scenario Administrators not to change the **Advanced** settings for the behavior to work correctly.

### 5.4.3.1 Target Specification

You can explicitly set the target enemy to pursue in Prepare / Preview / Execute / Trainee (C2) mode.

**Follow these steps:**

1. Right-click the Pursue Order waypoint, and select **Select Target**.
2. Click the enemy group to pursue.

The enemy group is set as the target group to pursue.

 **NOTE**

It is not possible to set a single enemy unit as a target to pursue. The target has to be a group.

## 5.4.4 Suppress Order

The infantry and / or vehicle group forms a firing line and starts firing at the given position area (the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map).

**Image-26: Suppress Order settings**



### Follow these steps:

1. Select the **(F3) Waypoints** Editor Object from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Suppress**.

- In the **Weapon Control Status** drop-down, select the behavior for engaging enemy forces, when they are encountered:

Option	Description
No Change	No change in the behavior upon encountering enemy forces.
Weapons Free	Fire at enemy forces, when they are encountered.
Hold Fire	Do not fire at enemy forces, when they are encountered.

 **NOTE**

The Weapon Control Status is related to the following SQF commands:

- [combatMode](https://sqf.bisimulations.com/display/SQF/combatMode) (<https://sqf.bisimulations.com/display/SQF/combatMode>)
- [setCombatMode](https://sqf.bisimulations.com/display/SQF/setCombatMode) (<https://sqf.bisimulations.com/display/SQF/setCombatMode>)
- [unitCombatMode](https://sqf.bisimulations.com/display/SQF/unitCombatMode) (<https://sqf.bisimulations.com/display/SQF/unitCombatMode>)
- [setUnitCombatMode](https://sqf.bisimulations.com/display/SQF/setUnitCombatMode)  
(<https://sqf.bisimulations.com/display/SQF/setUnitCombatMode>)

- In **Radius**, set the fire support radius (in meters).
- In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.
- Set the Order waypoint completion settings:

Option	Description
Condition to Complete	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
Code on Completion	SQF code to execute on waypoint completion.

- Click **OK**.

The Suppress Order behavior is set up.

 **WARNING**

Bohemia Interactive Simulations advises scenario Administrators not to change the **Advanced** settings for the behavior to work correctly.

#### 5.4.4.1 Range Visualization

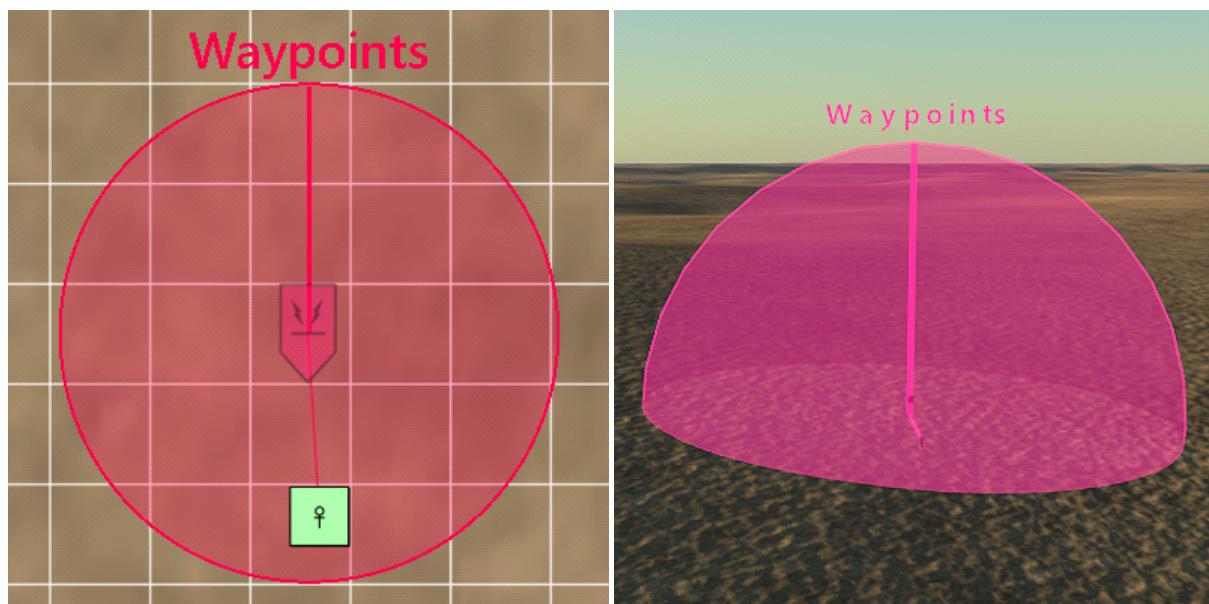
In VBS4, the **Radius** of the Suppress Order waypoint can be displayed as a range visualization.

**NOTE**

Use the Range Visibility Settings in the VBS4 Administrator Manual to enable / disable the entire range visualization or individual elements of it.

Range visualizations can only be seen by scenario Administrators and Instructors, not Trainees.

Image-27: 2D 3D range visualizations



## 5.4.5 Defend Order

The infantry and / or vehicle group moves to the given position (the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map), forms a line and begins to defend it by firing at enemy forces in the position area.

**Image-28: Defend Order settings**



### Follow these steps:

1. Select the **(F3) Waypoints** Editor Object from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Defend**.

3. In the **Weapon Control Status** drop-down, select the behavior for engaging enemy forces, when they are encountered:

Option	Description
No Change	No change in the behavior upon encountering enemy forces.
Weapons Free	Fire at enemy forces, when they are encountered.
Hold Fire	Do not fire at enemy forces, when they are encountered.

 **NOTE**

The Weapon Control Status is related to the following SQF commands:

- [combatMode](https://sqf.bisimulations.com/display/SQF/combatMode) (<https://sqf.bisimulations.com/display/SQF/combatMode>)
- [setCombatMode](https://sqf.bisimulations.com/display/SQF/setCombatMode) (<https://sqf.bisimulations.com/display/SQF/setCombatMode>)
- [unitCombatMode](https://sqf.bisimulations.com/display/SQF/unitCombatMode) (<https://sqf.bisimulations.com/display/SQF/unitCombatMode>)
- [setUnitCombatMode](https://sqf.bisimulations.com/display/SQF/setUnitCombatMode)  
(<https://sqf.bisimulations.com/display/SQF/setUnitCombatMode>)

4. In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.

5. Set the Order waypoint completion settings:

Option	Description
Condition to Complete	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
Code on Completion	SQF code to execute on waypoint completion.

6. Click **OK**.

The Defend Order behavior is set up.

 **WARNING**

Bohemia Interactive Simulations advises scenario Administrators not to change the **Advanced** settings for the behavior to work correctly.

## 5.4.6 Mount Order

The infantry team moves to the given position (the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map), and mounts the nearest vehicle at the selected vehicle positions (crew, cargo, or both).

### **WARNING**

For the Mount Order to execute, the vehicle needs to be within the allowed radius from the Mount Order waypoint (the radius is determined by allotting 50 m for every ORBAT level). Alternatively, you can explicitly set the target vehicle to mount - see [Target Specification \(on page 85\)](#).

### **NOTE**

The following considerations apply:

- You can assign the Mount Order to a mixed group of units and vehicles. This means that the units already mounted do nothing, while the units on foot mount the vehicle. Assigning a Mount Order to a vehicle or a group of vehicle does nothing.
- When used with aircraft, only rotary-wing aircraft are supported.
- Can be used with watercraft. Units can only mount on land.

To order units to dismount from vehicles, see the [Dismount Order \(on page 86\)](#).

**Image-29: Mount Order settings**



**Follow these steps:**

1. Select the **(F3) Waypoints** Editor Object from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Mount**.
3. In the **Weapon Control Status** drop-down, select the behavior for engaging enemy forces, when they are encountered:

Option	Description
<b>No Change</b>	No change in the behavior upon encountering enemy forces.
<b>Weapons Free</b>	Fire at enemy forces, when they are encountered.
<b>Hold Fire</b>	Do not fire at enemy forces, when they are encountered.

**NOTE**

The Weapon Control Status is related to the following SQF commands:

- [combatMode](https://sqf.bisimulations.com/display/SQF/combatMode) (<https://sqf.bisimulations.com/display/SQF/combatMode>)
- [setCombatMode](https://sqf.bisimulations.com/display/SQF/setCombatMode) (<https://sqf.bisimulations.com/display/SQF/setCombatMode>)
- [unitCombatMode](https://sqf.bisimulations.com/display/SQF/unitCombatMode) (<https://sqf.bisimulations.com/display/SQF/unitCombatMode>)
- [setUnitCombatMode](https://sqf.bisimulations.com/display/SQF/setUnitCombatMode)  
(<https://sqf.bisimulations.com/display/SQF/setUnitCombatMode>)

4. In the **Crew Role Category** drop-down, select vehicle mount positions:

Option	Description
All	Mount both as crew and cargo.
Crew	Mount as crew.
Cargo	Mount as cargo.

 **NOTE**

The following considerations apply:

- If the capacity of one vehicle is exceeded and there are additional vehicles within the allowed radius from the Mount Order waypoint (the radius is determined by allotting 50 m for every ORBAT level), the units mount the next vehicle, as needed, occupying the positions of the selected type.
- If **Crew Role Category** is set to **Crew** or **All**, and if a non-cargo vehicle position (see Vehicle Positions in the VBS4 Trainee Manual), such as the Driver, Commander, Gunner, and so on, becomes free or is free and should be occupied, other units (including those in the cargo, if there are any) try to fill in for the missing unit at that position.
- Destroyed vehicles are not mounted. Also, if the vehicle is destroyed, any units inside the vehicle that are capable of dismounting (not dead or incapacitated) automatically do so.

5. In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.

6. Set the Order waypoint completion settings:

Option	Description
Condition to Complete	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
Code on Completion	SQF code to execute on waypoint completion.

7. Click **OK**.

The Mount Order behavior is set up.

 **WARNING**

Bohemia Interactive Simulations advises scenario Administrators not to change the **Advanced** settings for the behavior to work correctly.

 **TIP**

If you want the vehicle to drive to the pick-up location before the units mount, you can synchronize the Mount waypoint / Order with another Order (for example, the [Advance Order \(on page 72\)](#)). For more information, see [Waypoint Synchronization \(on page 26\)](#).

#### 5.4.6.1 Target Specification

You can explicitly set the target vehicle to mount in Prepare / Preview / Execute / Trainee (C2) mode.

**Follow these steps:**

1. Right-click the Mount Order waypoint, and select **Select Target**.
2. Click the vehicle to mount.

The vehicle is set as the target vehicle to mount.

## 5.4.7 Dismount Order

The infantry teams dismounts the vehicle from the selected vehicle positions (crew, cargo, or both).

### NOTE

The following considerations apply:

- The Dismount Order can be assigned to a group of units, occupying one or more vehicles (that is, if the group of units is distributed among multiple vehicles). Alternatively, if the group of units is inside one vehicle, you can assign the Dismount Order to that vehicle.



### WARNING

If the units are distributed among several vehicles, and you want them to dismount from crew or cargo positions, make sure the positions they occupy are of the same type (crew, cargo).

- The Dismount Order waypoint position (the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map) has no effect on where the units dismount - the dismount happens immediately as soon as the Dismount Order waypoint becomes active.
- You can assign the Dismount Order to a mixed group of units and vehicles. This means that the units already on foot do nothing, while the units inside the vehicle dismount. Assigning a Dismount Order to a group of units on foot does nothing.
- When used with aircraft (only rotary-wing aircraft are supported), units can only dismount if the aircraft is on the ground.
- Can be used with watercraft. Units can only dismount on land. The maximum dismount distance between the shore and the watercraft is 50 m.

To order units to mount vehicles, see the [Mount Order \(on page 82\)](#).

### Image-30: Dismount Order settings



#### Follow these steps:

1. Select the **(F3) Waypoints** Editor Object from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Dismount**.
3. In the **Weapon Control Status** drop-down, select the behavior for engaging enemy forces, when they are encountered:

Option	Description
No Change	No change in the behavior upon encountering enemy forces.
Weapons Free	Fire at enemy forces, when they are encountered.
Hold Fire	Do not fire at enemy forces, when they are encountered.



#### NOTE

The Weapon Control Status is related to the following SQF commands:

- [combatMode](https://sqf.bisimulations.com/display/SQF/combatMode) (<https://sqf.bisimulations.com/display/SQF/combatMode>)
- [setCombatMode](https://sqf.bisimulations.com/display/SQF/setCombatMode) (<https://sqf.bisimulations.com/display/SQF/setCombatMode>)
- [unitCombatMode](https://sqf.bisimulations.com/display/SQF/unitCombatMode) (<https://sqf.bisimulations.com/display/SQF/unitCombatMode>)
- [setUnitCombatMode](https://sqf.bisimulations.com/display/SQF/setUnitCombatMode)  
(<https://sqf.bisimulations.com/display/SQF/setUnitCombatMode>)

4. In the **Crew Role Category** drop-down, select vehicle dismount positions:

Option	Description
All	Dismount both crew and cargo units.
Crew	Dismount crew units.
Cargo	Dismount cargo units.

5. In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.

6. Set the Order waypoint completion settings:

Option	Description
Condition to Complete	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
Code on Completion	SQF code to execute on waypoint completion.

7. Click **OK**.

The Dismount Order behavior is set up.

 **TIP**

If you want the vehicle to drive away from the drop-off location after the units dismount, you can synchronize the Dismount waypoint / Order with another Order (for example, the [Advance Order \(on page 72\)](#)). For more information, see [Waypoint Synchronization \(on page 26\)](#).

## 5.4.8 Return to Formation Order

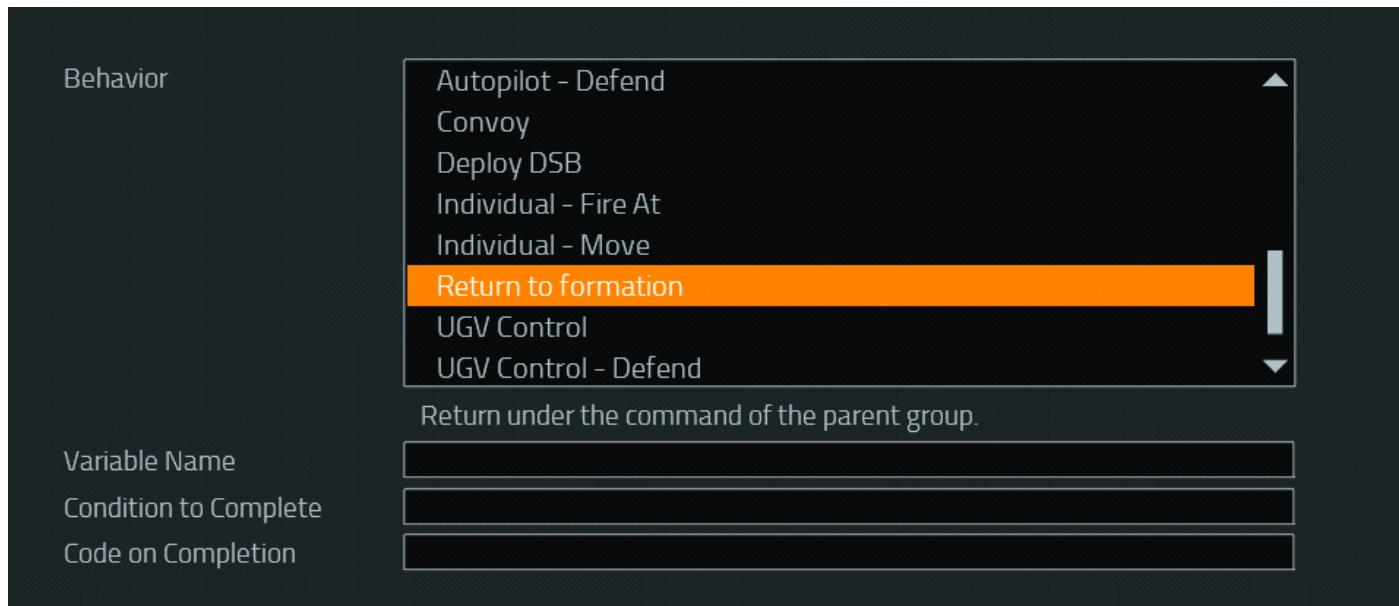
The infantry / vehicle group returns back under the command of a higher-echelon group.

### **NOTE**

Sub-groups can execute Orders apart from and in parallel with the Order executed by their larger group (for example, a higher echelon) - see [Interoperability Tutorial \(on page 62\)](#).

The Return to Formation Order is used to enable sub-groups to return to the formation of the larger group, and to continue executing the larger-group Order.

**Image-31: Return to Formation Order settings**



### Follow these steps:

1. Select the **(F3) Waypoints** Editor Object from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Return to Formation**.
3. In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.
4. Set the Order waypoint completion settings:

Option	Description
<b>Condition to Complete</b>	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
<b>Code on Completion</b>	SQF code to execute on waypoint completion.

5. Click **OK**.

The Return to Formation Order behavior is set up.

## 5.4.9 Tactical Move Order

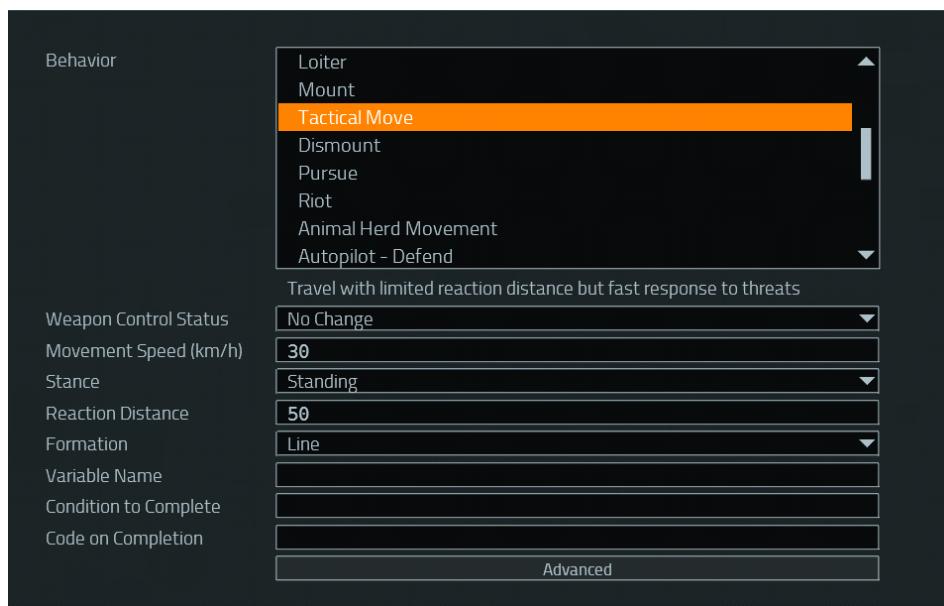
The infantry group performs a tactical move to the given position (the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map), using a given stance, formation, and a maximum range to engage enemy forces at.

### NOTE

The following considerations apply:

- When an enemy is spotted within the given reaction distance, or on incoming fire from anywhere (regardless of range), the group takes cover near its current position and holds the position indefinitely.
- Unlike the [Advance Order \(on page 72\)](#), units do not take cover at the destination position.
- The Tactical Move Order can only be used with infantry groups.

**Image-32: Tactical Move Order settings**



### Follow these steps:

1. Select the **(F3) Waypoints** Editor Object from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Tactical Move**.

3. In the **Weapon Control Status** drop-down, select the behavior for engaging enemy forces, when they are encountered:

Option	Description
No Change	No change in the behavior upon encountering enemy forces.
Weapons Free	Fire at enemy forces, when they are encountered.
Hold Fire	Do not fire at enemy forces, when they are encountered.

**NOTE**

The Weapon Control Status is related to the following SQF commands:

- [combatMode](https://sqf.bisimulations.com/display/SQF/combatMode) (<https://sqf.bisimulations.com/display/SQF/combatMode>)
- [setCombatMode](https://sqf.bisimulations.com/display/SQF/setCombatMode) (<https://sqf.bisimulations.com/display/SQF/setCombatMode>)
- [unitCombatMode](https://sqf.bisimulations.com/display/SQF/unitCombatMode) (<https://sqf.bisimulations.com/display/SQF/unitCombatMode>)
- [setUnitCombatMode](https://sqf.bisimulations.com/display/SQF/setUnitCombatMode)  
(<https://sqf.bisimulations.com/display/SQF/setUnitCombatMode>)

4. In **Movement Speed (km/h)**, set the maximum speed (in km/h) at which the infantry group should move to the destination position.
5. In the **Stance** drop-down, select the stance (Standing, Crouched, Prone) for the infantry group to move in.
6. In **Reaction Distance**, set the maximum range (in meters) to detect enemy forces at.
7. In the **Formation** drop-down, select the formation for the infantry group to move in:
- |                    |                |                 |
|--------------------|----------------|-----------------|
| • Line             | • Wedge        | • Echelon Right |
| • File             | • Column       | • Vee           |
| • Staggered Column | • Echelon Left | • Diamond       |
8. In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.
9. Set the Order waypoint completion settings:

Option	Description
Condition to Complete	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
Code on Completion	SQF code to execute on waypoint completion.

10. Click **OK**.

The Tactical Move Order behavior is set up.

 **WARNING**

Bohemia Interactive Simulations advises scenario Administrators not to change the **Advanced** settings for the behavior to work correctly.

## 6. Convoy AI

You can create a convoy in VBS4 using the [\(F3\) Waypoints Editor Object \(on page 18\)](#) that simulates the movement of a vehicle convoy. For more information on convoy properties, see [Convoy Order \(on page 105\)](#).

The convoy can be assigned waypoints (see [Waypoints \(on page 23\)](#)) and use branching (see [Branching Waypoints \(on page 25\)](#)).

The convoy Control AI behavior has the following use cases:

- [Fully Autonomous Convoy \(on the next page\)](#) - A convoy that is fully autonomous, with vehicle crews that consist only of Control AI entities.
- [Pause / Resume Convoy \(on page 96\)](#) - A convoy that can pause / resume movement, using the [Convoy SQF Functions \(on page 103\)](#).
- [Player Units in Convoy \(on page 101\)](#) - Transporting player units in a convoy.

## 6.1 Fully Autonomous Convoy

You can create a fully autonomous convoy, with vehicle crews that are Control AI entities.

### Follow these steps:

1. In the Editor (Prepare Mode), place several vehicles on the map using the **(F4) Vehicle** Editor Object in the Editor Objects List.

These vehicles are used for the convoy.

2. Rotate and position the vehicles so that they face the direction of the convoy movement.

The vehicles set the convoy order autonomously, based on how they are positioned on the road. The vehicle closest to the convoy destination is chosen as the lead.

3. To form the vehicles into a convoy, link the vehicles together, so that they are all in the same group.

#### NOTE

It does not matter which vehicle is set as the group leader. Also, you can link new vehicles to the convoy group (by linking to the convoy leader) in the Editor (Execute Mode), while the convoy is already driving - the new vehicles follow as part of the convoy.

4. To assign a destination, create a convoy waypoint (see [Waypoints \(on page 23\)](#)) by right-clicking the convoy group marker, selecting **Orders > Assign New Waypoint**, and clicking a position on the map, where the convoy waypoint needs to be created.

#### WARNING

If you need to create several convoy waypoints connected to one another, position them in such a way that would not require the convoy to turn around. For more information, see the **Convoy** section in [Known Issues \(on page 231\)](#).

5. Set the convoy properties (see [Convoy Order \(on page 105\)](#)) and click **OK**.
6. Preview the mission.

The convoy starts moving.

**Road usage** controls whether the convoy drives in the middle of the road (the **Use only roads** option), whether it respects lanes (the **Use only roads, respect lanes** option), whether it drives off the road (the **Ignore roads** option), or whether it prefers driving on roads while also being able to drive off-road (the **Prefer roads** option). The **Use only roads, respect lanes** option should be used in case there are other vehicles on the road, such as civilian traffic (see [Define Traffic Flows \(on page 151\)](#)).

A convoy is able to move through most bushes and fences. All objects which cannot be moved through are excluded from the vehicle navigation mesh.

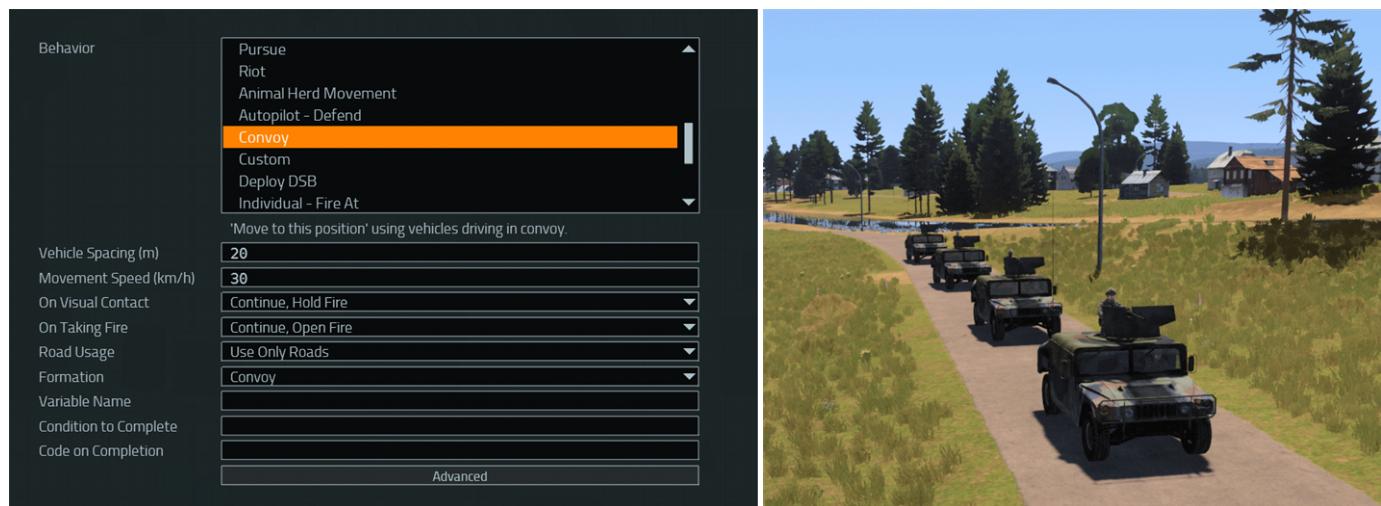
Convoy navigation with the **Ignore roads** option set may be limited in some areas:

Limitation	Solution
<b>Urban area</b>	Setting the convoy to drive on roads through the city or village may avoid issues with navigation through these areas.
<b>High object density area (for example, with small rocks, forests)</b>	Adding intermediate waypoints or removing objects from the high density area improves the navigation.

### TIP

For multi-lane traffic, if **Use Only Roads, Respect Lanes** is used, you can either use the **AI Debug** option in VBS Geo (see VBS Geo User Interface in the VBS Geo Manual), or the **Show Roads and Generate Roads** options in the AI Debug Panel (see [Control AI Visualization \(on page 210\)](#)), to see how your road network is set up. Also, to configure road lanes, see **Lanes** in Placing and Editing Roads in the VBS Geo Manual.

**Image-33: Convoy dialog and example**



## 6.2 Pause / Resume Convoy

You can control the convoy behavior during its execution by pausing / resuming its movement. Convoy movement can be paused / resumed using the SQF functions [fn\\_vbsCon\\_pauseConvoy \(on page 103\)](#) / [fn\\_vbsCon\\_resumeConvoy \(on page 104\)](#) in the following ways (each way is based on the [Fully Autonomous Convoy \(on page 94\)](#) setup), for example:

- Using a Radio Trigger (below)
- Using an Area Trigger (on page 98)
- Using the Player Quick Menu (on page 100)

### Using a Radio Trigger

You can create a basic user interface to control the convoy movement using two radio triggers - one to pause the convoy movement, and another to resume it.

#### Follow these steps:

1. Set up the convoy as described in the [Fully Autonomous Convoy \(on page 94\)](#) section.
2. Set the name of the lead convoy vehicle to `leadVehicle`.



#### TIP

Alternatively, you can use any other vehicle in the convoy. Since `leadVehicle` can get destroyed during the mission, it is best to save the convoy group of `leadVehicle` in a variable.

```
convoyGroup = group leadVehicle
```

3. Set the **Initialization Statements** of `leadVehicle` to:

```
convoyGroup = group leadVehicle
```

4. To populate the radio menu with a radio call to pause the convoy, create a trigger:

- **Text:** Pause convoy
- **Activation:** Radio Alpha
- **Repeatedly:** true
- **On Activation:**

```
convoyGroup call fn_vbsCon_pauseConvoy
```

5. To populate the radio menu with a radio call to resume the convoy, create a trigger:

- **Text:** Resume convoy
- **Activation:** Radio Bravo
- **Repeatedly:** true
- **On Activation:**

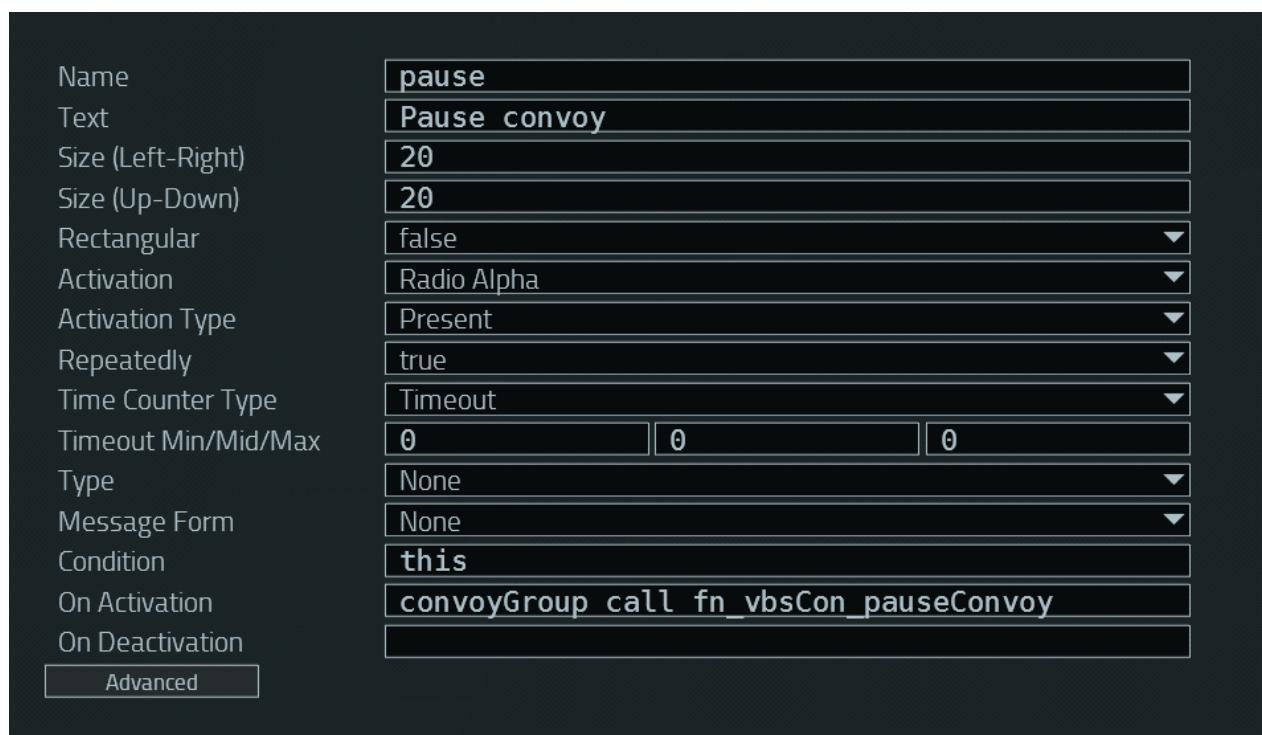
```
convoyGroup call fn_vbsCon_resumeConvoy
```

6. Preview the mission. To view the two radio calls:

- If you are an administrator, switch to the Editor (Execute Mode), then pause / resume the convoy by clicking the radio in the Editor (Execute Mode).
- Any group leader in the scenario can access the same convoy commands by pressing **0-0** to access the radio menu.

The two radio calls pause / resume the convoy movement.

**Image-34: The radio trigger dialog**



**Image-35: The radio menu with the two radio calls**

## Using an Area Trigger

Alternatively to radio triggers, you can pre-script the convoy to pause / resume movement by using area triggers.

### Follow these steps:

1. Set up the convoy as described in the [Fully Autonomous Convoy \(on page 94\)](#) section.
2. Set the name of the lead convoy vehicle to `leadVehicle`.



#### TIP

Alternatively, you can use any other vehicle in the convoy. Since `leadVehicle` can get destroyed during the mission, it is best to save the convoy group of `leadVehicle` in a variable.

```
convoyGroup = group leadVehicle
```

3. Set the **Initialization Statements** of `leadVehicle` to:

```
convoyGroup = group leadVehicle
```

4. Create a stopping zone for `leadVehicle` to enter, using an area trigger.

Set the area trigger to:

**Size:** Can be relatively small to avoid triggering the area by some other unit.

**Activation:** BLUFOR

**Activation Type:** Present

**On Activation:**

```
convoyGroup call fn_vbsCon_pauseConvoy
```

5. Preview the mission.

The convoy stops moving whenever any BLUFOR vehicle or unit enters the zone designated by the area trigger.



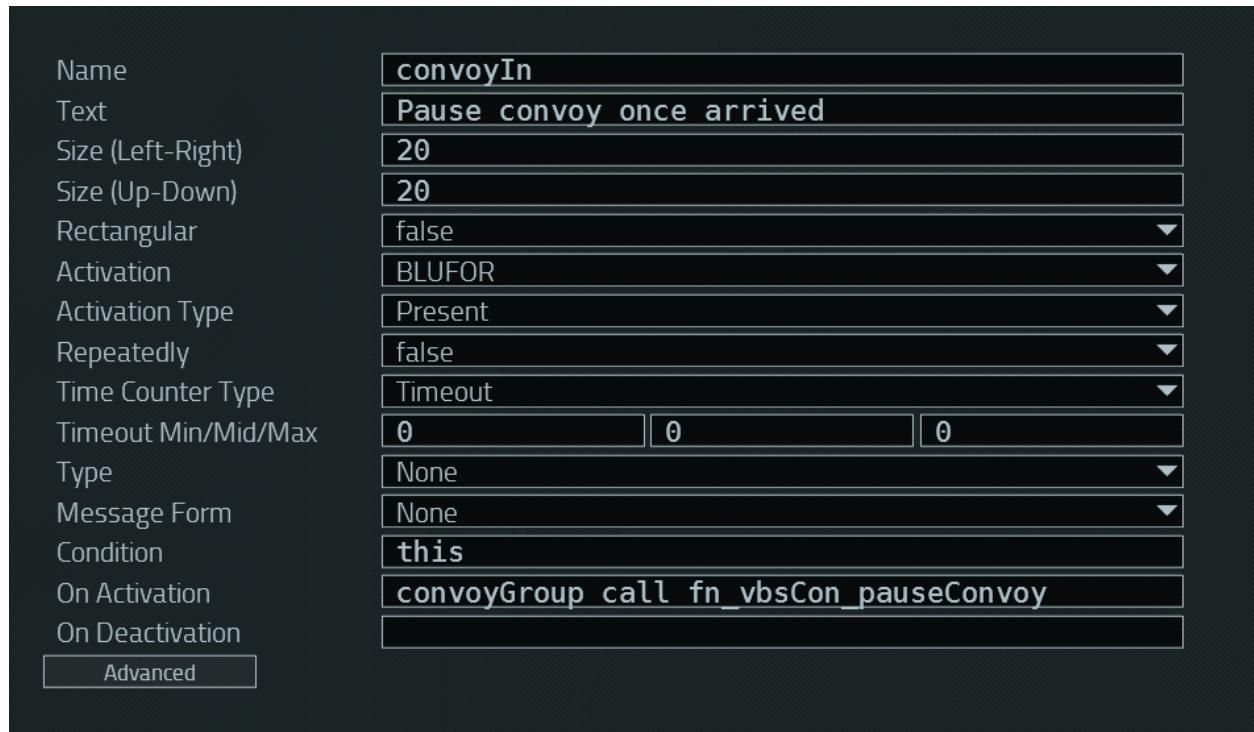
**TIP**

To modify the trigger so that it only causes the convoy to stop when `leadVehicle` enters the trigger, change the trigger **Condition** to:

```
this and (leadVehicle in thislist)
```

To resume convoy movement, run [fn\\_vbsCon\\_resumeConvoy \(on page 104\)](#).

**Image-36: The area trigger dialog**



## Using the Player Quick Menu

You can create user actions in the Quick Menu to pause / resume the convoy movement.

### Follow these steps:

1. Set up a convoy as described in the [Fully Autonomous Convoy \(on page 94\)](#) section.
2. Set the name of the lead convoy vehicle to `leadVehicle`.



#### TIP

Alternatively, you can use any other vehicle in the convoy.

3. Create an `init.sqf` file in the mission, with the following code to add to user actions (pause and resume) to the player:

```
_i1 = player addAction ["Pause convoy", "pause.sqf", leadVehicle];  
_i2 = player addAction ["Resume convoy", "resume.sqf", leadVehicle];
```

For more information, see [addAction](#) (<https://sqf.bisimulations.com/display/SQF/addAction>).

4. Create a `pause.sqf` file in the mission, with the following code:

```
_leadVehicle = _this select 3;  
_leadVehicle call fn_vbsCon_pauseConvoy
```

5. Create a `resume.sqf` file in the mission, with the following code:

```
_leadVehicle = _this select 3;  
_leadVehicle call fn_vbsCon_resumeConvoy
```

6. Preview the mission and use `Pause convoy` and `Resume convoy` user actions to pause / resume the convoy movement.

The convoy pauses / resumes its movement when the user action resumes.

## 6.3 Player Units in Convoy

You can transport player units in a Control AI convoy. The player can have different roles in a convoy vehicle:

- [Player Start as Driver \(below\)](#)
- [Player Entry as Cargo \(below\)](#)
- [Player Entry as Gunner or Commander \(on the next page\)](#)

### NOTE

Switching positions in Control AI convoy vehicles at runtime is not fully supported.

### Player Start as Driver

Players can start the mission as drivers in Control AI convoys.

#### Follow these steps:

1. In the Editor (Prepare Mode), place several vehicles on the map using the **(F4) Vehicle** Editor Object in the Editor Objects List, and make sure to select **Player** or **Playable** (if you want to allow Administrator / Instructor players to switch to Control AI drivers at mission runtime) in the Object Properties dialog for each vehicle.
2. Follow the process from Step 2 onwards in the [Fully Autonomous Convoy \(on page 94\)](#) section.

Players start the mission as convoy drivers.

### Player Entry as Cargo

Players can enter Control AI convoy vehicles as cargo at mission runtime.

#### Follow these steps:

1. Set up a convoy as described in the [Fully Autonomous Convoy \(on page 94\)](#) section.
2. Set the name of the lead convoy vehicle to `leadVehicle`.

### NOTE

Alternatively, you can use any other vehicle in the convoy. Make sure there are unoccupied seats in some of the convoy vehicles.

3. The goal is to keep the convoy paused at the beginning - so players have time to enter some of the vehicles. To achieve this, do one of the following:
  - Unlink the convoy waypoint from the group until the players are in the cargo.
  - Use one of the three [Pause / Resume Convoy \(on page 96\)](#) techniques.
4. Players can now get in the convoy vehicles using the IWF menu.

The players are in the convoy as cargo.

### Player Entry as Gunner or Commander

Players can enter Control AI convoy vehicles as gunners or commanders at mission runtime.

To set up a scenario where players are gunners or commanders in vehicles, use the same setup as with players as cargo.

#### Follow these steps:

1. Create a convoy that can be paused and pause it.
2. Players in the mission can enter the vehicles using the IWF interface (see Interact with Vehicles Interface (IWF) in the VBS4 Trainee Manual) and occupy any position in the vehicle (except the driver position), including gunner and commander positions.

#### **i** NOTE

If a position is already occupied by an AI unit, the unit is removed from the vehicle, according to the vehicle entry rules in VBS4. Control AI drivers can be created next to empty vehicles and moved into the these vehicles on mission start, using SQF commands. This prevents some positions from being already occupied.

3. Resume the convoy once all the players are inside the vehicles.

The players are in the convoy as gunners / commanders.

**Image-37: The IWF interface in one of the convoy vehicles**



## 6.4 Convoy SQF Functions

The SQF functions that pause / resume convoy movement are:

- [fn\\_vbsCon\\_pauseConvoy](#) (below)
- [fn\\_vbsCon\\_resumeConvoy](#) (on the next page)

### 6.4.1 fn\_vbsCon\_pauseConvoy

Pauses convoy movement.

The pause is executed as follows:

1. Convoy lead vehicle stops.
2. Other convoy vehicles continue driving until they reach the correct spacing distance from the vehicle before them.

#### NOTE

Gunner behavior remains unchanged. If the convoy reacts to lost vehicles, the reaction is completed before the convoy stops.

#### Syntax:

```
vehicle call fn_vbsCon_pauseConvoy
```

#### Alternative Syntax:

```
convoyGroup call fn_vbsCon_pauseConvoy
```

#### Parameters:

- vehicle: Object - Lead convoy vehicle.
- convoyGroup: Group - Convoy vehicle group.

#### Return Values:

Nothing



#### EXAMPLE

```
// Pauses convoy maneuver assigned to the group of vehicle leadVehicle1  
(group leadVehicle1) call fn_vbsCon_pauseConvoy  
  
// The same as the previous example, but the group is selected automatically  
leadVehicle1 call fn_vbsCon_pauseConvoy
```

## 6.4.2 fn\_vbsCon\_resumeConvoy

Resumes convoy movement after a pause.

The movement has to be paused using [fn\\_vbsCon\\_pauseConvoy](#) (on the previous page), otherwise, the function does nothing.

### Syntax:

```
vehicle call fn_vbsCon_resumeConvoy
```

### Alternative Syntax:

```
convoyGroup call fn_vbsCon_resumeConvoy
```

### Parameters:

- vehicle: Object - Lead convoy vehicle.
- convoyGroup: Group - Convoy vehicle group.

**Return Values:** Nothing



### EXAMPLE

```
// Resumes convoy maneuver assigned to the
(group leadVehicle1) call fn_vbsCon_resumeConvoy

// The same as the previous example, but the group
leadVehicle1 call fn_vbsCon_resumeConvoy
```

## 6.5 Convoy Order

Assigns a convoy movement behavior to a group of vehicles. The convoy destination is the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#).

For convoy uses cases, see [Convoy AI \(on page 93\)](#).

**Image-38: Convoy Order settings**



### Follow these steps:

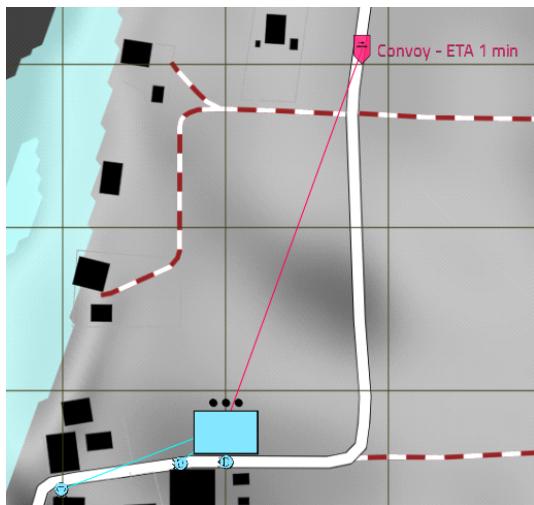
1. Select the [\(F3\) Waypoints Editor Object](#) from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Convoy**.
3. Set the [Convoy Settings \(on the next page\)](#).
4. Set the [Waypoint Completion Settings \(on page 108\)](#).
5. Click **OK** to confirm.
6. Once the Convoy Order is configured, you need to link it (see [Waypoints \(on page 23\)](#) for more information) to the lead convoy vehicle.

The Convoy Order behavior is set up.

While the convoy is moving, an ETA is displayed next to the Editor Object on the map for the current order. If the convoy stops (for instance during an engagement), the information about the stop is indicated instead of the ETA:

**NOTE**

Waypoint name types and any waypoint status information is only displayed, if the waypoint is selected.



## Convoy Settings

Setting	Description
<b>Vehicle Spacing (m)</b>	Desired spacing (in meters) between the vehicles in the convoy.
<b>Movement Speed (km/h)</b>	Suggested travel speed (in km/h) of the convoy (limited by the vehicle with the smallest maximal speed and / or the smallest acceleration).
<b>On Visual Contact</b>	Controls the reaction when spotting the enemy. <ul style="list-style-type: none"> <li><b>Continue, Hold Fire</b> (default) - Convoy continues moving towards the destination and gunners do not fire at the enemy.</li> <li><b>Continue, Open Fire</b> - Convoy continues moving towards the destination and gunners open fire at enemy entities.</li> </ul>
<b>On Taking Fire</b>	Controls the reaction to incoming enemy fire (defined by shots impacting or passing near the convoy at a short distance) - has a higher priority than <b>On Visual Contact</b> . <ul style="list-style-type: none"> <li><b>Continue, Hold Fire</b> - Convoy continues moving towards the destination and gunners do not fire at enemy entities.</li> <li><b>Continue, Open Fire</b> (default) - Convoy continues moving towards the destination and gunners open fire at enemy entities.</li> <li><b>Halt Until Clear, Open Fire</b> - Convoy stops while the gunners open fire at the enemy. The convoy automatically resumes its movement as the engagement is concluded.</li> </ul>

Setting	Description
<b>Road Usage</b>	<p>Controls whether the convoy should use roads for moving.</p> <ul style="list-style-type: none"><li>• <b>Use Only Roads</b> - Drives in the middle of the road, without respecting road lanes or directions.</li><li>• <b>Use Only Roads, Respect Lanes</b> - Drives on the road, respecting road lanes and directions.</li><li>• <b>Ignore Roads</b> - Ignores roads and moves directly to the waypoint.</li><li>• <b>Prefer Roads</b> - Prefers to move on the road, but can move off-road to bypass obstacles, take shortcuts, or when otherwise required.</li></ul>
	<b>NOTE</b>
	<p>It is necessary to select <b>Use Only Roads, Respect Lanes</b>, if you want to use military convoys together with civilian vehicles (see <a href="#">Define Traffic Flows (on page 151)</a>). Only convoys respecting lanes can encounter civilian vehicles on the same road, without the risk of collision.</p>
	<p>Off-road convoys cannot drive through areas with a high density of obstacles.</p>
	<b>TIP</b>
	<p>For multi-lane traffic, if <b>Use Only Roads, Respect Lanes</b> is used, you can either use the <b>AI Debug</b> option in VBS Geo (see <a href="#">VBS Geo User Interface in the VBS Geo Manual</a>), or the <b>Show Roads</b> and <b>Generate Roads</b> options in the <b>AI Debug Panel</b> (see <a href="#">Control AI Visualization (on page 210)</a>), to see how your road network is set up. Also, to configure road lanes, see <b>Lanes</b> in <a href="#">Placing and Editing Roads in the VBS Geo Manual</a>.</p>
<b>Formation</b>	<p>Used to define the convoy formation.</p> <p>The available formations are:</p> <ul style="list-style-type: none"><li>• <b>Convoy</b> (parallel to the direction of the convoy movement)</li><li>• <b>Staggered Column</b></li><li>• <b>Line</b> (perpendicular to the direction of the convoy movement)</li><li>• <b>Wedge</b></li><li>• <b>Vee</b></li></ul>
<b>Variable Name</b>	In <b>Variable Name</b> , enter the Order waypoint name, which can be used in SQF scripts.

Setting	Description
<b>Advanced</b>	<p>The advanced settings are:</p> <ul style="list-style-type: none"><li>• <b>Burst</b> - Sets the number of rounds fired in each burst by the convoy gunners during an engagement.</li><li>• <b>Dispersion (degrees)</b> - Sets the dispersion of convoy weapons, allowing you to decrease / increase convoy weapon accuracy.</li><li>• <b>Visibility Range (m)</b> - Can override the default line-of-sight settings for each soldier, allowing you to increase / decrease visual contact reaction distance.</li></ul> <p>Click <b>OK</b> to confirm.</p>

## Waypoint Completion Settings

Set the Order waypoint completion settings:

Option	Description
<b>Condition to Complete</b>	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
<b>Code on Completion</b>	SQF code to execute on waypoint completion.

## Aspects of Convoy Behavior

Behavior Aspect	Description
<b>Convoy Ordering</b>	The convoy vehicles are ordered automatically based on distance to destination. The vehicle closest to the destination is selected to drive on point, with the other vehicles being organized in a logical order.

Behavior Aspect	Description
Driving Towards Goal	<p>The convoy uses the following driving behavior:</p> <ul style="list-style-type: none"><li>• The lead vehicle uses the shortest path to the destination, while other vehicles follow, maintaining a distance based on <b>Vehicle Spacing (m)</b>.</li><li>• If the destination is not on the road and the convoy uses roads, the convoy drives to a point on the road that is closest to the actual destination. If the convoy ignores roads, it drives directly to the destination.</li><li>• If the destination is not on the road, the convoy drives to a point on the road that is closest to the actual destination.</li><li>• The convoy tries to reach the destination at all costs. This involves autonomous problem solving along the way (automatically dealing with disabled or destroyed vehicles). The behavior can be affected by setting reactions to contact.</li><li>• Spacing corrections - If the distance between vehicles is higher than the spacing, the vehicles in front may slow down or even stop to let the rest of the convoy catch up.</li><li>• Reaction to contact (based on the <b>On Visual Contact</b> and <b>On Taking Fire</b> settings).</li></ul>
ETA	<p>The ETA functionality is based on the following:</p> <ul style="list-style-type: none"><li>• Measured in minutes, next to the Editor Object in C2 / Editor (Execute Mode).</li><li>• ETA is paused if the convoy is paused using <a href="#">fn_vbsCon_pauseConvoy (on page 103)</a>.</li><li>• Disregards convoy stops for other reasons (for example, road blockage, vehicle losses).</li><li>• The ETA indicates an approximate time that may not be accurate if the convoy has problems driving through the terrain (due to slope or surface conditions).</li></ul>
	<b>NOTE</b>
	<p>Waypoint name types and any waypoint status information is only displayed, if the waypoint is selected.</p>
Movement Interruption	<p>Convoy movement can be interrupted under the following conditions:</p> <ul style="list-style-type: none"><li>• A convoy vehicle becomes disabled. The vehicle crew attempts to mount the nearest vehicle that is not disabled and has free seats, and the disabled vehicle is removed from the simulation after 15 seconds. A vehicle is considered disabled when:<ul style="list-style-type: none"><li>◦ It is damaged to the point of being unable to drive further.</li><li>◦ It has no Control AI driver, who is alive.</li><li>◦ It has no fuel.</li><li>◦ It is overturned.</li></ul></li><li>• Any soldier who is a member of the convoy group is dismounted.</li><li>• There is an obstacle on the road or there is a risk of collision with a moving object.</li></ul>

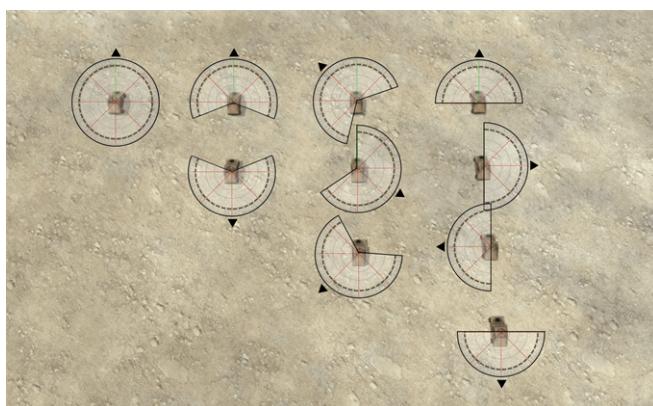
Behavior Aspect	Description
<b>Dismounted Soldier</b>	Any Control AI soldier in the convoy, who ends up dismounted for any reason (vehicle disabled, player occupying the position of the soldier, and so on), boards any other vehicle in the convoy automatically. If no vehicle has free seats, the Control AI soldier is deleted to allow the convoy to keep moving.

## Gunner Behavior

Gunner behavior applies to convoys that use the [Convoy Formation \(on page 107\)](#). In all other formations, gunners perform a 360-degree scan.

Gunner fire angles are defined according to the following convoy illustration:

**Image-39: Fire angles for convoys consisting of 1, 2, 3, and 4 vehicles**



Despite the preferred fire coverage shown in the preceding image, gunners are able to react to threats which get very close to the convoy and are in dead zones not covered by the illustrated firing angles. In such cases, fire coverage can be expanded up to 45 degrees in each direction for a short time, so that very close targets can be intercepted.

When the convoy is engaged and gunners have permission to fire (defined by the **On Visual Contact** and **On Taking Fire** settings), the gunners engage any targets in their arc of fire with bursts of fire from their turret weapon, up to their **Visibility Range (m)**. The length of the burst is determined by the **Burst** setting. When not engaging, the gunners scan their assigned arc of fire.

Gunners that control a turret with a large-caliber cannon (such as tank turrets or self-propelled artillery), or a turret only equipped with rocket or missile launchers (such as anti-air vehicles or anti-tank missile carriers) do not scan their arc of fire, but still turn the turret to engage targets.

### **NOTE**

For more than 4 vehicles, vehicles (other than first or last) alternate in the right / left firing angle, the same way as for 4 vehicles.

## 6.5.1 Troubleshooting

### Vehicles connected to a Control AI - Convoy waypoint do not move.

Check the following:

- All vehicles are in the same group.
- All vehicles have Control AI drivers (have **Control AI** specified in their object properties).
- Convoy is not intentionally stopped.
- Any of the convoy cars is not blocked by some object on the road.
- Link orientation from the [\(F3\) Waypoints Editor Object \(on page 18\)](#) to the convoy group is correct.

## 6.5.2 Limitations

Vehicles do not stop before obstacles that have their object center positioned outside of the road (for example, houses close to the road are ignored). Some rectangle objects (such as **Chernarus - Fortification > Barrier**, **HESCO**, **5 Elements**, or **Scenery - Military > Concertina wire**) have their center at the object edge, so if they are placed across the road with the center outside, vehicles can crash into them.

## 7. Bridge Laying Convoy AI

You can lay a Dry Support Bridge (DSB) with a bridge-laying convoy in VBS, using the [\(F3\) Waypoints Editor Object \(on page 18\)](#).

The bridge-laying convoy AI expands the [Convoy AI \(on page 93\)](#) use case.

For manual bridge-laying trainee operation, see Bridge Laying - HX45M in the VBS4 Trainee Manual.

The following vehicle models are required:

- **HX45M DSB** - Bridge-builder vehicle with a crane.
- **HX77 ILHS - Loaded** - Vehicle with support modules (bridge segments).

In addition, the following optional objects can be used:

- **Haulmark 3axle - Loaded** - Optional trailer, carrying support modules, that can be towed by the **HX77 ILHS - Loaded** vehicle.



### TIP

For example, you can either use 6 **HX77 ILHS - Loaded** vehicles, or 3 **HX77 ILHS - Loaded** vehicles attached to 3 **Haulmark 3axle - Loaded** trailers - the number of support modules is the same in both cases.

The simulation has the following runtime phases:

- The HX45M and HX77 vehicles drive into position, designated by the Control AI.
- The HX45M vehicle starts the bridge construction.
- The HX45M vehicle deploys the constructed bridge.



### WARNING

For autonomous DSB laying, the DSB construction and deployment area has to be spacious and clear of obstacles, due to vehicle maneuvering.

#### Follow these steps:

1. In the Editor (Prepare Mode), place one **HX45M DSB** and several **HX77 ILHS - Loaded** vehicles on the map, using **(F4) Vehicle** category in the Editor Objects List.

These vehicles are used for the bridge-laying convoy.

2. **Optional:** If you want to use Haulmark trailers with HX77 vehicles, do the following:

- a. Place one or more **Haulmark 3axle - Loaded** trailers, using the **(F8) Objects** category in the Editor Objects List, close to the **HX77 ILHS - Loaded** vehicles.
- b. Link the **Haulmark 3axle - Loaded** trailers to the **HX77 ILHS - Loaded** vehicles, using the **Trailer Hitch** option, as described in Enabling Automatic Towing in the VBS4 Editor Manual.

The Haulmark trailers are hitched to the HX77 vehicles.

3. To form the vehicles into a convoy, link the vehicles together, so that they are all in the same group (also, if necessary, add any other vehicles, such as security vehicles, from the **(F4) Vehicle** category to the convoy).

**⚠️ WARNING**

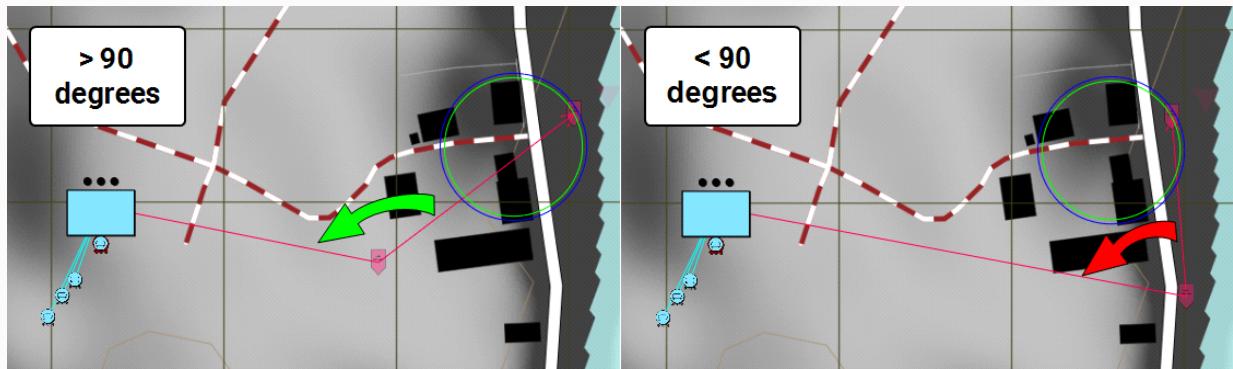
Vehicles with hitched trailers must not be group leaders.

4. Rotate and position the vehicles, as required.
5. (Optional) Create a convoy waypoint (see [Fully Autonomous Convoy \(on page 94\)](#)), if you want the vehicles to first get to the location, where the bridge should be laid.
6. Create a bridge-laying waypoint (see [Waypoints \(on page 23\)](#)) by right-clicking the bridge-laying convoy group marker, selecting **Orders > Assign New Waypoint**, and clicking a position on the map, where the bridge needs to be constructed and deployed.

**⚠️ WARNING**

Using the 2D view, it is important to place the convoy and bridge-laying waypoints in such a way, that the turn angle between the convoy waypoint and the bridge-laying waypoint is not smaller than 90 degrees, to avoid situations, where the convoy vehicles (particularly, the **HX45M DSB** and **HX77 ILHS - Loaded** vehicles) have to travel backward from the convoy waypoint in the direction of the bridge-laying waypoint.

**Image-40: Left to right: valid and invalid waypoint positioning**



- Set the bridge-laying properties (see [Deploy DSB Order \(on page 116\)](#)) and click **OK**.

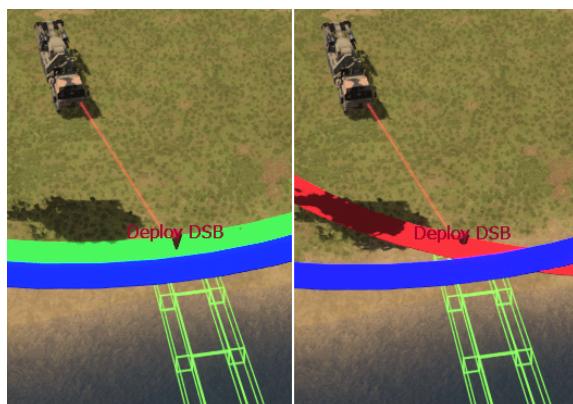
The Deploy DSB Order waypoint is created on the map.

The **Blue Circle** indicates the maneuvering space for the HX45M and HX77 vehicles.

The **Green Circle** indicates the harbor (parking space) for the HX45M and HX77 vehicles.

**NOTE**

If the offset between the **Blue Circle** (defined by the Deploy DSB Order waypoint) and the **Green Circle** (defined by the [Harbor Position \(on page 118\)](#)) is not big enough, so that the parked vehicles would interfere with construction maneuvering, the **Green Circle** turns red, and the error message [Invalid bridge placement \(on the next page\)](#) appears.

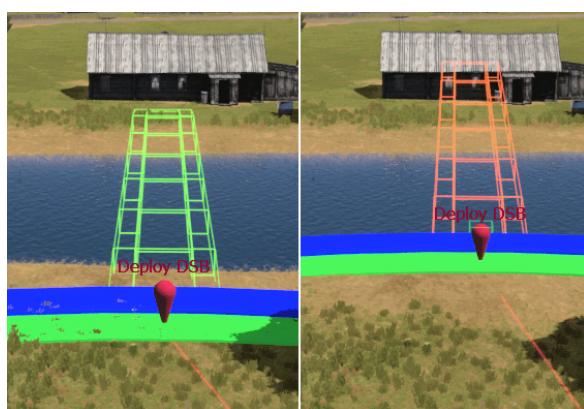


- Change the bridge deployment and construction waypoint position and / or its orientation, by rotating the waypoint.

The 3D bridge proxy appears in green / red, indicating whether they bridge can / cannot be constructed and deployed.

**WARNING**

If the red-bridge proxy error is not resolved, bridge-laying cannot be completed.



9. Preview the mission.
10. If any of the following runtime errors appear, fix them accordingly:

**NOTE**

The error messages only appear in the Editor (Execute Mode).

Error Message	Explanation and Resolution
<b>Missing HX45 Vehicle</b>	The HX45M bridge-builder vehicle, which operates the crane, is missing. Add the <b>HX45M DSB</b> vehicle to the mission.
<b>Missing HX77 Vehicle(s)</b>	There are not enough flatracks with cargo (modules), carried by the HX77 trucks and / or Haulmark trailers. Add more <b>HX77 ILHS - Loaded</b> and / or <b>Haulmark 3axle - Loaded</b> vehicles to the mission, until there are enough to <a href="#">Set the Bridge Length (in meters)</a> . <a href="#">(on the next page)</a>
<b>Invalid bridge placement</b>	The bridge and / or harbor cannot be placed at the specified position. Choose a different bridge deployment waypoint location and / or orientation and / or harbor position.

**WARNING**

If the runtime errors are not resolved, bridge-laying cannot be completed.

11. Preview the mission.

The bridge-laying convoy drives into position, starts the bridge construction, and deploys the bridge.

**Image-41: Left to right: ongoing / complete bridge construction**



## 7.1 Deploy DSB Order

Assigns a bridge-laying order to the HX45M vehicle, to deploy a Dry Support Bridge (DSB) at the given waypoint, as specified by the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map.

For more information, see [Bridge Laying Convoy AI \(on page 112\)](#).

**Image-42: Deploy DSB Order settings**



### Follow these steps:

1. Select the [\(F3\) Waypoints Editor Object](#) in the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Deploy DSB**.
3. Set the **Bridge Length** (in meters).

The following drop-down options are available:

- 46m (6 modules needed)
- 40m (5 modules needed)
- 34m (5 modules needed)
- 28m (4 modules needed)
- 22m (4 modules needed)

4. In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.
5. Set the [Waypoint Completion Settings \(on the next page\)](#).
6. Click **Advanced** to set any [Advanced Settings \(on the next page\)](#). Click **OK** to confirm.
7. Click **OK** to confirm.

8. Once the Deploy DSB Order is configured, you must link it to the HX45M vehicle:
  - a. Right-click the HX45M vehicle.
  - b. Select **Orders > Assign Existing Waypoint**.
  - c. Click the Deploy DSB Order.

The Deploy DSB Order is linked to the HX45M vehicle.

The Deploy DSB Order behavior is set up.

The HX45M and HX77 vehicles drive into position, and start constructing and deploying the bridge.

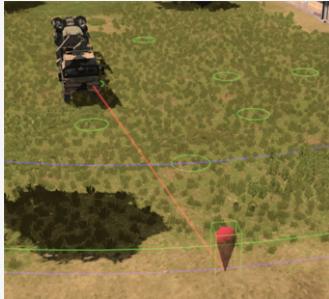
## 7.1.1 Waypoint Completion Settings

Set the Order waypoint completion settings:

Option	Description
<b>Condition to Complete</b>	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
<b>Code on Completion</b>	SQF code to execute on waypoint completion.

## 7.1.2 Advanced Settings

The Deploy DSB Order has the following advanced settings:

Setting	Description
<b>Show Debug</b>	Visualize the vehicle (HX45M and HX77) parking positions in 3D. To enable / disable, set this setting to <code>true</code> / <code>false</code> .
	
<b>Speed Up Animation</b>	Speeds up the bridge-laying animations by about 4 times the normal animation speed. To enable / disable, set this setting to <code>true</code> / <code>false</code> .
<b>Skip Vehicle Parking</b>	Skips vehicle parking animations (vehicles are teleported to their parking positions). To enable / disable, set this setting to <code>true</code> / <code>false</code> .

Setting	Description
<b>Harbor Position</b>	Defines the center of the Harbor (parking space) ASL2 position for the HX45M and HX77 vehicles. The value can be specified as an ASL2 position or as the name of an Editor Object, whose position is used.

 **TIP**

You can use any Editor Object (for example, a Marker Arrow Editor Object) to indicate the center of the harbor area.

**Follow these steps:**

1. In the Editor Objects List, select **(F8) Objects**.
2. Click a position on the map, where you want to place the harbor area.
3. Select any of the **Marker Arrow** Editor Objects, in the **VBS Objects** category.
4. In **Name**, specify the harbor area name (for example, "harbor").
5. Specify the **Harbor Position**, using the [getPosASL2](https://sqf.bisimulations.com/display/SQF/getPosASL2) (<https://sqf.bisimulations.com/display/SQF/getPosASL2>) SQF command and the Marker Arrow Editor Object:

```
getPosASL2 harbor
```

## 8. Aircraft AI

You can give Control AI rotary-wing and fixed-wing aircraft orders to follow fly to a specific location.

For land vehicles, also see the [Military AI \(on page 53\)](#), [Convoy AI \(on page 93\)](#), and [Bridge Laying Convoy AI \(on page 112\)](#).

The rotary-wing / fixed-wing aircraft order behavior is applied using the [\(F3\) Waypoints Editor Object \(on page 18\)](#). The orders create VBS4 waypoints (see [Waypoints \(on page 23\)](#)), and can use branching (see [Branching Waypoints \(on page 25\)](#)).

The following topics are discussed:

- [Aircraft Tutorial \(on the next page\)](#) - A tutorial that demonstrates the Control AI behaviors for rotary-wing UAVs.
- [Aircraft AI Orders and Behaviors \(on page 123\)](#) - A list of rotary-wing AI order behaviors.

## 8.1 Aircraft Tutorial

This tutorial demonstrates the use of Control AI behaviors for rotary-wing UAVs.

The workflow of the tutorial is as follows:

1. In Prepare mode, add a player unit and a DJI Spark UAV - see [Creating the Entities \(below\)](#).
2. Give the UAV several Fly Order waypoints (see [Fly Order \(on page 124\)](#)) to fly to - see [Creating the Flight Path \(on the next page\)](#).

### 8.1.1 Creating the Entities

First, create the entities - the player unit (to be able to run the scenario) and the DJI Spark UAV.

**Follow these steps:**

1. In the VBS Editor, select **(F1) Unit** in the Editor Objects List.
2. Double-click a location on the map to place the unit.  
The Object Properties dialog opens.
3. Select any unit, and make sure to select **Player** in the **AI Parameters** drop-down.
4. Click **OK**.

The player unit is placed on the map.

5. In the VBS Editor, select **(F4) Vehicle** in the Editor Objects List.
6. Double-click a location on the map to place the UAV.  
The Object Properties dialog opens.
7. In **Filters**, type **DJI Spark** and select the DJI Spark UAV.
8. Make sure that the **Special** drop-down is not set to **Flying**.

This ensures that the UAV is placed on a surface (such as the ground).

9. Click **OK**.  
The DJI Spark UAV is placed on the map.

The entities are placed in the scenario.

## 8.1.2 Creating the Flight Path

Add Fly Order waypoints to create the UAV flight path.

**Follow these steps:**

1. Right-click the UAV and select **Orders > Assign New Waypoint**.
2. Click the first location where you want your UAV to fly to, designated by the first Fly Order waypoint.

The Object Properties dialog opens.

3. In the **Behavior** list, select **Fly**.
4. Set **Destination Altitude (m)** to 10 meters.
5. Leave **Altitude Mode** set to AGL (Above Ground Level).
6. Leave **Maximum Speed (km/h)** at 5 meters per second.
7. Click **OK**.

The first Fly Order waypoint is created for the UAV.

8. Right-click the Fly Order waypoint and select **Assign Next Waypoint**.
9. Click the location of the next Fly Order waypoint.

The Object Properties dialog opens.

10. In the **Behavior** list, select **Fly**.
11. Set **Destination Altitude (m)** to 25 meters, leave **Altitude Mode** and **Maximum Speed (km/h)** set to their previous values, and click **OK**.

The second Fly Order waypoint is created for the UAV.

12. Repeat steps 8 - 11 for the second Fly Order waypoint, but set **Destination Altitude (m)** to 10 meters.

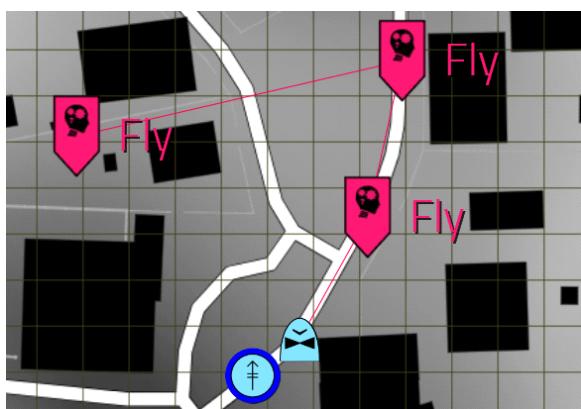
The third Fly Order waypoint is created.

The UAV flight path is created.

13. Click **Preview** to preview the scenario.

The UAV first ascends to 10 meters at the first waypoint, then to 25 meters at the second waypoint, and then descends back to 10 meters at the third waypoint, above which it stays hovering.

**Image-43: Fly Order waypoints in the 2D View**



**Image-44: Fly Order waypoints in the 3D View at different altitudes**



A yellow altitude line indicator appears in the 3D View at the Fly Order waypoint position.



## 8.2 Aircraft AI Orders and Behaviors

Rotary-wing and fixed-wing aircraft AI uses the [\(F3\) Waypoints Editor Object \(on page 18\)](#) and consists of the following orders / behaviors:

Order / Behavior Type	Description
Fly Order (on the next page)	The single fixed-wing or rotary-wing aircraft flies to the given position (the location of the <a href="#">(F3) Waypoints Editor Object (on page 18)</a> on the map).
Land Order (on page 127)	The single rotary-wing aircraft lands at the given position (the location of the <a href="#">(F3) Waypoints Editor Object (on page 18)</a> on the map).
Loiter Order (on page 129)	The single fixed-wing aircraft loiters around the given position (the location of the <a href="#">(F3) Waypoints Editor Object (on page 18)</a> on the map).

In addition, aircraft AI uses the following [Military AI \(on page 53\)](#) orders / behaviors:

 **NOTE**

These orders / behaviors can only be used for rotary-wing aircraft.

Order / Behavior Type	Description
Mount Order (on page 82)	The infantry team moves to the given position (the location of the <a href="#">(F3) Waypoints Editor Object (on page 18)</a> on the map), and mounts the nearest vehicle at the selected vehicle positions (crew, cargo, or both).
Dismount Order (on page 86)	The infantry teams dismounts the vehicle from the selected vehicle positions (crew, cargo, or both).

 **NOTE**

VBS Plan (see VBS Plan Overview in the VBS Plan Manual) relies on the rotary-wing and fixed-wing aircraft AI behaviors. For more information on how to use them, see Order Tactical Tools in Tactical Objects in the VBS Plan Manual.

## 8.2.1 Fly Order

The single fixed-wing or rotary-wing aircraft flies to the given position (the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map).

To order the aircraft to loiter around a given position or land at it, see the [Loiter Order \(on page 129\)](#) and [Land Order \(on page 127\)](#).

### NOTE

The following considerations apply:

- The Fly Order cannot be used with omnicopters.
- The Fly Order can only be used with single aircraft, not aircraft groups.
- The **Weapon Control Status** is ignored for fixed-wing aircraft.

**Image-45: Fly Order settings**



**Follow these steps:**

1. Select the **(F3) Waypoints** Editor Object from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Fly**.

3. In the **Weapon Control Status** drop-down, select the behavior for engaging enemy forces, when they are encountered:

Option	Description
No Change	No change in the behavior upon encountering enemy forces.
Weapons Free	Fire at enemy forces, when they are encountered.
Hold Fire	Do not fire at enemy forces, when they are encountered.

 **NOTE**

The Weapon Control Status is related to the following SQF commands:

- [combatMode](https://sqf.bisimulations.com/display/SQF/combatMode) (<https://sqf.bisimulations.com/display/SQF/combatMode>)
- [setCombatMode](https://sqf.bisimulations.com/display/SQF/setCombatMode) (<https://sqf.bisimulations.com/display/SQF/setCombatMode>)
- [unitCombatMode](https://sqf.bisimulations.com/display/SQF/unitCombatMode) (<https://sqf.bisimulations.com/display/SQF/unitCombatMode>)
- [setUnitCombatMode](https://sqf.bisimulations.com/display/SQF/setUnitCombatMode)  
(<https://sqf.bisimulations.com/display/SQF/setUnitCombatMode>)

4. In **Destination Altitude (m)**, set the altitude (in meters) at which the aircraft should hover above the destination position, after reaching it.

 **WARNING**

The VBS Editor Undo and Redo operations do not take into account altitude changes in Fly Order waypoints.

 **NOTE**

Wind (see Weather Settings in the VBS4 Editor Manual) does not affect aircraft, piloted by Control AI.

5. In the **Altitude Mode** drop-down, select between AGL (Above Ground Level) and ASL (Above Sea Level) altitude modes for **Destination Altitude (m)**.
6. In **Maximum Speed (km/h)**, set the maximum speed (in km/h) at which the aircraft should fly to the destination position.
7. In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.

8. Set the Order waypoint completion settings:

Option	Description
<b>Condition to Complete</b>	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
<b>Code on Completion</b>	SQF code to execute on waypoint completion.

9. Click **OK**.

The Fly Order behavior is set up.

 **WARNING**

Bohemia Interactive Simulations advises scenario Administrators not to change the **Advanced** settings for the behavior to work correctly.

A yellow altitude line indicator appears in the 3D View at the Fly Order waypoint position.



## 8.2.2 Land Order

The single rotary-wing aircraft lands at the given position (the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map).

### NOTE

The following considerations apply:

- The Land Order can only be used with rotary-wing aircraft.
- The Land Order can only be used with single aircraft, not aircraft groups.
- For more realistic behavior, it is recommended to assign a [Fly Order \(on page 124\)](#) to the aircraft, before assigning it a Land Order.

**Image-46: Land Order settings**



### Follow these steps:

1. Select the [\(F3\) Waypoints Editor Object](#) from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Land**.

3. In the **Weapon Control Status** drop-down, select the behavior for engaging enemy forces, when they are encountered:

Option	Description
No Change	No change in the behavior upon encountering enemy forces.
Weapons Free	Fire at enemy forces, when they are encountered.
Hold Fire	Do not fire at enemy forces, when they are encountered.

 **NOTE**

The Weapon Control Status is related to the following SQF commands:

- [combatMode](https://sqf.bisimulations.com/display/SQF/combatMode) (<https://sqf.bisimulations.com/display/SQF/combatMode>)
- [setCombatMode](https://sqf.bisimulations.com/display/SQF/setCombatMode) (<https://sqf.bisimulations.com/display/SQF/setCombatMode>)
- [unitCombatMode](https://sqf.bisimulations.com/display/SQF/unitCombatMode) (<https://sqf.bisimulations.com/display/SQF/unitCombatMode>)
- [setUnitCombatMode](https://sqf.bisimulations.com/display/SQF/setUnitCombatMode)  
(<https://sqf.bisimulations.com/display/SQF/setUnitCombatMode>)

4. In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.

5. Set the Order waypoint completion settings:

Option	Description
Condition to Complete	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
Code on Completion	SQF code to execute on waypoint completion.

6. Click **OK**.

The Land Order behavior is set up.

 **WARNING**

Bohemia Interactive Simulations advises scenario Administrators not to change the **Advanced** settings for the behavior to work correctly.

## 8.2.3 Loiter Order

The single fixed-wing aircraft loiters around the given position (the location of the [\(F3\) Waypoints Editor Object \(on page 18\)](#) on the map).

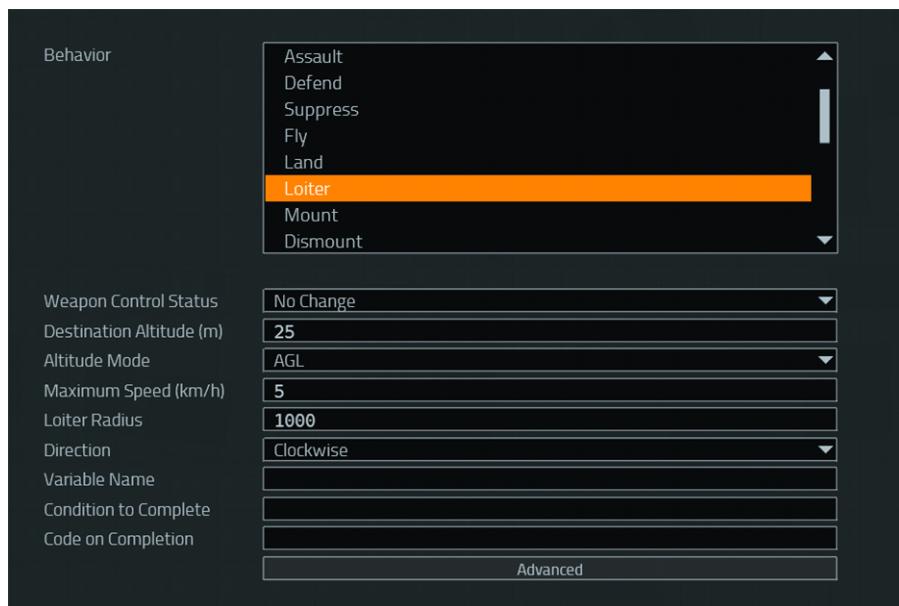
To land an aircraft, see the [Land Order \(on page 127\)](#).

### NOTE

The following considerations apply:

- The Loiter Order can only be used with fixed-wing aircraft.
- The Loiter Order can only be used with single aircraft, not aircraft groups.
- The **Weapon Control Status** is ignored for fixed-wing aircraft.
- Setting the **Special** drop-down to **Flying** (see Adding Vehicles in the VBS4 Editor Manual) creates a Loiter Order at the aircraft position, with a 5 km radius.

**Image-47: Loiter Order settings**



**Follow these steps:**

1. Select the **(F3) Waypoints** Editor Object from the Editor Objects List, and place it on the map.
2. In the **Behavior** list, select **Loiter**.

3. In the **Weapon Control Status** drop-down, select the behavior for engaging enemy forces, when they are encountered:

Option	Description
No Change	No change in the behavior upon encountering enemy forces.
Weapons Free	Fire at enemy forces, when they are encountered.
Hold Fire	Do not fire at enemy forces, when they are encountered.

 **NOTE**

The Weapon Control Status is related to the following SQF commands:

- [combatMode](https://sqf.bisimulations.com/display/SQF/combatMode) (<https://sqf.bisimulations.com/display/SQF/combatMode>)
- [setCombatMode](https://sqf.bisimulations.com/display/SQF/setCombatMode) (<https://sqf.bisimulations.com/display/SQF/setCombatMode>)
- [unitCombatMode](https://sqf.bisimulations.com/display/SQF/unitCombatMode) (<https://sqf.bisimulations.com/display/SQF/unitCombatMode>)
- [setUnitCombatMode](https://sqf.bisimulations.com/display/SQF/setUnitCombatMode)  
(<https://sqf.bisimulations.com/display/SQF/setUnitCombatMode>)

4. In **Destination Altitude (m)**, set the altitude (in meters) at which the aircraft should loiter above the destination position, after reaching it.

 **WARNING**

The VBS Editor Undo and Redo operations do not take into account altitude changes in Loiter Order waypoints.

 **NOTE**

Wind (see Weather Settings in the VBS4 Editor Manual) does not affect aircraft, piloted by Control AI.

5. In the **Altitude Mode** drop-down, select between AGL (Above Ground Level) and ASL (Above Sea Level) altitude modes for **Destination Altitude (m)**.
6. In **Maximum Speed (km/h)**, set the maximum speed (in km/h) at which the aircraft should loiter around the destination position.
7. In **Loiter Radius** (meters), set the loiter radius around the destination position.
8. In the **Direction** drop-down, select the loitering direction (clockwise or counterclockwise).
9. In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.

10. Set the Order waypoint completion settings:

Option	Description
<b>Condition to Complete</b>	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
<b>Code on Completion</b>	SQF code to execute on waypoint completion.

11. Click **OK**.

The Loiter Order behavior is set up.

 **WARNING**

Bohemia Interactive Simulations advises scenario Administrators not to change the **Advanced** settings for the behavior to work correctly.

## 9. Civilian AI

Control AI allows you to create a civilian pattern of life, using pedestrians and car traffic, where civilians engage in various activities in the given area and respond to various events in various ways.

The general workflow for defining a civilian pattern of life is as follows:

1. Define the population (pedestrians and car traffic).

For more information, see [Define Populations \(on the next page\)](#).

2. Define the civilian pedestrian flow that consists of idle activities, performed in parallel by civilian entity groups, and mandatory activities, performed in a sequence by each civilian entity as part of their simulation lifecycle.

For more information, see [Define Pedestrian Flows \(on page 141\)](#).

3. Define the car traffic flow, using traffic lights and areas.

For more information, see [Define Traffic Flows \(on page 151\)](#).

4. Define responsive behavior, when an event happens, causing a temporary or permanent disruption in idle and mandatory-sequence activities.

For more information, see [Defining Responsive Behavior \(on page 163\)](#).

5. To troubleshoot your civilian population behavior, use the civilian Control AI visualization options.

For more information, see [Debug Visualization \(on page 168\)](#).

In addition, a train-station example scenario, which demonstrates pedestrian, traffic, and responsive civilian behaviors, is available: [Basic Example Scenario \(on page 172\)](#).

For additional civilian AI demo scenarios, see [Example Content \(on page 230\)](#).

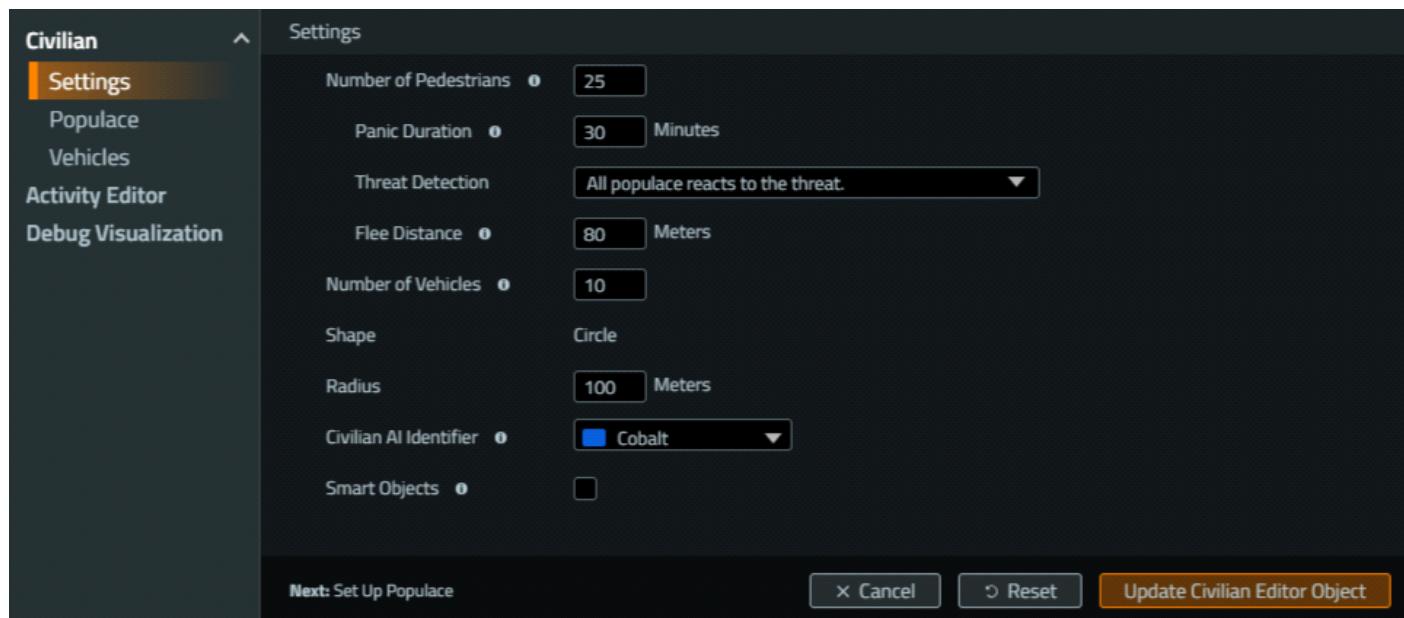
For a full list of civilian Editor Objects used by the civilian AI, see [Civilian AI Editor Objects \(on page 188\)](#).

## 9.1 Define Populations

To create a pattern of life, first define a population, consisting of pedestrians and / or car traffic.

In the VBS Editor Objects List, select the **Control AI - Civilian** Editor Object and place it on the map, where you want to simulate the civilian pattern of life.

The Civilian AI dialog opens.



### Follow this process:

1. Specify Civilian Settings (on the next page)
2. Specify Civilian Populace (on page 136)
3. Specify Vehicle Traffic (on page 138)
4. Click **Place Civilian Editor Object** to place the population on the map.

The [Population Editor Object \(on page 189\)](#) is created and placed on the map, and you can define activities for your population.

For more information on defining activities, see [Define Pedestrian Flows \(on page 141\)](#) and [Define Traffic Flows \(on page 151\)](#).

If you want to edit an existing population, see [Edit Populations \(on page 140\)](#).

 **TIP**

Before you define the population, you can place Smart Objects, which are special Editor Objects for the population to interact with, using automatically-defined population activities. If you decide to do so, check the [Smart Objects \(on the next page\)](#) setting in step 3 of this procedure.

Alternatively, define the population first, and add Smart Objects later. However, you must manually define the activities for these Editor Objects. For more information, see [Define Pedestrian Flows \(on page 141\)](#).

**EXAMPLE**

The [Ticket Machine \(on page 201\)](#) is a Smart Object that requires a relevant activity for civilians (**Buy Item (Vending Machine)**), see [Define Pedestrian Flows \(on page 141\)](#)).

## 9.1.1 Specify Civilian Settings

In the Civilian AI dialog, expand the **Civilian** menu and select **Settings**.

Setting	Description
<b>Number of Pedestrians</b>	Number of pedestrians in the population.
<b>Panic Duration</b>	Panic duration (if panic ensues, due to a threat) (in minutes).
<b>Threat Detection</b>	Defines how panic starts, based on the detected threat. Has the following options: <ul style="list-style-type: none"><li><b>All populace reacts to the threat</b></li><li><b>Only populace within hearing range reacts to threats</b></li><li><b>Only populace within hearing range or line of sight reacts to threats</b></li></ul>

**NOTE**

If Panic Duration is less than the amount of time it takes civilians to get to a Safe Zone (see [Define Pedestrian Flows \(on page 141\)](#)), the civilians stop panicking before reaching the Safe Zone.

Setting	Description
<b>Flee Distance</b>	Defines the distance (in meters) from the threat to which pedestrians flee in panic.
	<div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"><p><b>NOTE</b></p><p>If Flee Distance is smaller than the distance to a Safe Zone (see <a href="#">Define Pedestrian Flows (on page 141)</a>), some civilians hide in the Safe Zone, while others only rely on Flee Distance.</p></div>
<b>Number of Vehicles</b>	Number of vehicles in the population.
	<div style="border: 2px solid red; padding: 10px; margin-top: 10px;"><p><b>WARNING</b></p><p>If you add vehicles to your civilian population (<b>Number of Vehicles</b> is greater than 0), you must define <b>Vehicle Spawn</b> and <b>Vehicle Despawn</b> activities. Also, <b>Vehicle Spawn</b> and <b>Vehicle Despawn</b> activities must be placed on roads. Placing them elsewhere does not create vehicle traffic. For more information, see <a href="#">Spawn / Despawn (on page 195)</a>.</p></div>
<b>Shape</b>	Population area shape on the map. Only <b>Circle</b> is available.
<b>Radius</b>	Radius of the population area (in meters).
<b>Civilian AI Identifier</b>	Population color identifier. Different populations can have different roles, and color identifiers are used to distinguish between these roles.
	<div style="border: 1px solid green; padding: 10px; margin-top: 10px;"><p><b>TIP</b></p><p>Click the <b>Information</b> icon to see how populations only perform activities defined using the same color identifier (see <a href="#">Define Pedestrian Flows (on page 141)</a>).</p></div>
<b>Smart Objects</b>	Check this to automatically create relevant population activities for any Smart Objects placed on the map. For a full list of Smart Objects, see <a href="#">Smart Objects (on page 200)</a> .
	<div style="border: 2px solid red; padding: 10px; margin-top: 10px;"><p><b>WARNING</b></p><p>To automatically create relevant population activities for your Smart Objects, make sure to place them first, before creating the population.</p></div>

## 9.1.2 Specify Civilian Populace

In the Civilian menu, select **Populace**, to set up the pedestrian entities:



Click the **Add Model** icon to add a model, which is used to generate the pedestrian population.



### TIP

The more pedestrian models you add, the more realistic the population looks.

**Image-48: Add Pedestrian Models dialog**



### WARNING

Child entity models do not perform any pedestrian activities, apart from the **Start Panic** activity. For more information on pedestrian activities, see [Define Pedestrian Flows \(on page 141\)](#).

## Follow these steps:

1. Use the **Filter** and **Search** fields to find a specific pedestrian model you want to insert.

 **NOTE**

The search results are based on the selected filter.

2. In the **Library** list, select the models you want to insert, and click the **Right Arrow** icon.

 **TIP**

The **Preview** box illustrates how the selected pedestrian model looks.



The model is added to the **My Populace** list.

To remove a model from the My Populace list, select the model and click the **Left Arrow** icon.



3. To confirm the added models, select **Apply Populace Changes**.

Alternatively, click **Reset** to revert the pedestrian model changes.

To delete a model, select the model row and click the **Trash** icon.

 **TIP**

To delete multiple models, use the **Type** column checkbox.



Use the sliders in the **Ratio** column to control the model ratio in the pedestrian population (how many pedestrians are generated, using the model, as a percentage of the pedestrian population).

Click the **Lock** icon to lock / unlock the ratio (if locked, the ratio slider cannot be modified).



## 9.1.3 Specify Vehicle Traffic

In the Civilian menu, select **Vehicles**, to set up the vehicle entities:

### **⚠️ WARNING**

Unlike pedestrians, vehicles require at least one **Vehicle Spawn** activity and one **Vehicle Despawn** activity to appear in the scenario. For more information on how to add activities, see [Create Activities \(on page 142\)](#).



Click the **Add Vehicle** icon to add a model, which is used to generate the vehicle population.



### **✓ TIP**

The more vehicle models you add, the more realistic the population looks.

**Image-49: Add Vehicle Models dialog**



## Follow these steps:

1. Use the **Filter** and **Search** fields to find a specific vehicle model you want to insert.

 **NOTE**

The search results are based on the selected filter.

2. In the **Library** list, select the models you want to insert, and click the **Right Arrow** icon.

 **TIP**

The **Preview** box illustrates the selected vehicle model.



The model is added to the **My Vehicles** list.

To remove a model from the My Vehicles list, select the model and click the **Left Arrow** icon.



3. To confirm the added models, select **Apply Vehicles Changes**.

Alternatively, click **Reset** to revert the vehicle model changes.

To delete a model, select the model row and click the **Trash** icon.

 **TIP**

To delete multiple models, use the **Type** column checkbox.



Use the sliders in the **Ratio** column to control the model ratio in the vehicle population (how many vehicles are generated, using the model, as a percentage of the vehicle population).

Click the **Lock** icon to lock / unlock the ratio (if locked, the ratio slider cannot be modified).



## 9.1.4 Edit Populations

You can edit an existing population.

**Follow these steps:**

1. Double-click the [Population Editor Object \(on page 189\)](#) on the map.

The Civilian AI dialog opens.

2. Follow the previous procedures, to make the required modifications:

- [Specify Civilian Settings \(on page 134\)](#)
- [Specify Civilian Populace \(on page 136\)](#)
- [Specify Vehicle Traffic \(on page 138\)](#)

3. Click **Update Control AI - Civilian Editor Object** to save the population changes.

The population is modified.

## 9.2 Define Pedestrian Flows

Once the civilian population is defined (see [Define Populations \(on page 133\)](#)), you can define the pedestrian activity flow.

The process to create a pedestrian flow is as follows:

1. Make sure that the [Number of Pedestrians \(on page 134\)](#) to generate is defined (bigger than 0), as well as the pedestrian models (see step 4 of [Define Populations \(on page 133\)](#)).
2. Create activities and areas for your pedestrian crowd.

 **NOTE**

If the activities are area-specific, you can create areas for them. For some activity types, area linking is mandatory (this is done automatically - see [Create Area \(on page 194\)](#)), while for other types it is not (activities that have no areas defined occur at the activity position, or as close to the position as possible, on the map).

For more information, see:

- [Create Activities \(on the next page\)](#)
- [Create Areas \(on page 144\)](#)

For more information on how to define responsive behavior for various events, see [Defining Responsive Behavior \(on page 163\)](#).

3. If you want to modify existing activities and / or areas, you can edit them. See:

- [Edit Activities \(on page 146\)](#)
- [Edit Areas \(on page 147\)](#)

In addition, you can define riot behavior for civilian groups. See [Define Civilian Riot \(on page 148\)](#).

For how to define the traffic flow, see [Define Traffic Flows \(on page 151\)](#).

## 9.2.1 Create Activities

You can create an activity for your population.

**Follow these steps:**

1. In the Editor Objects List, select **Control AI - Activity** and double-click a location on the map, where you want to place the activity.

### **WARNING**

If you place the activity outside the population radius (see [Radius \(on page 135\)](#)), it is not functional. It only becomes functional, if you move it within the population radius.

The Activity Creator opens.



2. In the **Activity Creator**, set the activity **Name**.

### **NOTE**

The activity **Name** setting cannot be used as an Editor Object variable name.

3. Set the [Activity Types \(on page 194\)](#) settings.
4. Set the remaining [Activity Settings \(on page 192\)](#).
5. Check **Create Area** to create an area, where the activity takes place.

### **NOTE**

For activities that require the presence of an area, this option is automatically checked.

This creates an area ([Area Editor Object \(on page 199\)](#)) with the default size of 25 meters x 25 meters (to change the area size, see [Edit Areas \(on page 147\)](#)), linked to the [Activity Editor Object \(on page 192\)](#).

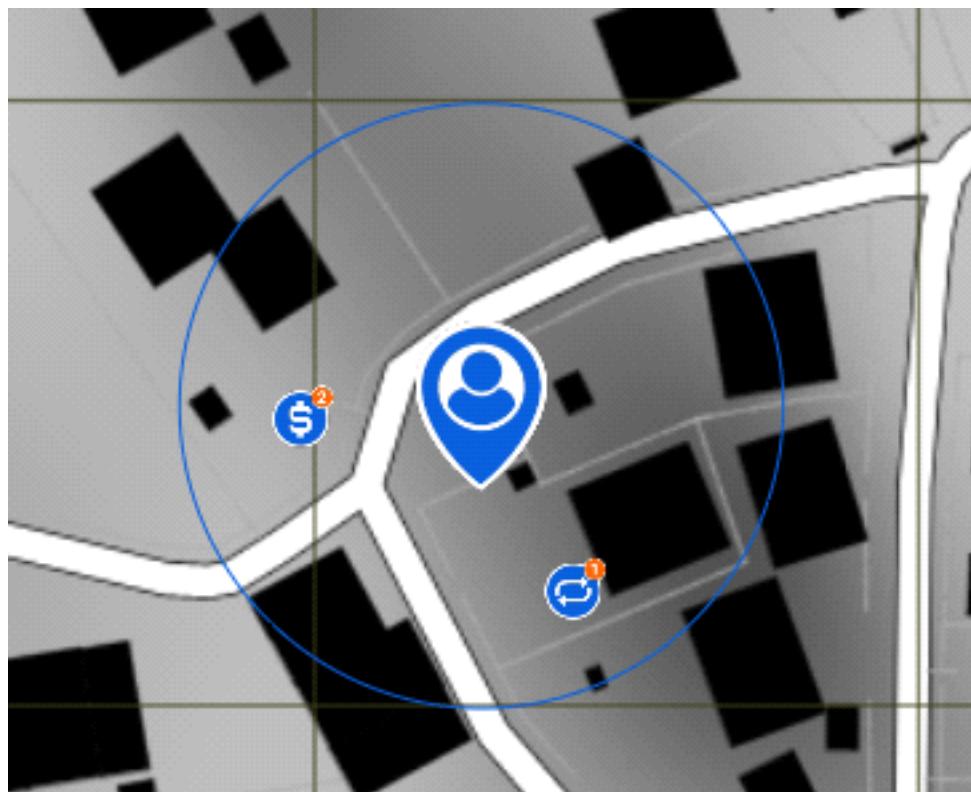
Alternatively, you can create the activity and the area separately, and then link the former to the latter. For more information, see [Create Areas \(on page 144\)](#).

6. Click **Place Activity**.

7. (Optional) Add additional interactive objects for your pedestrians. For more information, see [Smart Objects \(on page 200\)](#).

The [Activity Editor Object \(on page 192\)](#) is created and placed on the map.

**Image-50: Population with two activities**



You can create more activities in the same way, or edit existing ones (see [Edit Activities \(on page 146\)](#)).

## 9.2.2 Create Areas

You can create an area, within the civilian pattern of life area.

### **i** NOTE

If you create an area without an activity, the created area is an abstract area, represented by the [Area Editor Object \(on page 199\)](#), and has no purpose, until an [Activity Editor Object \(on page 192\)](#) is linked to it, which gives the area its purpose.

**Follow these steps:**

1. In the Editor Objects List, select **Control AI - Area** and double-click a location on the map, where you want to place the area.



### **WARNING**

If you place the area outside the population radius (see [Radius \(on page 135\)](#)), it is not functional. It only becomes functional, if you move it within the population radius.

2. In the **Object Properties** dialog, set the following:

Area Setting	Description
<b>Size (Left-Right)</b>	Area width (in meters).
<b>Size (Up-Down)</b>	Area height (in meters).

3. Click **OK**.

The [Area Editor Object \(on page 199\)](#) is created and placed on the map.



#### 4. Linking an activity to an area, or attaching an area to an entity:

- To link an activity to an area (for example, for an activity to be performed in a certain area), right-click an [Activity Editor Object \(on page 192\)](#), select **Link to Area** in the context menu, and click the [Area Editor Object \(on page 199\)](#).

The area changes color and shows the linked activity icon.

To unlink the activity from the area, repeat this step.

 **WARNING**

You can only link one activity to an area.

- You can attach the area to an entity, such as a human or a vehicle. To do that, right-click the [Area Editor Object \(on page 199\)](#), select **Attach to Unit or Vehicle** in the context menu, and click the entity.

 **WARNING**

Trying to attach **Restrict Access (Pedestrians)** and **Restrict Access (Vehicles)** areas to moving entities may result in unexpected behavior, as areas of these types cannot move with the entity.

The area is attached to the entity.

 **NOTE**

You can attach multiple areas to an entity.

To detach the area from the entity, repeat this step.

To edit an existing area, see [Edit Areas \(on page 147\)](#).

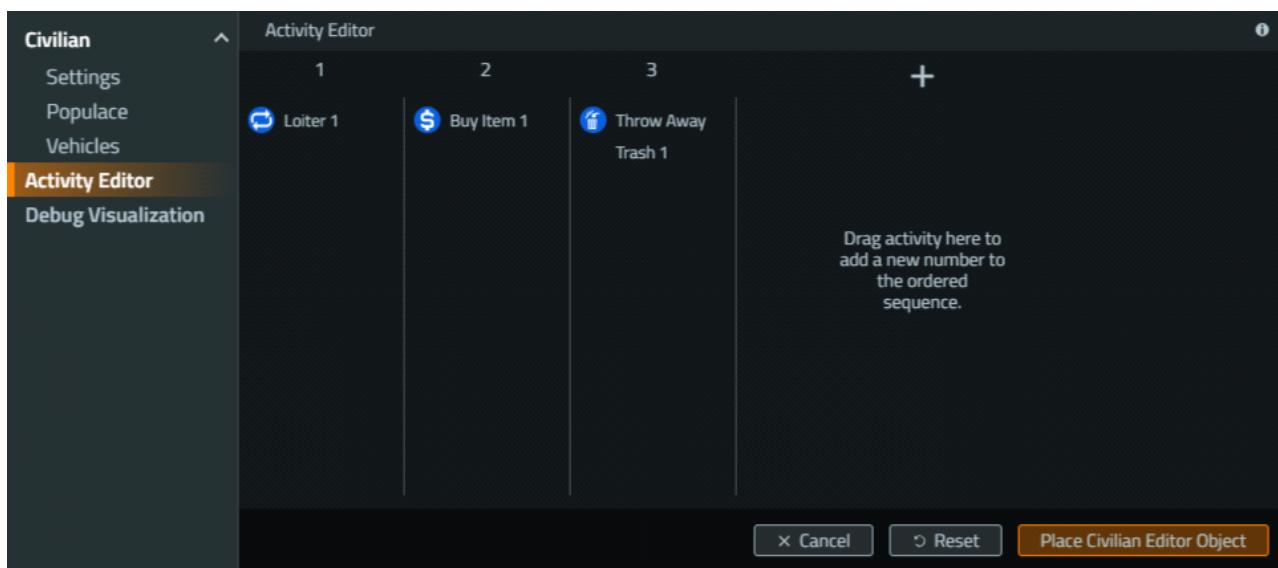
## 9.2.3 Edit Activities

You can edit existing population activities.

### Follow these steps:

Do any of the following:

- To edit the activity settings:
  1. Double-click the [Activity Editor Object \(on page 192\)](#).  
The Activity Creator opens.
  2. In the **Activity Creator**, set the activity settings, as described in steps 3 and 4 of [Create Activities \(on page 142\)](#).
  3. Click **Place Activity**.
- To move the activity to a different position on the map, click the [Activity Editor Object \(on page 192\)](#), and drag it.
- To change the [Sequence Number \(on page 192\)](#) of an activity, using the Activity Editor:
  1. Click the [Population Editor Object \(on page 189\)](#) and select **Activity Editor**.  
A table with population activities appears, where the activities are ordered into columns based on their sequence number.



2. Drag any activity to a column with a different sequence number.
3. Click **Place Civilian Editor Object**.

The activity is modified.

## 9.2.4 Edit Areas

You can edit existing areas.

### Follow these steps:

Do any of the following:

- To edit the area settings:
  1. Double-click the [Area Editor Object \(on page 199\)](#).
  2. Update the area settings, as described in steps 2 - 3 of [Create Areas \(on page 144\)](#).
  3. Click **OK**.
- To move the area to a different position on the map, click the [Area Editor Object \(on page 199\)](#), and drag it.
- To link an activity to an area (for example, for an activity to be performed in a certain area), right-click an [Activity Editor Object \(on page 192\)](#), select **Link to Area** in the context menu, and click the [Area Editor Object \(on page 199\)](#).

The area changes color and shows the linked activity icon.

To unlink the activity from the area, repeat this step.

#### **WARNING**

You can only link one activity to an area.

- You can attach the area to an entity, such as a human or a vehicle. To do that, right-click the [Area Editor Object \(on page 199\)](#), select **Attach to Unit or Vehicle** in the context menu, and click the entity.

#### **WARNING**

Trying to attach **Restrict Access (Pedestrians)** and **Restrict Access (Vehicles)** areas to moving entities may result in unexpected behavior, as areas of these types cannot move with the entity.

The area is attached to the entity.

#### **NOTE**

You can attach multiple areas to an entity.

To detach the area from the entity, repeat this step.

The area is modified.

## 9.3 Define Civilian Riot

You can add riot behavior to civilian groups to simulate protest marches and on-site demonstrations.

### **i** NOTE

Riot behavior is not defined using the [Control AI - Civilian Editor Object \(on page 20\)](#), but the [\(F3\) Waypoints Editor Object \(on page 18\)](#).

**Image-51: Riot behavior settings**



### Follow these steps:

1. In the Editor (Prepare Mode), select **(F1) Unit** in the Editor Objects List, and double-click a location on the map, where you want to place your group of rioting civilians.  
The Object Properties dialog opens.
2. In **Filters**, select Civilian in the following drop-down:



- The Object Properties dialog displays a list of all the available civilians in VBS4.
3. Select a civilian you want to place on the map.
  4. Make sure **Control AI** is selected (see [Waypoints \(on page 23\)](#)).
  5. Click **OK**.

6. If you want to place more civilians of the same type, press **LCtrl + C** and then **LCtrl + V** to copy and paste the civilian entity. Otherwise, if you want to place civilians of different types, repeat steps 3 - 5.

The civilians are placed on the map.

7. Select all the civilians you placed, click the **RMB**, and select **Orders > Group Selected**.

The civilians are grouped.

8. To assign a riot destination to the civilian group, create a Riot waypoint (see [Waypoints \(on page 23\)](#)) by right-clicking the civilian group marker, selecting **Orders > Assign New Waypoint**, and clicking a position on the map, where the civilian group needs to riot.

**NOTE**

The civilian group marches in riot along the way to the riot destination.

9. In the **Behavior** list, select **Riot**.

10. In the **Weapon Control Status** drop-down, select the behavior for engaging enemy forces, when they are encountered:

Option	Description
<b>No Change</b>	No change in the behavior upon encountering enemy forces.
<b>Weapons Free</b>	Fire at enemy forces, when they are encountered.
<b>Hold Fire</b>	Do not fire at enemy forces, when they are encountered.

**NOTE**

The Weapon Control Status is related to the following SQF commands:

- [combatMode](#) (<https://sqf.bisimulations.com/display/SQF/combatMode>)
- [setCombatMode](#) (<https://sqf.bisimulations.com/display/SQF/setCombatMode>)
- [unitCombatMode](#) (<https://sqf.bisimulations.com/display/SQF/unitCombatMode>)
- [setUnitCombatMode](#)  
(<https://sqf.bisimulations.com/display/SQF/setUnitCombatMode>)

**NOTE**

This controls how the civilians use whatever qualifies as weapons assigned to them in their loadout. For more information, see [Edit Equipment Loadout](#) in the VBS4 Editor Manual.

11. In **Variable Name**, enter the Order waypoint name, which can be used in SQF scripts.

12. Set the Order waypoint completion settings:

Option	Description
<b>Condition to Complete</b>	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
<b>Code on Completion</b>	SQF code to execute on waypoint completion.

13. Click **OK**.

14. Preview the mission.

The civilian group starts a riot march, and continues rioting at the riot destination.

**⚠️ WARNING**

Bohemia Interactive Simulations advises scenario Administrators not to change the **Advanced** settings for the behavior to work correctly.

**ℹ️ NOTE**

After reaching the destination, the riot behavior continues indefinitely.



## 9.4 Define Traffic Flows

Besides defining the pedestrian flow (see [Define Pedestrian Flows \(on page 141\)](#)), you can define a civilian car traffic flow.

The process to create a pedestrian flow is as follows:

### Follow these steps:

1. Make sure that the [Number of Vehicles \(on page 135\)](#) to generate is defined (bigger than 0), as well as the vehicle models (see step 5 of [Define Populations \(on page 133\)](#)).
2. Create traffic spawn and despawn points, areas where vehicles can / cannot go, and traffic signs and lights to regulate the traffic. Once this is done, you can also debug the traffic rules using 3D visualization.

For more information, see:

- [Create Traffic Spawn / Despawn Points \(on the next page\)](#)
- [Create Traffic Areas \(on page 153\)](#)
- [Traffic Signs and Lights \(on page 154\)](#)
- [Traffic Debug Visualization \(on page 159\)](#)

Unlike pedestrians, vehicles can only respond to panic activity. For more information, see [Defining Responsive Behavior \(on page 163\)](#).

3. If you want to modify existing vehicle activities and / or areas, you can edit them. See:
  - [Edit Activities \(on page 146\)](#) (to modify spawn / despawn points)
  - [Edit Areas \(on page 147\)](#) (to modify where vehicles can / cannot go)

## 9.4.1 Create Traffic Spawn / Despawn Points

To create a traffic flow, you need to place spawn and despawn points (activities), where vehicles originate and disappear, respectively.

### Follow these steps:

1. In the Editor Objects List, select **Control AI - Activity** and double-click a location on the map, where you want to place the traffic spawn / despawn activity.

#### **WARNING**

If you place the activity outside the population radius (see [Radius \(on page 135\)](#)), it is not functional. It only becomes functional, if you move it within the population radius.

The Activity Creator opens.



2. In the **Activity Creator**, select any of the vehicle-related spawn / despawn activity types and sub-types (see step 2 in [Create Activities \(on page 142\)](#)):

- **Type** - Select **Spawn / Despawn**.
- **Activity** - Select any of the following:
  - **Vehicle Spawn**
  - **Vehicle Despawn**

#### **WARNING**

Any **Vehicle Spawn** activity requires the presence of a **Vehicle Despawn** activity. Also, **Vehicle Spawn** and **Vehicle Despawn** activities must be placed on roads. Placing them elsewhere does not create vehicle traffic.

3. Set the other activity settings, as needed, in the **Activity Setup**, as described in step 3 of [Create Activities \(on page 142\)](#).

4. Click **OK**.

The spawn / despawn point ([Activity Editor Object \(on page 192\)](#)) is placed on the map.

You can create more spawn / despawn points in the same way. If you want to edit the points, see [Edit Activities \(on page 146\)](#).

## 9.4.2 Create Traffic Areas

You can define areas, where vehicles can / cannot go.

**Follow these steps:**

1. Create an area-based activity, as described in the [Create Activities \(on page 142\)](#) procedure, and in step 3 of the procedure, select:
  - **Type** - Select **Area**.
  - **Activity** - Select **Restrict Access (Vehicles)** in the drop-down.

**i NOTE**

This activity requires the presence of an area, which is why **Create Area** is automatically checked, to create the linked area.

2. Set the other activity settings, as needed, in the **Activity Creator**, as described in step 3 of [Create Activities \(on page 142\)](#), and click **OK**.

The area-based [Activity Editor Object \(on page 192\)](#) and [Area Editor Object \(on page 199\)](#) are created.

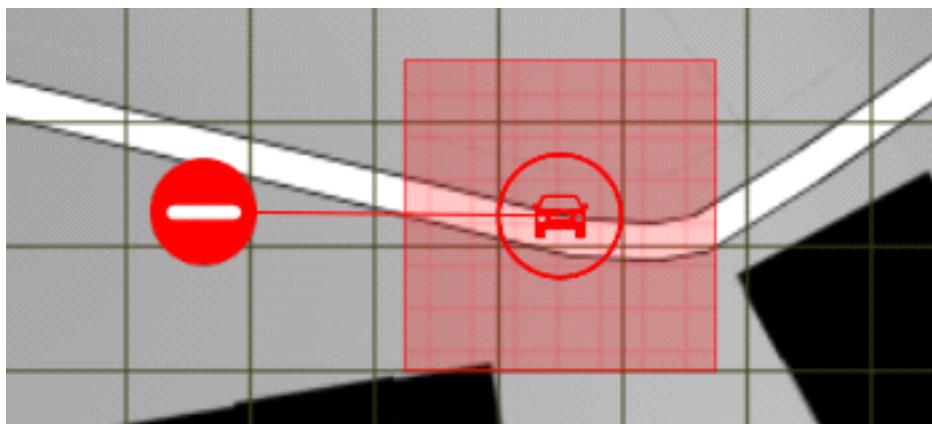
3. Modify the area settings for the [Area Editor Object \(on page 199\)](#), as described in [Edit Areas \(on page 147\)](#).

The [Area Editor Object \(on page 199\)](#) is modified.

The restricted [Area Editor Object \(on page 199\)](#) is set up.

You can create more restricted areas in the same way.

**Image-52: Road segment with a Restricted Access area**



## 9.4.3 Traffic Signs and Lights

The vehicle traffic flow can be regulated by placing traffic signs and lights near junctions or crosswalks, which is a good way to avoid congestion in dense traffic patterns.

### To use signs / traffic lights, follow these steps:

1. Make sure you have added a [Population Editor Object \(on page 189\)](#), as described in [Define Populations \(on page 133\)](#).
2. In the Editor Objects List, select **(F8) Objects**.
3. Right-click a placement position near a junction and select **New Object**.
4. In the **Object Properties** dialog, select the desired [Traffic Signs and Lights Models \(on page 157\)](#).

 **NOTE**

You can switch non-pedestrian traffic lights to red. For more information, see [Changing Lights \(on page 158\)](#).

5. Click **OK**.
6. Rotate the traffic light / sign object in the correct direction.
7. Repeat steps 3-6 for each road in the junction.

Preview the mission to view the vehicle traffic at the junction being controlled by the traffic signs / lights.

 **NOTE**

Unlike the preview mode, in large multiplayer scenarios, to reduce network traffic, traffic signals (signs and lights) pre-placed on the terrain are imported at a limited rate. Because of this, it may take a few minutes to prepare traffic signals, and spawn the civilian population and traffic.

After previewing the mission, a message may appear, noting that some of the junctions have their traffic lights configured incorrectly. A list of positions of the affected junctions is included, and the traffic debug visualization is enabled for those junctions. You can use the debug visualization (see [Traffic Debug Visualization \(on page 159\)](#)) to determine which traffic lights are missing, or need to be adjusted.

Image-53: Message listing junctions with incorrectly placed traffic lights

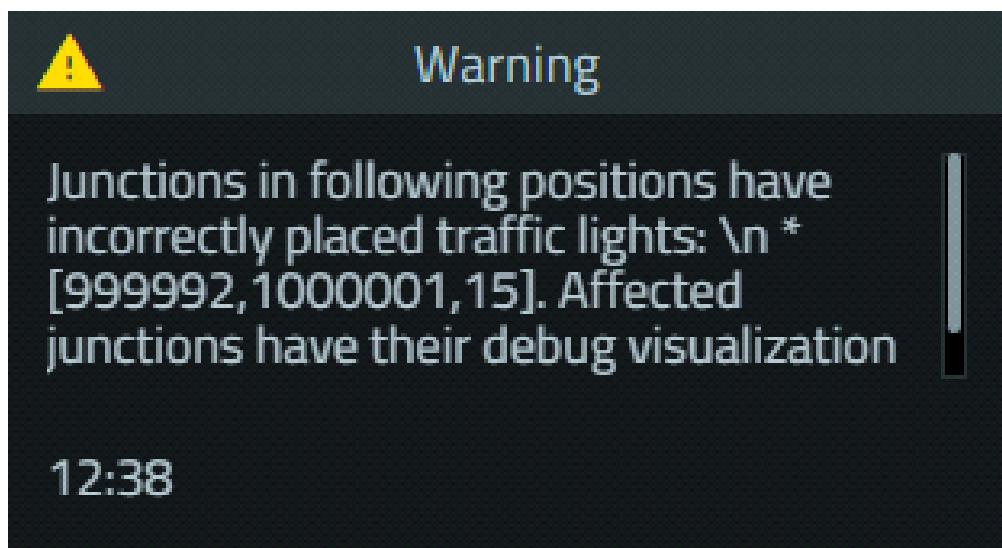
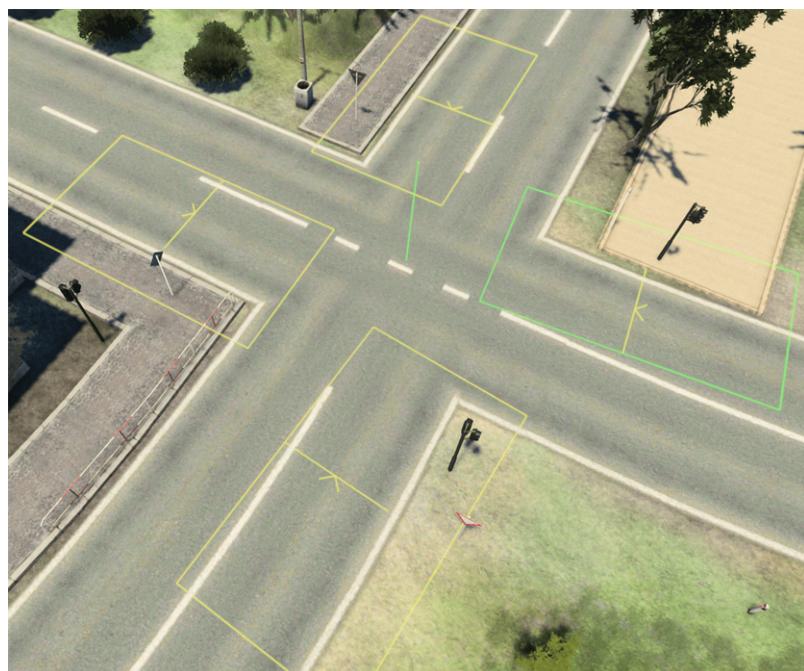


Image-54: Affected junction with debug visualization enabled



In this illustration, the traffic light on the right is placed correctly, the rest are not:

- The bottom traffic light is aligned incorrectly.
- The left traffic light is placed outside the expected area.
- The top traffic light is missing entirely.

**NOTE**

Some traffic light objects have more than one traffic light. For example, the **Traffic Light 01, DE Pedestrian (Left)** object has one light for vehicles, and one light for pedestrians at a 90 degree angle.

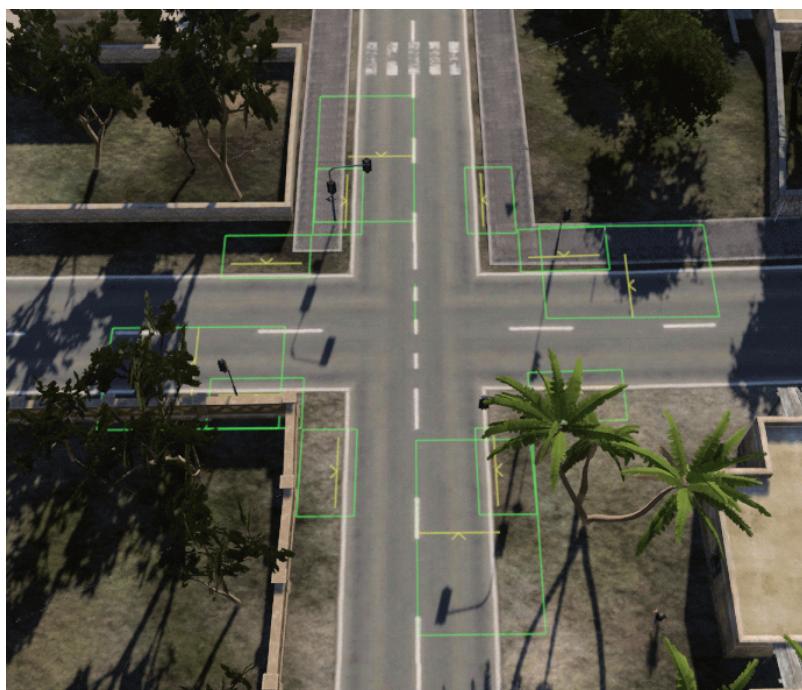
A traffic light is considered to be placed correctly, if it is:

- Aligned with the road.
- Placed within the expected area, as shown in the traffic debug visualization.
- Of the correct type (for example, a pedestrian light is disregarded, when determining the traffic light for a road intended for vehicles).

Each road in a junction needs to have a correctly placed traffic light, in order to be controlled by traffic lights. If only some of the roads have correctly placed traffic lights, the message listing junctions with incorrectly placed traffic lights appears, and the junction traffic lights are ignored.

You can use the "`trafficLightAreas`" option of the Traffic Debug Visualization to see where each road expects its traffic light to be placed. This rectangle turns red, if a problem is encountered on that road. If a traffic light is placed correctly, it is connected to the road by a cyan line, when the "`trafficLightAssignment`" option is enabled.

**Image-55: Debug visualization of a road with all traffic lights placed correctly**



### 9.4.3.1 Traffic Signs and Lights Models

The traffic lights / signs are located in the **Scenery - European Traffic Signs** category.

#### Traffic Signs:

Object Display Name	Object Class
Stop	<code>land_eu_trafficsign_stop_01</code>
Yield	<code>land_eu_trafficsign_yield_01</code>

#### Traffic Lights:

Object Display Name	Object Class
Traffic Light 00, DE Pedestrian (Left)	<code>land_eu_trafficlight_00_011_de</code>
Traffic Light 00, FR Pedestrian (Left)	<code>land_eu_trafficlight_00_011_fr</code>
Traffic Light 01	<code>land_eu_trafficlight_01</code>
Traffic Light 01, DE Pedestrian (Left)	<code>land_eu_trafficlight_01_011_de</code>
Traffic Light 01, DE Pedestrian (Right)	<code>land_eu_trafficlight_01_01r_de</code>
Traffic Light 01, FR Pedestrian (Left)	<code>land_eu_trafficlight_01_011_fr</code>
Traffic Light 01, FR Pedestrian (Right)	<code>land_eu_trafficlight_01_01r_fr</code>
Traffic Light 02	<code>land_eu_trafficlight_02</code>
Traffic Light 02, DE Pedestrian (Left)	<code>land_eu_trafficlight_02_011_de</code>
Traffic Light 02, FR Pedestrian (Left)	<code>land_eu_trafficlight_02_011_fr</code>
Traffic Light 02, Turn Lane	<code>land_eu_trafficlight_02_01tlr</code>
Traffic Light 02, Turn Lane, DE Pedestrian (Left)	<code>land_eu_trafficlight_02_01tlr_011_de</code>
Traffic Light 02, Turn Lane, FR Pedestrian (Left)	<code>land_eu_trafficlight_02_01tlr_011_fr</code>
Traffic Light 03	<code>land_eu_trafficlight_03</code>
Traffic Light 03, DE Pedestrian (Left)	<code>land_eu_trafficlight_03_011_de</code>
Traffic Light 03, FR Pedestrian (Left)	<code>land_eu_trafficlight_03_011_fr</code>
Traffic Light 03, Turn Lane	<code>land_eu_trafficlight_03_01tlr</code>
Traffic Light 03, Turn Lane, DE Pedestrian (Left)	<code>land_eu_trafficlight_03_01tlr_011_de</code>
Traffic Light 03, Turn Lane, FR Pedestrian (Left)	<code>land_eu_trafficlight_03_01tlr_011_fr</code>

### 9.4.3.2 Changing Lights

Non-pedestrian traffic lights can be switched to red in the Editor (Execute Mode).

#### Follow these steps:

1. Right-click the traffic light and select **Force traffic light to red**.

The traffic light switches to red, and all the traffic lights that belong to the same traffic direction switch to red as well. Traffic lights pointing in other traffic directions are unaffected.

2. To resume the traffic light cycle, right-click the traffic light and select **Resume traffic light cycle**.

## 9.4.4 Traffic Debug Visualization

Visualize traffic rules by enabling the traffic debug visualization to gain insight into AI driver intent.

Enable traffic debug visualization using the [fn\\_vbsCon\\_civ\\_setTrafficDebug](https://sqf.bisimulations.com/display/SQF/fn_vbsCon_civ_setTrafficDebug) (https://sqf.bisimulations.com/display/SQF/fn\_vbsCon\_civ\_setTrafficDebug) SQF function.



### TIP

Additional traffic debug visualization can be activated, using the following **Civilian Debug Visualization** UI options (see [Debug Visualization \(on page 168\)](#)):

- **Display Narrow Roads**
- **Display Problematic Turns**
- **Display Problematic Intersections**
- **Display Objects Interfering with Road Traffic**
- **Display Junctions for Crosswalks and Crossroads**

For multi-lane traffic, you can either use the **AI Debug** option in VBS Geo (see VBS Geo User Interface in the VBS Geo Manual), or the **Show Roads** and **Generate Roads** options in the AI Debug Panel (see [Control AI Visualization \(on page 210\)](#)), to see how your road network is set up. Also, to configure road lanes, see **Lanes** in Placing and Editing Roads in the VBS Geo Manual.

### fn\_vbsCon\_civ\_setTrafficDebug

An SQF function that enables / disables traffic debug visualization.



### NOTE

This SQF function only works with civilian vehicles created using the Population Editor Object (see [Population Editor Object \(on page 189\)](#)).

The traffic debug visualization consists of the following, which can be enabled separately:

- **"shape"** - Displays the collision shape of the junction.
- **"entities"** - Displays the intent of vehicles or pedestrians interacting with junctions.
- **"trafficLightAreas"** - Displays the areas where traffic lights are expected.
- **"trafficLightAssignment"** - Displays traffic light assignment to roads or crosswalks.
- **"connections"** - Displays valid paths through the junction, and whether they are free, in use, or blocked.

### Syntax:

```
options call fn_vbsCon_civ_setTrafficDebug
```

**Parameters:**

- **options**: Array of Strings - Contains the visualization options to enable. To enable all the options, use "all"; or "none", to disable all of them.

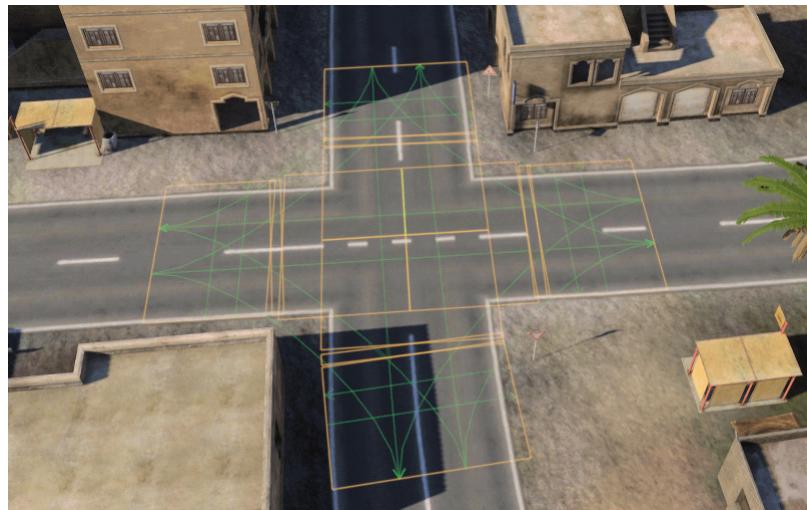
**Return Values:** Nothing**EXAMPLE**

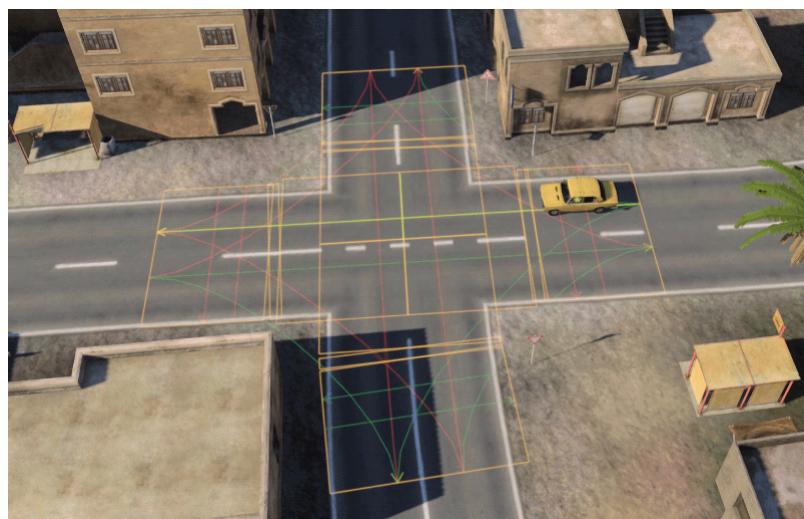
```
["connections", "entities"] call fn_vbsCon_civ_setTrafficDebug  
["all"] call fn_vbsCon_civ_setTrafficDebug  
["none"] call fn_vbsCon_civ_setTrafficDebug
```

With the visualization enabled, all junctions in civilian areas of all instances of the Population Editor Object (see [Population Editor Object \(on page 189\)](#)) display a representation of their current state. The visualization consists of the following elements:

- Orange boxes, enabled by the "shape" option, represent the area considered part of the junction.
- Arrows through the junction, enabled by the "connections" option, represent the directions possible to pass through the junction.

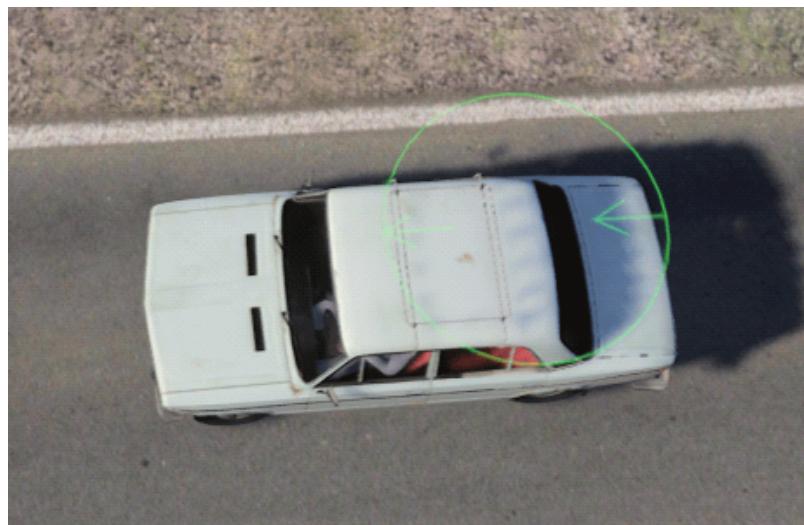
**Image-56: Junction visualization with no vehicles present**



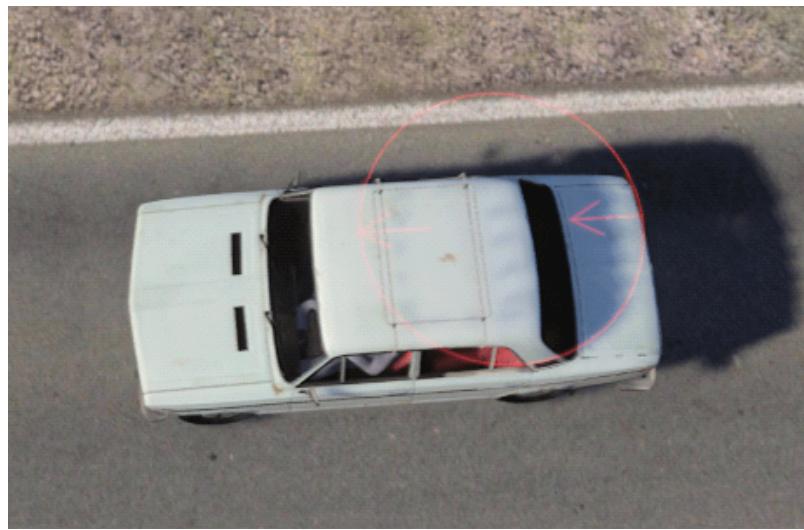
**Image-57: Junction visualization with a vehicle passing through, blocking the intersecting road**

The "entities" option enables a visual representation of intent that civilian entities have when approaching a junction. This visualization has the following elements:

- The arrows represent the direction that the entity takes, when entering and leaving the junction.
- The visualization is green when the entity is free to pass through, and turns red when the entity does not have priority or the direction is blocked.

**Image-58: Entity intent visualization for an entity passing from left to right**

**Image-59: Entity intent visualization for the same entity, when it does not have priority**



## 9.5 Defining Responsive Behavior

With the standard pedestrian and vehicle flows defined (see [Define Pedestrian Flows \(on page 141\)](#) and [Define Traffic Flows \(on page 151\)](#)), you can proceed to define responsive behavior, which occurs when an event happens, causing a temporary or permanent disruption in pedestrian idle and mandatory-sequence activities, as well as vehicle traffic.

### Follow these steps:

1. Define a responsive behavior, based on the **Interrupt** activity sub-type (see [Create Activities \(on page 142\)](#)). See [Interrupt Activity Sub-Types \(below\)](#).
2. In addition to responsive-behavior activities, as a scenario administrator, you can manually trigger events in the VBS Editor (Execute Mode), see:
  - [Resetting Panic \(on the next page\)](#)
  - [Enabling / Disabling the Population \(on page 165\)](#)
  - [Initiating Activities \(on page 166\)](#)

### 9.5.1 Interrupt Activity Sub-Types

The available **Interrupt** activity sub-types are:

Interrupt Activity Sub-Type	Description
<b>Start Panic</b>	Makes civilians run in panic (for example, when a shooting or an explosion occurs). <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"><p><b>NOTE</b></p><p>The following considerations apply:</p><ul style="list-style-type: none"><li>• When panic is active, pedestrians flee, while vehicles stop.</li><li>• Panic is automatically activated as a result of a shooting or an explosion.</li></ul><p>For more information, see the following population settings for panic: <a href="#">Panic Duration (on page 134)</a>, <a href="#">Threat Detection (on page 134)</a>, <a href="#">Flee Distance (on page 135)</a>.</p></div>

Interrupt Activity Sub-Type	Description
Gather Crowd	Causes the civilian entities to stop whatever they are doing, and look at a location.
	<div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"><p><b>NOTE</b></p><p>The following considerations apply:</p><ul style="list-style-type: none"><li>• To look at the specified location, the civilian entities first need to be in an area, linked to the activity (see <a href="#">Create Areas (on page 144)</a>). To force the civilian entities to move to the area, use either the <b>Walk To</b> or <b>Run To</b> activities.</li><li>• Once the civilian entities are in the area, they assume a random position in it, and look at the specified location.</li><li>• The crowd continues to look at the location, until the <b>Gather Crowd</b> activity is removed or disabled with a trigger (see <a href="#">Initiating Activities (on page 166)</a>).</li><li>• Civilians cannot gather on roads.</li></ul></div>
Walk To	Causes the crowd to stop whatever they are doing, and walk to a certain location.
	<div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"><p><b>NOTE</b></p><p>Once the civilians arrive to the location, they resume their other activities.</p></div>
Run To	Causes the crowd to stop whatever they are doing, and run to a certain location.
	<div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"><p><b>NOTE</b></p><p>Once the civilians arrive to the location, they resume their other activities.</p></div>

## 9.5.2 Resetting Panic

As a scenario administrator, you can reset (stop) panic behavior manually, when the behavior is running.

### Follow these steps:

1. Open the Editor (Execute Mode).
2. Right-click the [Population Editor Object \(on page 189\)](#) that generates a population, and select one of the following options:
  - **Reset Panic** - Resets the selected population panic behavior.

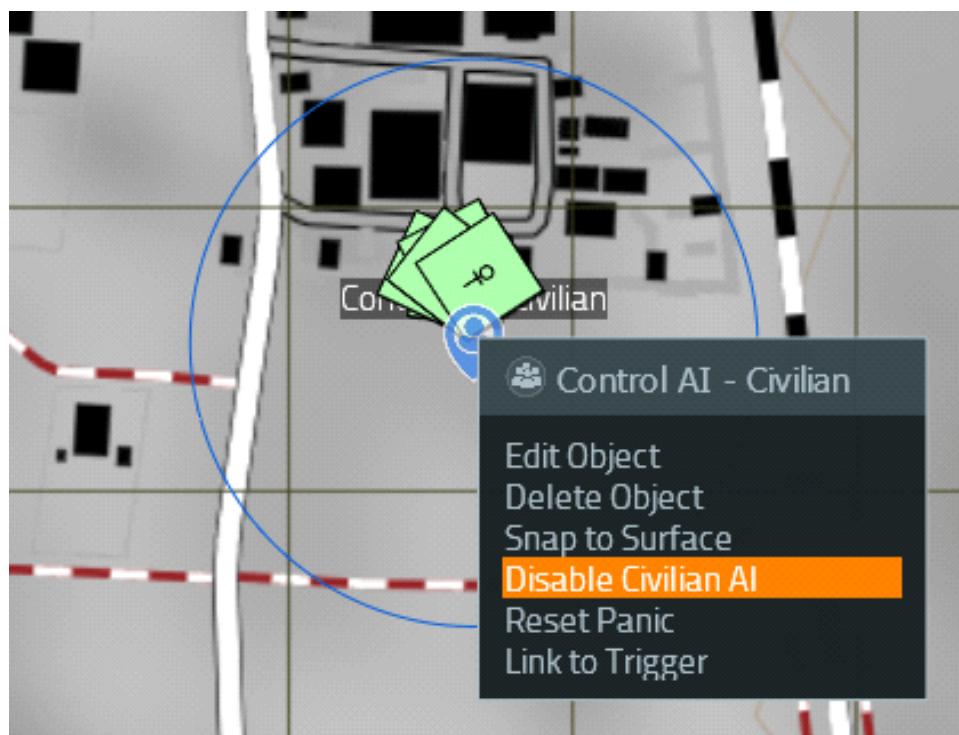
The panic behavior is reset and stopped.

### 9.5.3 Enabling / Disabling the Population

As a scenario administrator, you can enable / disable an entire population using Population Editor Object or by using a Trigger. A disabled population disappears from the simulation, and re-enabling it makes it reappear.

In the Editor (Execute Mode), right-click the [Population Editor Object \(on page 189\)](#) that generates a population, and select one of the following options:

- **Enable Civilian AI** - The selected population appears and starts the civilian pattern of life simulation.
- **Disable Civilian AI** - The selected population disappears and stops the civilian pattern of life simulation.



#### Use a Trigger Editor Object:

1. Open the Editor (Prepare Mode).
2. Select (**F7**) **Trigger** in the Editor Objects List, and place a Trigger Editor Object on the map.

- Fill the Trigger Editor Object settings, as required (for more information, see Triggers in the VBS4 Editor Manual), and click **OK**.

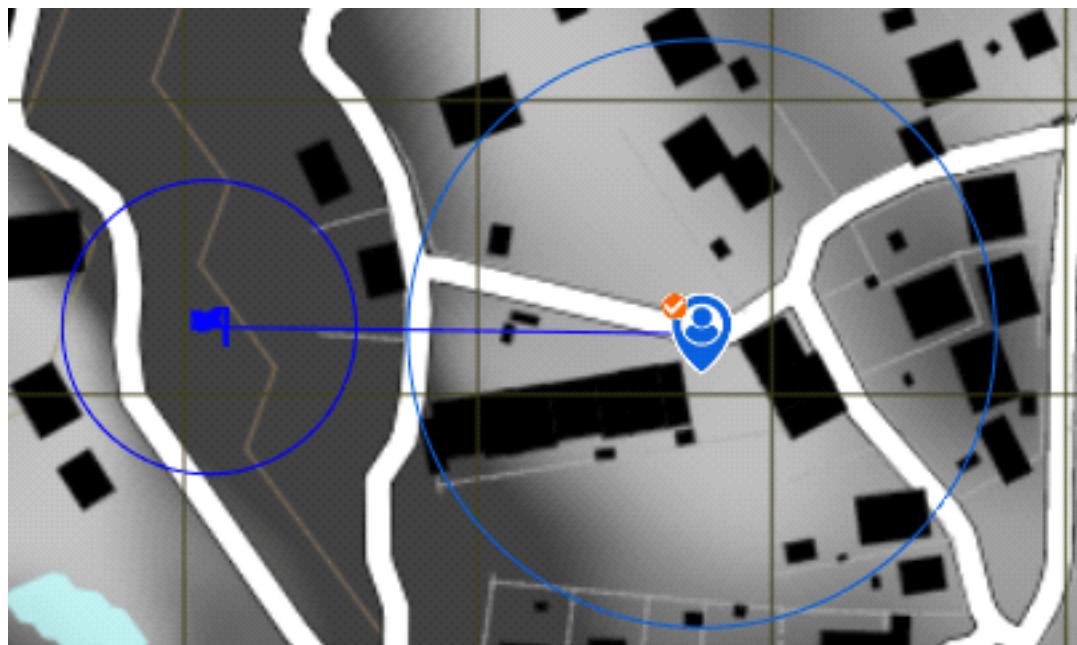
**NOTE**

It is recommended to set the **Repeatedly** setting to **false** to allow repeated activation / deactivation of the trigger, to start / stop the population activity.

The Trigger Editor Object is created.

- Link the Trigger Editor Object to an [Population Editor Object \(on page 189\)](#) by right-clicking the Population Editor Object, selecting **Link to Trigger** in the context menu, and clicking the Trigger Editor Object.
- In the Editor (Execute Mode), right-click the Trigger Editor Object, and select **Activate Trigger / Deactivate Trigger** to activate / deactivate the Trigger Editor Object.

**Image-60: Trigger Editor Object linked to a Population Editor Object**



The civilian pattern of life simulation appears / disappears for the selected population.

## 9.5.4 Initiating Activities

As a scenario administrator, you can use triggers to start / stop (or lock / unlock) various population activities in areas.

**Follow these steps:**

- Open the Editor (Prepare Mode).
- Select (**F7**) **Trigger** in the Editor Objects List, and place a Trigger Editor Object on the map.

- Fill the Trigger Editor Object settings, as required (for more information, see Triggers in the VBS4 Editor Manual), and click **OK**.

**i** **NOTE**

It is recommended to set the **Repeatedly** setting to **false** to allow repeated activation / deactivation of the trigger, to start / stop the population activity.

The Trigger Editor Object is created.

- Link the Trigger Editor Object to an [Activity Editor Object \(on page 192\)](#), which generates a certain population activity, by right-clicking the Activity Editor Object, selecting **Link to Trigger** in the context menu, and clicking the Trigger Editor Object.

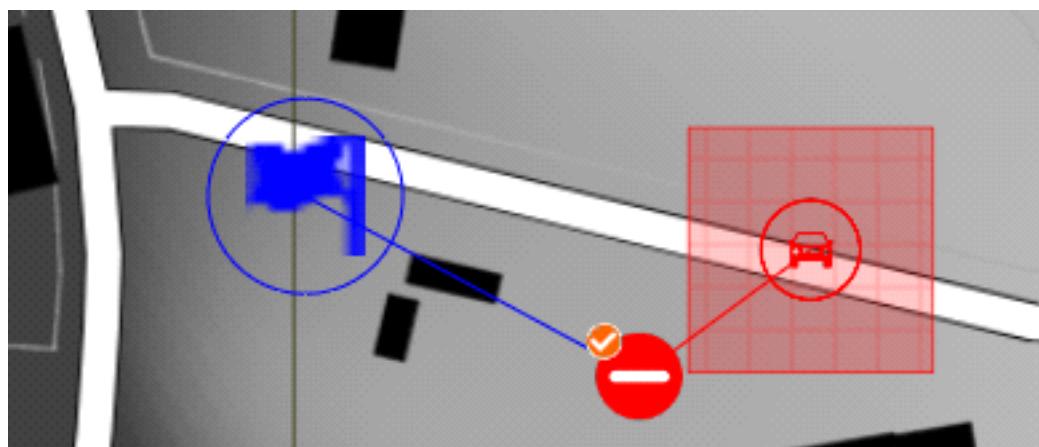
**!** **WARNING**

Since the activity is activated / deactivated in a certain area, make sure there is also an [Area Editor Object \(on page 199\)](#) linked to the Activity Editor Object (see step 4 of [Create Areas \(on page 144\)](#)).

- In the Editor (Execute Mode), right-click the Trigger Editor Object, and select **Activate Trigger** / **Deactivate Trigger** to activate / deactivate the Trigger Editor Object.

Activating / deactivating the trigger starts / stops the linked activity in the given area.

**Image-61: Trigger Editor Object linked to an Activity Editor Object**



## 9.6 Debug Visualization

The Population Editor Object (see [Population Editor Object \(on page 189\)](#) in the VBS Control AI Manual) has civilian AI visualizations, which can be used for debugging purposes.

For general Control AI visualizations, see [Control AI Visualization \(on page 210\)](#).

### NOTE

The visualizations are only displayed in the 2D and 3D views of VBS Editor (Execute Mode).

**Follow these steps:**

1. Double-click the Population (**Control AI - Civilian**) Editor Object.
2. Select **Debug Visualization**, and set any of the available debug options:

3D Visualization Options;

- [Display Narrow Roads \(on the next page\)](#)
- [Display Problematic Turns \(on the next page\)](#)
- [Display Problematic Intersections \(on page 170\)](#)
- [Display Objects Interfering with Road Traffic \(on page 170\)](#)

2D Visualization Options:

- [Display Junctions for Crosswalks and Crossroads \(on page 171\)](#)

3. Click **Apply Selected**, to activate the debug options.

The visualizations are activated in the population area.

### NOTE

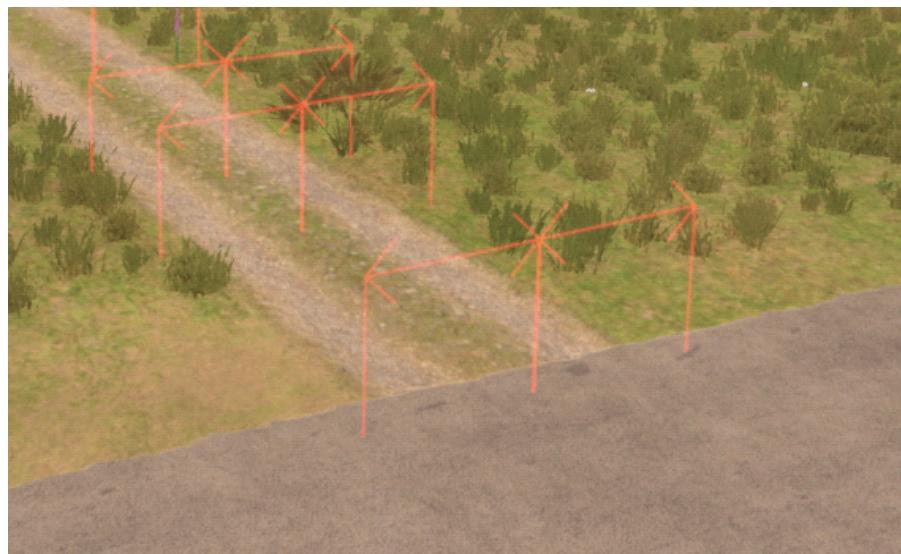
The Civilian AI visualizations only apply to the Control AI - Civilian Editor Object selected in step 1, and not to any other Control AI - Civilian Editor Objects on the map.

## 9.6.1 Display Narrow Roads

Displays road sections that are too narrow to drive through.

This option can assist with defining the vehicle traffic flow. For more information, see [Define Traffic Flows \(on page 151\)](#).

**Image-62: Road section that is too narrow**



## 9.6.2 Display Problematic Turns

Displays road turns that cars cannot use properly.

This option can assist with defining the vehicle traffic flow. For more information, see [Define Traffic Flows \(on page 151\)](#)

**Image-63: Problematic turn**



### 9.6.3 Display Problematic Intersections

Displays junction pairs that are too close to each other, which can cause cars to get stuck.

This option can assist with defining the vehicle traffic flow. For more information, see [Define Traffic Flows \(on page 151\)](#)

**Image-64: Problematic intersection**

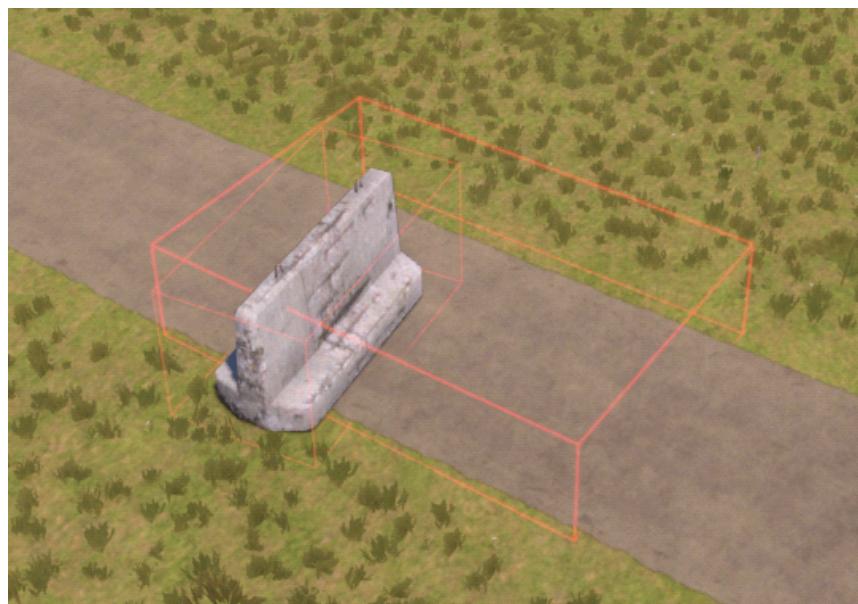


### 9.6.4 Display Objects Interfering with Road Traffic

Displays any objects that prevent cars from driving on roads.

This option can assist with defining the vehicle traffic flow. For more information, see [Define Traffic Flows \(on page 151\)](#)

**Image-65: The red cube indicates road interference**



## 9.6.5 Display Junctions for Crosswalks and Crossroads

Displays several visualizations for crosswalks and crossroads (junctions).

This option can assist with defining the pedestrian and vehicle traffic flows. For more information, see [Define Pedestrian Flows \(on page 141\)](#) and [Define Traffic Flows \(on page 151\)](#).

The visualizations are:

Icon	Description	Icon	Description
	Indicates a junction with crosswalks and without traffic lights.		Indicates a crosswalk.
	Indicates a junction with crosswalks and correctly placed traffic lights.		Indicates a crosswalk with correctly placed traffic lights.
	Indicates a junction with crosswalks and incorrectly placed traffic lights.		Indicates a crosswalk with incorrectly placed traffic lights.

**Image-66: Visualized crosswalks and crossroads**



## 9.7 Basic Example Scenario

This topic demonstrates the main features of the Civilian AI functionality, discussed in the previous [Civilian AI \(on page 132\)](#) topics, and how to use them to design an example train station scenario.

### Scenario Definition

The example scenario is the simulation of civilians at a train station.

The example uses three populations:

- **Departing Passengers** - A pedestrian population that spawns at the train station entrance, enters the station, buys items at the train station shops and / or vending machines, gets tickets, and despawns at the train platform (simulating boarding the train).
- **Arriving Passengers** - A pedestrian population that spawns at the trains (simulating getting off the train), buys items at the station shops and / or vending machines, leaves the station, and despawns.
- **City Population** - A general (pedestrian and vehicle traffic) ambient population that serves as the city population, where the train station is located.

The scenario also defines a train-station attacker threat, which is a moving hostile entity that causes panic within a predefined distance / area from civilians, and a safe zone, where departing and arriving passengers flee from the threat.

In addition, the train station also has a restricted area, which can be restricted / allowed in-game by the scenario administrator, using a Trigger Editor Object (see [Triggers in the VBS4 Editor Manual](#)). This can be used to simulate a train-station area, where passengers are not allowed.

For more information, see:

- [Define Smart Objects \(on the next page\)](#)
- [Define Departing Passengers \(on page 175\)](#)
- [Define Arriving Passengers \(on page 179\)](#)
- [Define the City Population \(on page 179\)](#)
- [Define the Threat \(on page 181\)](#)
- [Define the Safe Zone \(on page 184\)](#)
- [Define the Restricted Area \(on page 186\)](#)
- [Run the Scenario as Administrator \(on page 187\)](#)

## 9.7.1 Define Smart Objects

Place [Smart Objects \(on page 200\)](#) for the **Departing Passengers** and **Arriving Passengers** populations to interact with on the train-station premises (the Smart Objects are used to automatically create civilian interaction activities - for more information, see [Define Pedestrian Flows \(on page 141\)](#)).

### ★ FEATURE NOTICE

This feature is part of French VBS4, a licensed product. For more information, contact [sales@bisimulations.com](mailto:sales@bisimulations.com).

#### Follow these steps:

1. Place ticket machines, where civilians can buy train tickets: in the Editor Objects List, select the **(F8) Objects > FR Structures > Ticket Machine**, and place it on the map.



2. Place trash bins, where civilians can dispose trash: in the Editor Objects List, select the **(F8) Objects > FR Structures > Trashbin**, and place it on the map.



3. Place benches for civilians to sit on: in the Editor Objects List, select **(F8) Objects > FR Structures > Bench**, and place it on the map.

**Image-67: Civilians sitting on a bench**



4. Place timetable displays for civilians to use: in the Editor Objects List, select **(F8) Objects > FR Structures > Info Screen**, and place it on the map.

**Image-68: Civilians checking a timetable display**



5. Repeat any of the preceding procedure steps to place more Smart Objects.

Once the Smart Objects are placed, the relevant activities can be created automatically in:

- [Define Departing Passengers \(on the next page\)](#)
- [Define Arriving Passengers \(on page 179\)](#)

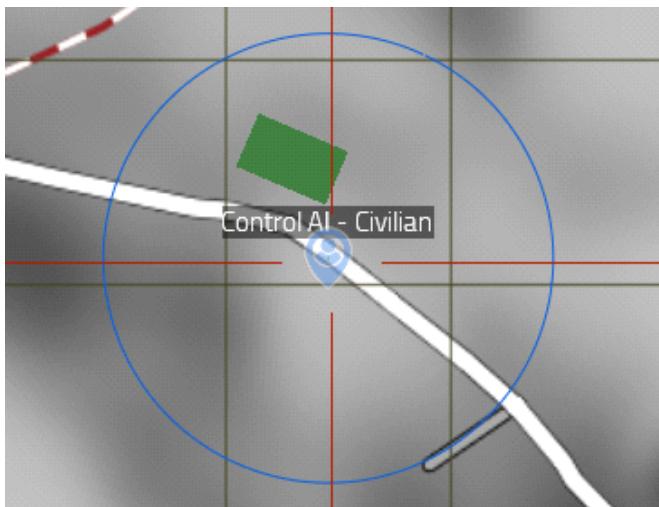
## 9.7.2 Define Departing Passengers

The departing passengers population is a pedestrian-only (no vehicles defined) population that spawns at the station entrance, enters, buys items at the train station shops and / or vending machines, gets tickets, and despawns at the train platform (simulating boarding the train).

### Follow these steps:

1. In the Editor Objects List, select the **Control AI - Civilian** Editor Object, and place it in such a way on the map, so that it covers the train station area.
2. In **Civilian > Settings**, set:
  - **Number of Pedestrians** - The number of departing passengers you want.
  - Set the following panic-related settings (the panic behavior is expanded on in [Define the Threat \(on page 181\)](#) and [Define the Safe Zone \(on page 184\)](#)):
    - **Panic Duration** - Set the panic duration (in minutes) for the departing passengers.
    - **Threat Detection** - Set the departing-passengers threat detection to **Only populace within hearing range or line of sight reacts to threats**.
    - **Flee Distance** - Set the departing-passenger flee distance from the threats (in meters). Choose any value.
  - **Number of Vehicles** - Set to 0 as this is only a departing-passenger population.
  - **Radius** - Make the population radius big enough to cover the train station area.

**Image-69: Control AI - Civilian Editor Object covering the train station**



- **Civilian AI Identifier** - Select **Cobalt** in the drop-down.
- **Smart Objects** - Check this to automatically create relevant activities for the Smart Objects you defined in [Define Smart Objects \(on page 173\)](#).

3. **Optional:** In **Civilian > Populace**, you change the default pedestrian models. For more information, see [Specify Civilian Populace \(on page 136\)](#).

**i** **NOTE**

Since the departing-passengers population has no vehicles, **Civilian > Vehicles** should be left as is. By setting **Number of Vehicles** to 0, no vehicles are generated.

4. Click **Place Civilian Editor Object** to place the population on the map.

The departing-passengers population is placed on the map, along with Smart Object activities for departing passengers.

Now, you can add spawn points at the train station entrance, to simulate departing passengers entering the station.

5. In the Editor Objects List, select **Control AI - Activity** and place it next to the station entrance.
6. In the **Activity Creator**, set:

- **Name** - The name of your spawn point (for example, "Enter Station").
- **Type** - Select **Spawn / Despawn** in the drop-down.
- **Activity** - Select **Spawn** in the drop-down.
- **Civilian AI Identifier** - Select **Cobalt** in the drop-down, to match the departing-passengers population **Civilian AI Identifier**.

7. Click **Place Activity**.

The spawn point is placed by the train station entrance.

Repeat steps 5 - 7 to create additional spawn points by the station entrance, if needed.

8. In the Editor Objects List, select **Control AI - Activity** and place it as a loitering activity anywhere in the train station area.

**Image-70: Loitering civilians**



9. In the **Activity Creator**, set:

- **Name** - The name of your loitering activity (for example, "Walk Around").
- **Type** - Select **Generic** in the drop-down.
- **Activity** - Select **Loiter** in the drop-down.
- **Civilian AI Identifier** - Select **Cobalt** in the drop-down, to match the departing-passengers population **Civilian AI Identifier**.
- **Sequence Number** - To make loitering follow ticket-purchasing, this should be set to 2.
- **Capacity** - Specify how many departing passengers you want to loiter at the loitering activity location.

**NOTE**

If there is no free capacity left at the activities with a relevant **Mandatory** sequence (see step 9 and the **Buy Item (Vending Machine)** activity), the civilian entities loiter around the place. It is not necessary to explicitly specify a loitering area / activity. However, if loitering has to happen at a specific location, this can be achieved by specifying an activity or an area for it.

- **Priority** - To make loitering optional, select **Optional**.
- **Create Area** - Check this option to create a loitering area.

10. Click **Place Activity**.

The loitering activity and area are placed on the map. If you want to change the loitering area size, see [Edit Areas \(on page 147\)](#).

11. In the Editor Objects List, select **Control AI - Activity** and place it on the train-station platform, next to the train.

**TIP**

The **(F8) Objects > FR Structures > Train Wagon** is specifically intended for the despawn train-boarding activity.

**FEATURE NOTICE**

This feature is customer exclusive. For more information, contact [sales@bisimulations.com](mailto:sales@bisimulations.com).

**Image-71: Civilians boarding the train**



12. In the **Activity Creator**, set:

- **Name** - The name of your despawn point (for example, "Board Train").
- **Type** - Select **Spawn / Despawn** in the drop-down.
- **Activity** - Select **Despawn** in the drop-down.
- **Civilian AI Identifier** - Select **Cobalt** in the drop-down, to match the departing-passengers population **Civilian AI Identifier**.

13. Click **Place Activity**.

The despawn point is placed by the train.

Create additional despawn points along other train sections, if needed.

The departing-passengers population definition is complete.

## 9.7.3 Define Arriving Passengers

The arriving passengers population is a pedestrian-only (no vehicles defined) population that spawns at the trains (simulating getting off the train), buys items at the station shops and / or vending machines, leaves the station, and despawns.

### Follow these steps:

1. Repeat steps 1 - 4 of [Define Departing Passengers \(on page 175\)](#) to create the population. Select **Sky** as the **Civilian AI Identifier**, to distinguish from departing passengers.

The arriving population is placed, along with Smart Object activities for departing passengers.

Now, add spawn points at the platform, next to the train, to simulate passengers getting off the train.

2. Repeat steps 5 - 7 of [Define Departing Passengers \(on page 175\)](#) to create spawn points. Select **Sky** as the **Civilian AI Identifier**, to match the arriving population **Civilian AI Identifier**.
3. Repeat steps 8 - 10 of [Define Departing Passengers \(on page 175\)](#) to create the loitering activity. Select **Sky** as the **Civilian AI Identifier**, to match the arriving population **Civilian AI Identifier**:

Now, add despawn points at the station exit, to simulate passengers leaving the station.

4. Repeat steps 11 - 13 of [Define Departing Passengers \(on page 175\)](#) to create despawn points. Select **Sky** as the **Civilian AI Identifier**, to match the arriving population **Civilian AI Identifier**.

The arriving passengers population definition is complete.

## 9.7.4 Define the City Population

The city population is a general ambient population, surrounding the train station.

### Follow these steps:

1. In the Editor Objects List, select the **Control AI - Civilian** Editor Object, and place it in such a way on the map, so that it covers the city area, where the train station is located.
2. In **Civilian > Settings**, set:
  - **Number of Pedestrians** - The number of city pedestrians you want.
  - **Number of Vehicles** - The number of city vehicles you want.
  - **Radius** - Make the city population radius big enough to cover the city area.
  - **Civilian AI Identifier** - Select **Shamrock** to distinguish from the departing and arriving populations.
3. **Optional:** In **Civilian > Populace** and **Civilian > Vehicles**, you change the default pedestrian and vehicle models. For more information, see [Specify Civilian Populace \(on page 136\)](#) and [Specify Vehicle Traffic \(on page 138\)](#).

4. To create a vehicle spawn point, in the Editor Objects List, select **Control AI - Activity** and place it on one of the city roads.
5. In the **Activity Creator**, set:
  - **Name** - The name of your spawn point (for example, "Vehicles Entering City").
  - **Type** - Select **Spawn / Despawn** in the drop-down.
  - **Activity** - Select **Vehicle Spawn** in the drop-down.
  - **Civilian AI Identifier** - Select **Shamrock** in the drop-down, to match the city population **Civilian AI Identifier**.
6. Click **Place Activity**.

The vehicle spawn point is placed on the city road.  
Create additional vehicle spawn points on other city roads, if needed.
7. To create vehicle despawn points, repeat steps 4 - 6, and for each despawn point, select **Vehicle Despawn** in the **Activity** drop-down.
8. Create pedestrian spawn and despawn points in the same way as described in [Define Departing Passengers \(on page 175\)](#) next to city buildings, to simulate city pedestrians leaving and entering them. For each pedestrian spawn and despawn activity, select **Shamrock** in the **Civilian AI Identifier** drop-down, to match the city population **Civilian AI Identifier**.

The city population definition is complete.

**Image-72: City population with pedestrians and vehicle traffic**



## 9.7.5 Define the Threat

The threat is a moving attacker entity that causes panic within a predefined distance / area from pedestrians and vehicles.

### NOTE

Panic has the following considerations:

- When panic is active, pedestrians flee, while vehicles stop.
- Panic is automatically activated as a result of a shooting or an explosion.

### Follow these steps:

1. In the Editor Objects List, select **(F1) Unit** and double-click a location on the map next to or within the train station, where you want to place the attacker entity.
2. Click **OK**.

The attacker entity is placed on the map.

Now, make the attacker move to the train station.

3. Right-click the attacker entity, select **Orders > Assign New Waypoint** in the context menu, and click a location inside the train station.
4. In the **Objects Properties** dialog, select **Individual - Move** in the **Behavior** list, and click **OK**.

The attacker entity has a Control AI Individual - Move waypoint, represented by the [Individual - Move Order \(on page 50\)](#), to enter the train station.

Now, you can define panic behavior for departing and arriving passengers. The panic behavior consists of two panic activities - one for departing and another for arriving passengers. Each panic activity is linked to its own panic area, and both panic areas are then linked to the attacker entity. For realism, it is recommended for the two areas to overlap, to simulate both departing and arriving passengers reacting in panic to the attacker entity.

5. In the Editor Objects List, select **Control AI - Activity** and place the activity on the map, close to the attacker entity.

## 6. In the **Activity Creator**, set:

- **Name** - The name of your panic activity (for example, "Departing Passengers Panic").
- **Type** - Select **Interrupt** in the drop-down.
- **Activity** - Select **Start Panic** in the drop-down.

### **NOTE**

This activity requires the presence of an area, which is why **Create Area** is automatically checked, to create the linked area.

This allows you to create a departing-passengers panic area.

- **Civilian AI Identifier** - Select **Cobalt** in the drop-down, to match the departing-passengers population **Civilian AI Identifier**.

## 7. Click **Place Activity**.

The departing-passengers panic activity and area are created. If you want to change the panic area size, see [Edit Areas \(on page 147\)](#).

## 8. Right-click the area, select **Attach to Unit or Vehicle** in the context menu, and click the attacker entity.

The departing-passengers panic area is attached to the attacker.

## 9. In the Editor Objects List, select **Control AI - Activity** and place the activity on the map, close to the attacker entity.

## 10. In the **Activity Creator**, set:

- **Name** - The name of your panic activity (for example, "Arriving Passengers Panic").
- **Type** - Select **Interrupt** in the drop-down.
- **Activity** - Select **Start Panic** in the drop-down.

### **NOTE**

This activity requires the presence of an area, which is why **Create Area** is automatically checked, to create the linked area.

This allows you to create an arriving-passengers panic area.

- **Civilian AI Identifier** - Select **Sky** in the drop-down, to match the arriving-passengers population **Civilian AI Identifier**.

## 11. Click **Place Activity**.

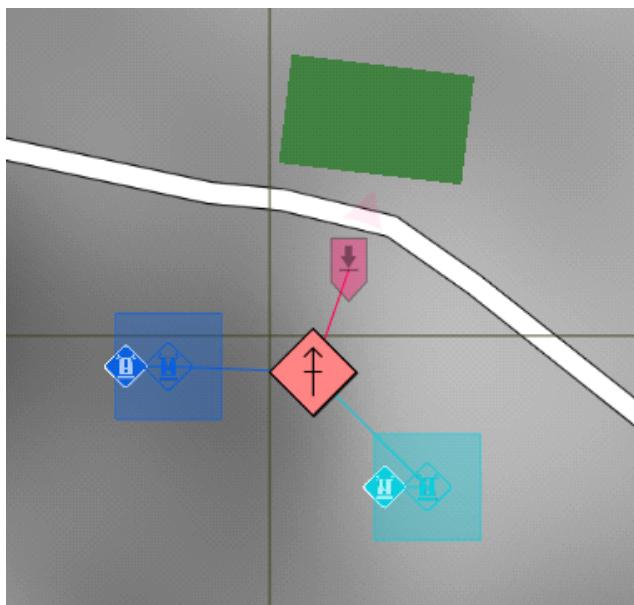
The arriving-passengers panic activity and area are created. If you want to change the panic area size, see [Edit Areas \(on page 147\)](#).

12. Right-click the area, select **Attach to Unit or Vehicle** in the context menu, and click the attacker entity.

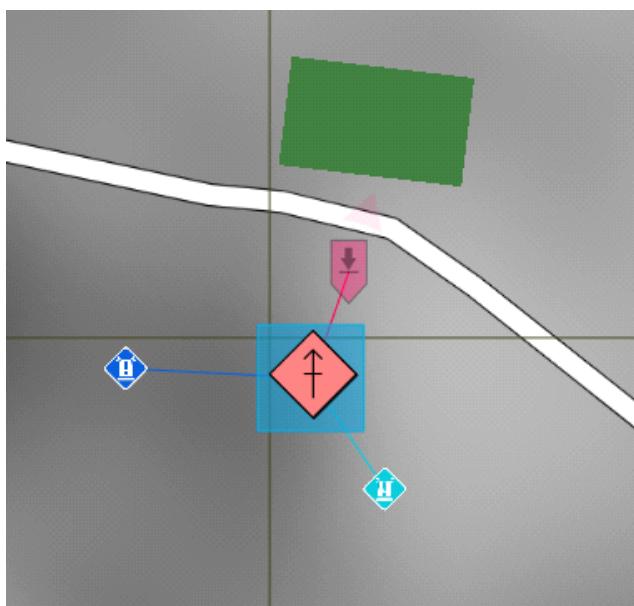
The arriving-passengers panic area is attached to the attacker.

13. Move both departing-passengers and arriving-passengers areas so that they overlap, with the attacker entity at the center of each.

**Image-73: Areas with an attacker entity, before overlap**



**Image-74: Overlapping areas with an attacker entity**



The attacker entity is set up to cause panic among departing and arriving passengers.

## 9.7.6 Define the Safe Zone

The scenario has a safe zone, where departing and arriving passengers flee from the threat.

From a scenario-design perspective, there are two safe zones (one for departing and another for arriving passengers) that are used to hide from the train-station attacker, where civilians run in panic. For realism, to simulate a common safe zone, the two safe zones are made to overlap and occupy the same area that all the train-station civilians (departing and arriving passengers) can use.

### Follow these steps:

1. In the Editor Objects List, select **Control AI - Activity** and place it on the map, close to where you want the common safe zone to be.
2. In the **Activity Creator**, set:
  - **Name** - The name of your panic activity (for example, "Departing Passengers Safe Zone").
  - **Type** - Select **Area Based** in the drop-down.
  - **Activity** - Select **Safe Zone** in the drop-down.
  - **Civilian AI Identifier** - Select **Cobalt** in the drop-down, to match the departing-passengers population **Civilian AI Identifier**.
  - **Create Area** - Check this option to create a departing-passengers safe zone area.

### 3. Click **Place Activity**.

The departing-passengers safe zone activity and area are created. If you want to change the safe-zone area size, see [Edit Areas \(on page 147\)](#).

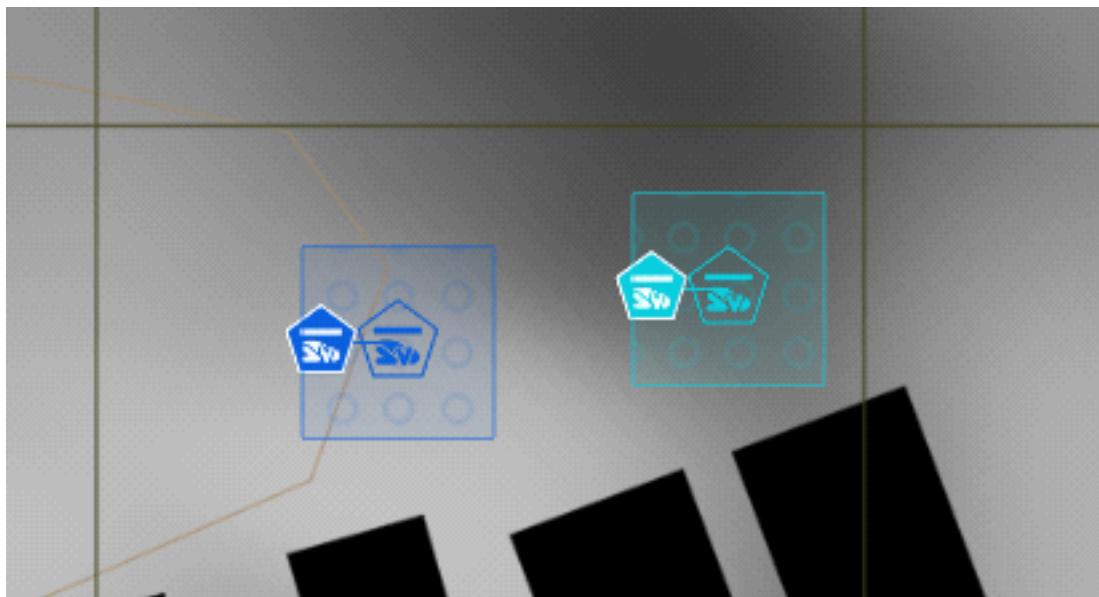
4. In the Editor Objects List, select **Control AI - Activity** and place it on the map, close to where you want the common safe zone to be.
5. In the **Activity Creator**, set:
  - **Name** - The name of your panic activity (for example, "Arriving Passengers Safe Zone").
  - **Type** - Select **Area Based** in the drop-down.
  - **Activity** - Select **Safe Zone** in the drop-down.
  - **Civilian AI Identifier** - Select **Sky** in the drop-down, to match the arriving-passengers population **Civilian AI Identifier**.
  - **Create Area** - Check this option to create an arriving-passengers safe zone area.

### 6. Click **Place Activity**.

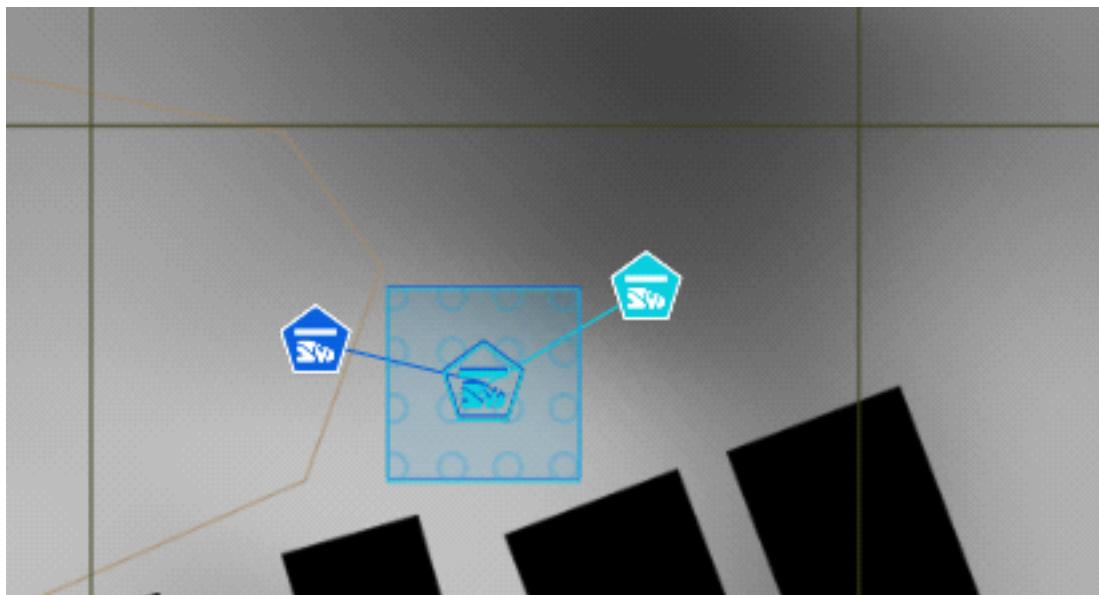
The arriving-passengers safe zone activity and area are created. If you want to change the safe-zone area size, see [Edit Areas \(on page 147\)](#).

7. Move both departing-passengers and arriving-passengers safe zone areas so that they overlap, to create a common safe zone.

**Image-75: Safe zone areas, before overlap**



**Image-76: Overlapping safe zone areas**



The common safe zone, for both departing and arriving passengers, is defined.

## 9.7.7 Define the Restricted Area

The train station also has a restricted area for pedestrian civilians, which can be restricted / unrestricted in-game by the scenario administrator, using a Trigger Editor Object (see Triggers in the VBS4 Editor Manual). This can be used to simulate a train-station area, where passengers are not allowed.

### Follow these steps:

1. In the Editor Objects List, select **Control AI - Activity** and place it on the map, close to where you want the restricted area to be.
2. In the **Activity Creator**, set:
  - **Name** - The name of your panic activity (for example, "Staff-Only Area").
  - **Type** - Select **Area Based** in the drop-down.
  - **Activity** - Select **Restrict Access (Pedestrians)** in the drop-down.

 **NOTE**

This activity requires the presence of an area, which is why **Create Area** is automatically checked, to create the linked area.

3. Click **Place Activity**.

The restricted-access activity and area are created. If you want to change the restricted area size, see [Edit Areas \(on page 147\)](#).

4. In the Editor Objects List, select **(F7) Trigger** and place it next to the restricted-access area.
5. Fill the Trigger Editor Object settings, as required (for more information, see Triggers in the VBS4 Editor Manual), and click **OK**.

 **WARNING**

To repeatedly activate / deactivate the trigger, make sure to set **Repeatedly** to **true**.

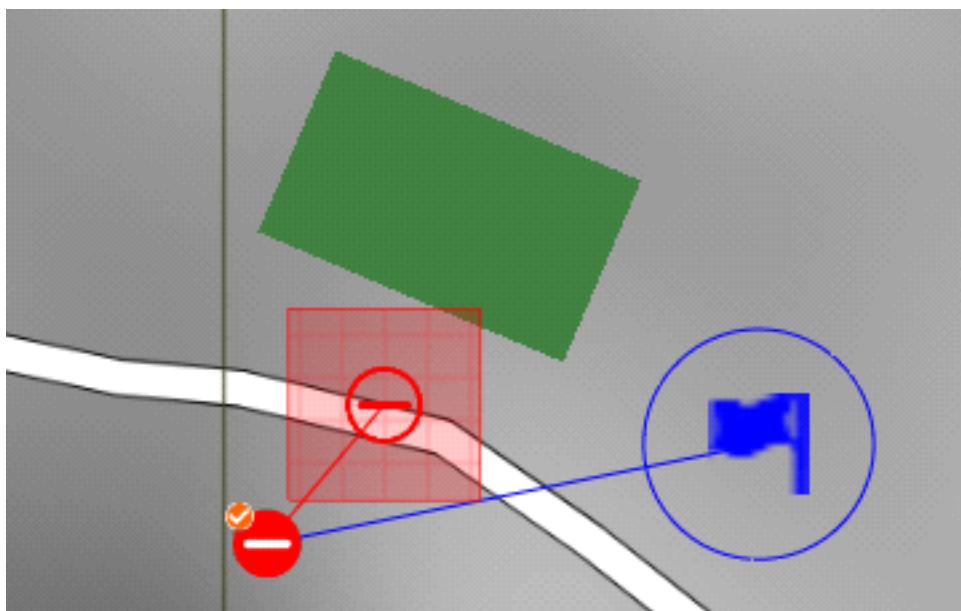
The Trigger Editor Object is created.

6. Right-click the restricted-access activity, select **Link to Trigger** in the context menu, and click the Trigger Editor Object.

The restricted-access activity is linked to the trigger.

The restricted-access area is defined.

### Image-77: Restricted-access activity, area, and trigger



### 9.7.8 Run the Scenario as Administrator

Once the scenario is designed, you can run it as the scenario administrator.

When you run the scenario, you observe the following events in the Editor (Execute Mode):

1. The three populations are generated:
  - Departing and arriving passengers appear at the train station, and engage in their assigned activities, as described in [Define Departing Passengers \(on page 175\)](#) and [Define Arriving Passengers \(on page 179\)](#).
  - The city population (pedestrians and vehicle traffic) appears across the city. Pedestrians engage in their assigned activities, while vehicles drive from their spawn to their despawn points, as described in [Define the City Population \(on page 179\)](#).
2. The attacker entity enters the train station and causes departing and arriving passengers to flee in panic to their common safe zone, as described in [Define the Threat \(on page 181\)](#) and [Define the Safe Zone \(on page 184\)](#).

You can manually restrict / allow civilian access to the restricted-access area created in [Define the Restricted Area \(on the previous page\)](#).

**Follow these steps:**

1. In the Editor (Execute Mode), right-click the Trigger Editor Object.
2. Select **Activate Trigger / Deactivate Trigger** to activate / deactivate the Trigger Editor Object.

Activating / deactivating the trigger restricts / allows civilian access to the restricted-access area.

## 9.8 Civilian AI Editor Objects

The civilian Control AI uses the following Editor Objects:

Editor Object	Description
<a href="#">Population Editor Object (on the next page)</a>	Defines the population (pedestrians and vehicles).
<a href="#">Activity Editor Object (on page 192)</a>	Defines the population activities (for a given population).
<a href="#">Area Editor Object (on page 199)</a>	Defines an area within the general population area.
<a href="#">Smart Objects (on page 200)</a>	Lists additional Editor Objects that civilians can interact with.

## 9.8.1 Population Editor Object

The Population Editor Object (appears as **Control AI - Civilian** in the Editor (Prepare / Execute Mode) Editor Objects List - see [Control AI UI Overview \(on page 17\)](#)) defines the population (pedestrians and vehicles). For more information, see [Define Populations \(on page 133\)](#).

It consists of the following settings:

Population Setting	Description
<b>Number of Pedestrians</b>	Number of pedestrians in the population.
<b>Panic Duration</b>	Panic duration (if panic ensues, due to a threat) (in minutes).
	<div style="border: 1px solid #0070C0; padding: 5px;"><p><b>NOTE</b></p><p>If Panic Duration is less than the amount of time it takes civilians to get to a Safe Zone (see <a href="#">Define Pedestrian Flows (on page 141)</a>), the civilians stop panicking before reaching the Safe Zone.</p></div>
<b>Threat Detection</b>	Defines how panic starts, based on the detected threat. Has the following options: <ul style="list-style-type: none"><li>• <b>All populace reacts to the threat</b></li><li>• <b>Only populace within hearing range reacts to threats</b></li><li>• <b>Only populace within hearing range or line of sight reacts to threats</b></li></ul>
<b>Flee Distance</b>	Defines the distance (in meters) from the threat to which pedestrians flee in panic.
	<div style="border: 1px solid #0070C0; padding: 5px;"><p><b>NOTE</b></p><p>If Flee Distance is smaller than the distance to a Safe Zone (see <a href="#">Define Pedestrian Flows (on page 141)</a>), some civilians hide in the Safe Zone, while others only rely on Flee Distance.</p></div>
<b>Number of Vehicles</b>	Number of vehicles in the population.
	<div style="border: 2px solid #C00000; padding: 5px;"><p><b>WARNING</b></p><p>If you add vehicles to your civilian population (<b>Number of Vehicles</b> is greater than 0), you must define <b>Vehicle Spawn</b> and <b>Vehicle Despawn</b> activities. Also, <b>Vehicle Spawn</b> and <b>Vehicle Despawn</b> activities must be placed on roads. Placing them elsewhere does not create vehicle traffic. For more information, see <a href="#">Spawn / Despawn (on page 195)</a> activities.</p></div>

Population Setting	Description
Shape	Population area shape on the map. <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"><span style="color: #0070C0; font-size: 1.5em;">i</span> <b>NOTE</b> Only <b>Circle</b> is available.</div>
Radius	Radius of the population area (in meters).
Civilian AI Identifier	Population color identifier. Different populations can have different roles, and color identifiers are used to distinguish between these roles. <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"><span style="color: #0070C0; font-size: 1.5em;">✓</span> <b>TIP</b> Click the <b>Information</b> icon for an explanation of how a population can only perform activities (see <a href="#">Define Pedestrian Flows (on page 141)</a>), defined using the same color identifier. </div>
Smart Objects	Check this to automatically create relevant population activities for any Smart Objects placed on the map. For a full list of Smart Objects, see <a href="#">Smart Objects (on page 200)</a> . <div style="border: 2px solid red; padding: 10px; margin-top: 10px;"><span style="color: red; font-size: 1.5em;">⚠</span> <b>WARNING</b> To automatically create relevant population activities for your Smart Objects, make sure to place them first, before creating the population.</div>

### 9.8.1.1 Range Visualization

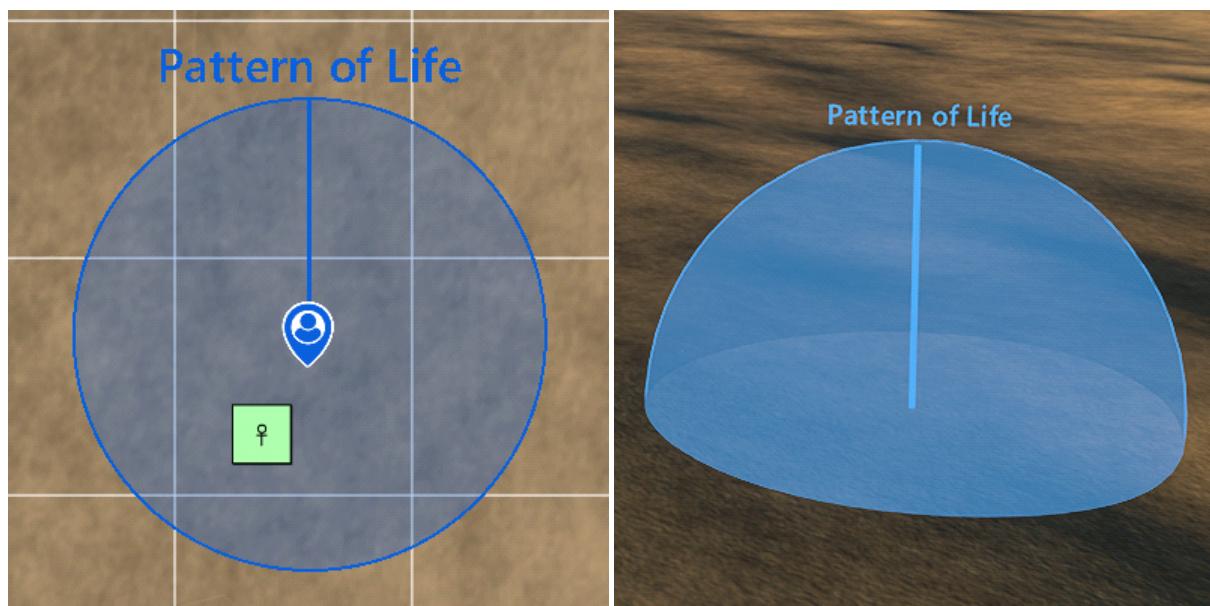
In VBS4, the **Radius** of the population area can be displayed as a range visualization.

#### NOTE

Use the Range Visibility Settings in the VBS4 Administrator Manual to enable / disable the entire range visualization or individual elements of it.

Range visualizations can only be seen by scenario Administrators and Instructors, not Trainees.

Image-78: 2D and 3D range visualization



## 9.8.2 Activity Editor Object

The Activity Editor Object (appears as **Control AI - Activity** in the Editor (Prepare / Execute Mode) Editor Objects List - see [Control AI UI Overview \(on page 17\)](#)) defines the population activities (for a given population - see [Population Editor Object \(on page 189\)](#) and [Define Populations \(on page 133\)](#)).

### 9.8.2.1 Activity Settings

It consists of the following settings:

Setting	Description
<b>Name</b>	Activity name.  <b>NOTE</b> The activity <b>Name</b> setting cannot be used as an Editor Object variable name.
<b>Type</b>	The following activity types are available for Civilian AI: <ul style="list-style-type: none"><li>• <a href="#">Spawn / Despawn (on page 195)</a></li><li>• <a href="#">Generic (on page 196)</a></li><li>• <a href="#">Interrupt (on page 196)</a></li><li>• <a href="#">Area (on page 198)</a></li></ul> <b>WARNING</b> Civilian children do not perform any activities, apart from the <b>Start Panic</b> activity.
<b>Civilian AI Identifier</b>	Activity color identifier.  <b>NOTE</b> Has to match the population <a href="#">Civilian AI Identifier (on page 135)</a> (by default, it matches the color of the latest created population). Use the color to distinguish one population from another, if each has different roles in the simulation.
<b>Sequence Number</b>	Sequence step number for generic activities (for example, an activity with a sequence number of 3 cannot begin before an activity with a sequence number of 2).  <b>NOTE</b> Only works with <b>Generic</b> activities.

Setting	Description
<b>Capacity</b>	<p>Maximum number of entities to perform the activity.</p> <p>The following considerations apply:</p> <ul style="list-style-type: none"> <li>• The automatic Control AI reservation system checks the activity capacity, before entities start moving towards the activity. This is done to avoid the problem of having entities arriving to the activity area / position, when the capacity is full.</li> <li>• For <b>Safe Zone</b> activities, entities cluster about the activity area / point in parallel.</li> <li>• For all other activity types, one entity performs the activity at a time, the rest form a queue, where:</li> </ul> <pre><code>queue length = capacity - 1</code></pre> <p>The -1 is due to one entity performing the activity.</p> <ul style="list-style-type: none"> <li>• If the activity is linked to an area (see <a href="#">Create Areas (on page 144)</a>), no queue is formed, and all the civilian entities up to the defined capacity can perform the activity at the same time, at a random position within the defined area.</li> </ul>
<b>Priority</b>	<p>Activity priority. Can be:</p> <ul style="list-style-type: none"> <li>• <b>Optional</b> - The activity is optional and does not have to be performed.</li> <li>• <b>Mandatory</b> - The activity must be performed.</li> </ul> <p>The following considerations apply:</p> <ul style="list-style-type: none"> <li>• Only works with <b>Generic</b> activities.</li> <li>• Civilians always try to perform at least one <b>Mandatory</b> activity in a sequence step (see <a href="#">Sequence Number (on the previous page)</a>).</li> <li>• If there are only <b>Optional</b> activities in a sequence step, the entities skip the sequence step.</li> <li>• If there are also <b>Mandatory</b> activities in the sequence step, the entities cannot proceed, until they complete a <b>Mandatory</b> activity.</li> <li>• If the <b>Mandatory</b> activities are at full <a href="#">Capacity (above)</a> or locked (if the activity is linked to a trigger - see <a href="#">Initiating Activities (on page 166)</a>), the entities keep performing <b>Optional</b> activities in the same sequence step, until a <b>Mandatory</b> one becomes available.</li> <li>• If the <b>Mandatory</b> and <b>Optional</b> activities are at full <a href="#">Capacity (above)</a> or locked (if the activity is linked to a trigger - see <a href="#">Initiating Activities (on page 166)</a>), the entities keep performing <b>Idle</b> (free loitering) activities in the same sequence step, until a <b>Mandatory</b> or an <b>Optional</b> activity becomes available.</li> </ul>

Setting	Description
Create Area	<p>Check <b>Create Area</b> to create an area, where the activity takes place.</p> <p><b>NOTE</b> For activities that require the presence of an area, this option is automatically checked.</p> <p>This creates an area (<a href="#">Area Editor Object (on page 199)</a>) with the default size of 25 meters x 25 meters (to change the area size, see <a href="#">Edit Areas (on page 147)</a>), linked to the Activity Editor Object.</p> <p>Alternatively, you can create the activity and the area separately, and then link the former to the latter. For more information, see <a href="#">Create Areas (on page 144)</a>.</p>

### 9.8.2.2 Activity Types

The following activity types are available for Civilian AI:

- [Spawn / Despawn \(on the next page\)](#)
- [Generic \(on page 196\)](#)
- [Interrupt \(on page 196\)](#)
- [Area \(on page 198\)](#)

Each **Type** has several **Activity** sub-types.

#### **WARNING**

Civilian children do not perform any activities, apart from the **Start Panic** activity.

### 9.8.2.2.1 Spawn / Despawn

Locations where pedestrians or vehicles appear and disappear.

#### NOTE

The **Spawn / Despawn** activity type has the following considerations:

- The **Spawn / Despawn** are the first activities you need to create, for civilian entities to appear and disappear in your simulation.
- If no **Spawn / Despawn** activities are created, and only the [Population Editor Object \(on page 189\)](#) is placed on the map, only pedestrians spawn anywhere within the population [Radius \(on page 135\)](#) and never despawn, while vehicles never spawn, even if vehicle models are defined for the given population.

#### Spawn / Despawn Activities:

- **Spawn** - Pedestrian spawn point.
- **Despawn** - Pedestrian despawn point.
- **Vehicle Spawn** - Vehicle spawn point.
- **Vehicle Despawn** - Vehicle despawn point.

#### WARNING

Any **Vehicle Spawn** activity requires the presence of a **Vehicle Despawn** activity. Also, **Vehicle Spawn** and **Vehicle Despawn** activities must be placed on roads. Placing them elsewhere does not create vehicle traffic.

#### NOTE

**Spawn / Despawn** activities can be optionally linked to an area (see [Create Area \(on the previous page\)](#)).

### 9.8.2.2 Generic

Any generic crowd activity.

Generic activities form the backbone of the pattern of life simulation, and are used in a sequenced flow, which is determined by the [Sequence Number \(on page 192\)](#), [Capacity \(on page 193\)](#), and [Priority \(on page 193\)](#) of each sequence step activity.

#### Generic Activities:

##### NOTE

The **Generic** activity is performed differently, depending on the defined [Capacity \(on page 193\)](#) and an optionally linked area (see [Create Area \(on page 194\)](#)).

- **Loiter** - The civilians loiter around.
- **Buy Item** - Plays a stand or shop counter buy animation.
- **Buy Item (Vending Machine)** - Plays a vending-machine buy animation.
- **Throw Away Trash** - Plays an animation of throwing away trash.
- **Sit Down** - Plays an animation of sitting down.
- **Look At** - Plays an animation of looking at something.

### 9.8.2.3 Interrupt

Any activity that interrupts other activities.

##### NOTE

All **Interrupt** activities are always automatically linked to an area (see [Create Area \(on page 194\)](#)).

#### Interrupt Activities:

##### NOTE

The **Interrupt** activity is always automatically linked to an area (see and [Create Area \(on page 194\)](#)). **Start Panic** and **Move To / Run To** activities can be also attached to moving entities.

- **Start Panic** - Makes civilians run in panic.

**NOTE**

The following considerations apply:

- When panic is active, pedestrians flee, while vehicles stop.
- Panic is automatically activated as a result of a shooting or an explosion.

For more information, see the following population settings for panic: [Panic Duration \(on page 134\)](#), [Threat Detection \(on page 134\)](#), [Flee Distance \(on page 135\)](#).

- **Gather Crowd** - Causes the civilian entities to stop whatever they are doing, and look at a location.

**NOTE**

The following considerations apply:

- To look at the specified location, the civilian entities first need to be in an area, linked to the activity (see [Create Area \(on page 194\)](#)). To force the civilian entities to move to the area, use either the **Walk To** or **Run To** activities.
- Once the civilian entities are in the area, they assume a random position in it, and look at the specified location.
- The crowd continues to look at the location, until the **Gather Crowd** activity is removed or disabled with a trigger (see [Initiating Activities \(on page 166\)](#)).
- Civilians cannot gather on roads.

- **Walk To** - Causes the crowd to stop whatever they are doing, and walk to a certain location.

**NOTE**

Once the civilians arrive to the location, they resume their other activities.

- **Run To** - Causes the crowd to stop whatever they are doing, and run to a certain location.

**NOTE**

Once the civilians arrive to the location, they resume their other activities.

For more information on using the **Interrupt** activity sub-type, see [Defining Responsive Behavior \(on page 163\)](#).

### 9.8.2.2.4 Area

Any activity that is area-defining.

#### Area Activities:

- **Restrict Access (Pedestrians)** - Keeps civilian pedestrians from entering an area (ignored by civilians in panic).

##### NOTE

The following considerations apply:

- The access restriction applies to all the civilian pedestrians of all the populations (see [Define Populations \(on page 133\)](#) defined in the scenario).
- This area-based activity is always automatically linked to an area (see [Create Area \(on page 194\)](#)).

- **Restrict Access (Vehicles)** - Keeps civilian vehicles from entering any roads that the area partially / fully overlaps (the roads are excluded from vehicle path-planning).

##### NOTE

The following considerations apply:

- The access restriction applies to all the civilian vehicles of all the populations (see [Define Populations \(on page 133\)](#) defined in the scenario).
- This area-based activity is always automatically linked to an area (see [Create Area \(on page 194\)](#)).

- **Safe Zone** - Designates where civilians hide, when panic sets in.

##### NOTE

It is not mandatory for this area-based activity to have an area linked to it. In the absence of an area, civilians gather around the activity position on the map, creating a crowd numbering up to the activity [Capacity \(on page 193\)](#).

## 9.8.3 Area Editor Object

The Area Editor Object (appears as **Control AI - Area** in the Editor (Prepare / Execute Mode) Editor Objects List - see [Control AI UI Overview \(on page 17\)](#)) defines an area within the general population area, created using the [Population Editor Object \(on page 189\)](#), which is used by pedestrian and vehicle traffic flows. For more information, [Define Pedestrian Flows \(on page 141\)](#) and [Define Traffic Flows \(on page 151\)](#).

It consists of the following settings:

Area Setting	Description
<b>Size (Left-Right)</b>	Area width (in meters).
<b>Size (Up-Down)</b>	Area height (in meters).

## 9.8.4 Smart Objects

Specific Editor Objects work as Smart Objects, or additional points of interest for civilians.

### TIP

You can automatically create the relevant activities for Smart Objects. For more information, see [Define Populations \(on page 133\)](#).

Add these objects in the population zone [Radius \(on page 135\)](#), to create alternate points of interest that civilians can interact with.

### WARNING

If you place the objects outside the population radius (see [Radius \(on page 135\)](#)), they are not functional. They only becomes functional, if you move them within the population radius.

### Follow these steps:

1. In the Editor Objects List, select **(F8) Objects**.
2. Right-click a placement position in the population zone, and select **New Object**.
3. In the Object Properties dialog, select any of the Smart Objects listed below to serve as a point of interest:
4. Click **OK**.

The Smart Objects are placed on the map.

### Generic Objects

Object Display Name	Object Class
Billboard - Arcadia	<code>Land_Bilboard_Bienvenudo</code>
Billboard - Banana King	<code>Land_Bilboard_Konstantin</code>
Billboard - blurred	<code>Land_Bilboard_HELLMART</code>
Billboard - For rent	<code>Land_Bilboard_work</code>
Billboard - Imholol	<code>Land_Bilboard_Ada</code>
Billboard - Riviera	<code>Land_Bilboard_Riviera</code>
Stands (search for stanek_3 or stanek_3B)	<code>Land_stanek_3</code> and <code>Land_stanek_3B</code>
Trashcan - concrete	<code>Land_Odpadkovy_kos</code>

## Customer Exclusive Objects

### ★ FEATURE NOTICE

These features are customer exclusive. For more information, contact [sales@bisimulations.com](mailto:sales@bisimulations.com).

Object Display Name	Object Class
Bench	land_fr_ts_bench
Info Screen	land_fr_ts_arrivalDepartureTV
Ticket Machine	land_fr_ts_ticket_machine
Trashbin	land_fr_ts_bin

# 10. Animal AI

You can add animal-herd movement to your scenario in VBS4 using the ([F3\) Waypoints Editor Object \(on page 18\)](#).



The following is discussed:

- [Creating a Herd \(below\)](#)
- [Reaction to Threats \(on page 204\)](#)

## 10.1 Creating a Herd

Add a Control AI animal herd to your mission.

**Follow these steps:**

1. In the Editor (Prepare Mode), select (**F1) Unit** in the Editor Objects List, and double-click a location on the map, where you want to place your animal herd.  
The Object Properties dialog opens.
2. In **Filters**, type **Animals**.  
The Object Properties dialog displays a list of all the available animals in VBS4.
3. Select an animal you want to place on the map.
4. Click **OK**.

- If you want to place more animals of the same type, press **LCtrl + C** and then **LCtrl + V** to copy and paste the animal entity. Otherwise, if you want to place animals of different types, repeat steps 3 - 5.

The animals are placed on the map.

- Select all the animals you placed, click the **RMB**, and select **Orders > Group Selected**.

The animals are grouped into a herd.

- To assign a destination to the animal herd, create a herd waypoint (see [Waypoints \(on page 23\)](#)) by right-clicking the herd group marker, selecting **Orders > Assign New Waypoint**, and clicking a position on the map, where the herd waypoint needs to be created.

- Set the following herd-waypoint settings:

Setting	Description
<b>Behavior</b>	Select the <b>Animal Herd Movement</b> behavior in the drop-down.
<b>Loiter Radius</b>	Radius (meters) of the circular area where the animals loiter.
<b>Loiter Timeout</b>	Time (seconds) between the first animal entering the circular loiter area, defined by the waypoint position and the <b>Loiter Radius</b> , and the waypoint completion. Any further chained waypoints are executed after the <b>Loiter Timeout</b> elapses. In case of no further waypoints, the current waypoint is completed but the animals keep loitering.
<b>Variable Name</b>	In <b>Variable Name</b> , enter the Order waypoint name, which can be used in SQF scripts.

- Set the Order waypoint completion settings:

Option	Description
<b>Condition to Complete</b>	Condition that needs to be fulfilled to complete the waypoint for the selected Order behavior.
<b>Code on Completion</b>	SQF code to execute on waypoint completion.

- Click **OK**.

- Preview the mission.

The animal herd starts moving.

### **WARNING**

Bohemia Interactive Simulations advises scenario Administrators not to change the **Advanced** settings for the behavior to work correctly.



## 10.2 Reaction to Threats

Animals automatically detect threats and react to them.

Animal threat detection works similarly to the [Civilian AI \(on page 132\)](#) threat detection. Animals react to shooting and explosions.

Threat reaction consists of two stages:

1. At first, the animals react individually by fleeing away from the detected threats.
2. Later (when situated at a safe distance from the threats, or after a certain period of time), animals move at full speed to a safe area, where they keep loitering, until a new **Animal Herd Movement** waypoint is assigned to the herd, or until the current waypoint is updated or moved.

# 11. Navigation Meshes

The most essential feature of Control AI entities is movement. To determine how to get from one point to the next, the Control AI uses navigation meshes. A navigation mesh is composed of convex polygons that cover traversable areas on the terrain, including both interiors and exteriors. To see why a Control AI entity moves in a specific way, you can visualize its navigation meshes (see [Control AI Visualization \(on page 210\)](#)).

The Control AI uses two navigation meshes, with differing levels of detail.

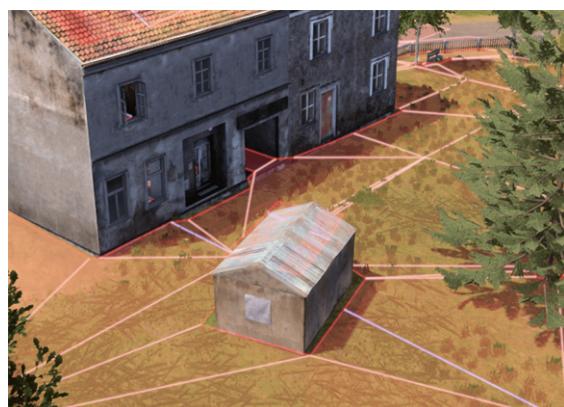
- **Coarse Navigation Mesh** - Only considers very large impassable obstacles, such as bodies of water, cliff faces, or ravines. A Control AI entity uses this navigation mesh to construct a high-level path to the destination.
- **Detailed Navigation Mesh** - Includes all the details about the environment, such as individual trees, walls, or building interiors. This navigation mesh is generated at runtime, and for performance reasons, only in the immediate vicinity of the entity, and some distance ahead along the high-level path. The detailed mesh is used to construct a detailed path that determines the specific steps needed to navigate through the environment. It only contains static obstacles, such as buildings, trees, or stationary vehicles. Therefore, after a Control AI entity finds a path using the navigation mesh, that entity needs to avoid other entities, which may have planned their paths through the same points. The technique to avoid other non-static entities (people or vehicles) is called collision avoidance.

There are two types of detailed navigation mesh:

- **Humanoid Navigation Mesh** - Used for path-planning by Control AI entities on foot.
- **Vehicle Navigation Mesh** - Used for vehicle path-planning that ignores roads, when the Control AI entity drives a vehicle. Vehicles are able to drive through "soft objects" (for example, bushes), which are not part of the navigation mesh.

To manually update a detailed navigation mesh, to reflect changes made in VBS Editor (Execute Mode), see [Updating the Navigation Mesh at Runtime \(on page 208\)](#).

An example navigation mesh, using [Visible Nav Mesh \(on page 215\)](#) and [Generate Nav Mesh \(on page 218\)](#) options.



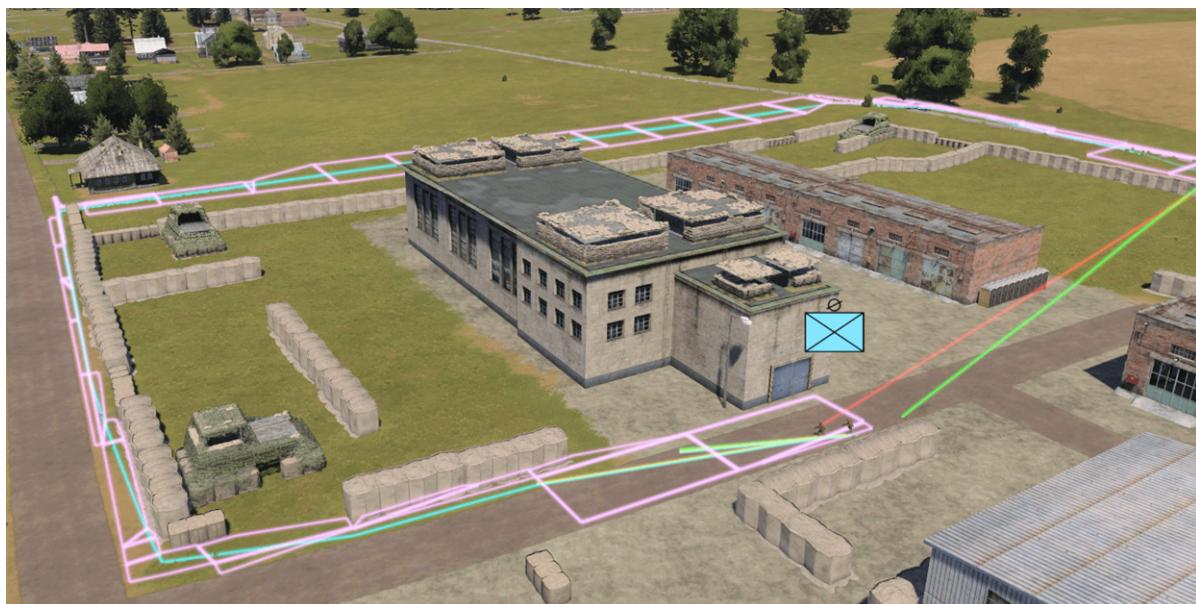
## Navigation Mesh Limitations

An entity may be unable to find a path around very large obstacles.

The following situations might occur:

- A path around a long wall to the other side, where the entity stops before the wall, at the closest spot to its destination.
- A path that involves moving out of a big area which is completely surrounded by impassable walls, objects, and other buildings, so there is only one way from this complex (such as a military base, big garden / park, or cemetery, enclosed with fences).

The following image illustrates moving outside a complex of buildings and walls:



The following image illustrates that the exit from an enclosed area is too far away to be included in the search for a path around the coarse navigation mesh path (green line), during the move:



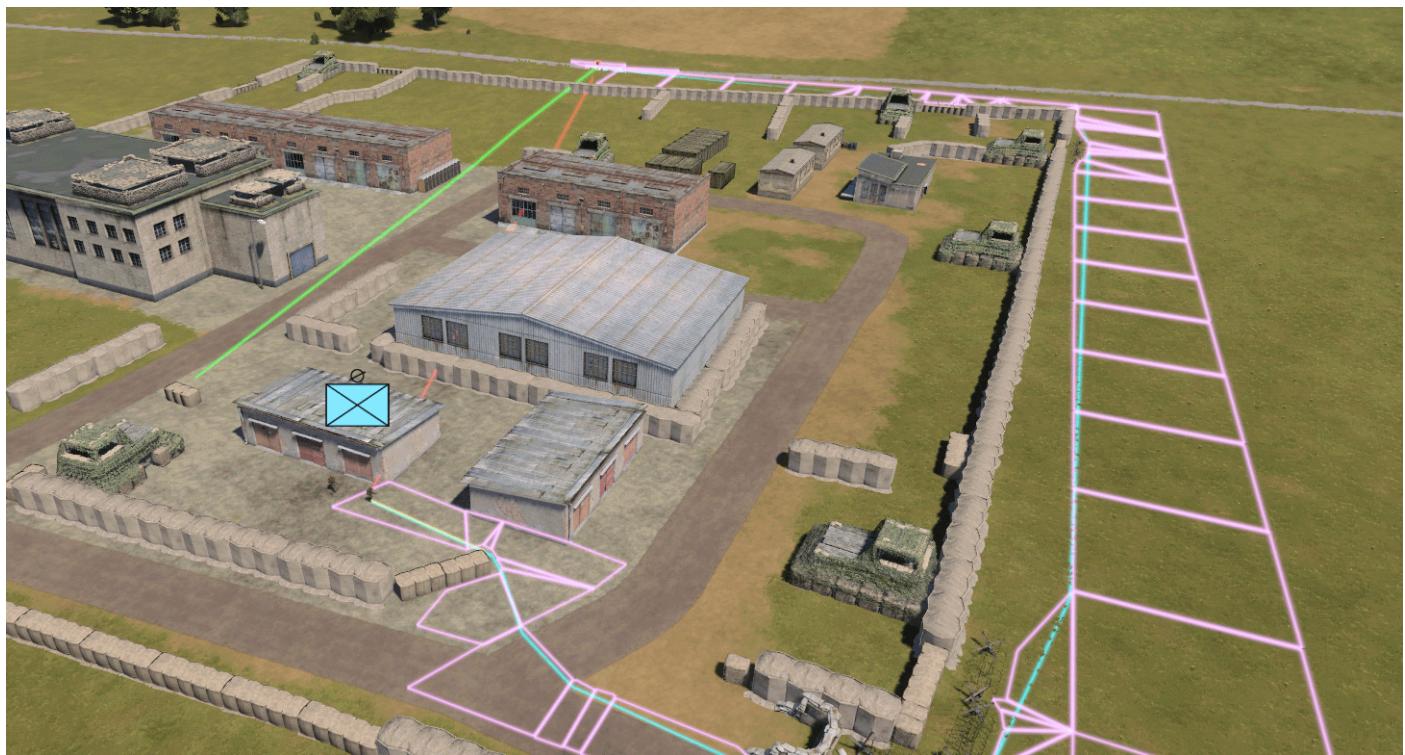
The following image illustrates that the obstacle is too big, and the entity cannot find a path around it, moving to the closest spot to its destination:



The workaround to the situation illustrated in the previous image is to place more Waypoints, to guide the entity around the obstacle (see [Waypoints \(on page 23\)](#) in the VBS Control AI Manual):



A different workaround is to create more entrances in the enclosed area. For example, you can achieve this by deleting a part of the wall:



## 11.1 Updating the Navigation Mesh at Runtime

You can manually update a part of a detailed navigation mesh (humanoid or vehicle) at runtime, using the [markNavmeshOutdated](#) (<https://sqf.bisimulations.com/display/SQF/markNavmeshOutdated>) SQF command.

### **i** NOTE

Since the navigation mesh updates automatically, running the command is optional and allows to manually force the update, if necessary.

#### Follow these steps:

1. Perform your desired changes in VBS Editor (Execute Mode) in an area, where a detailed navigation mesh is already generated.

### **i** NOTE

The detailed navigation mesh is generated as Control AI entities move through an area.

### **✓** TIP

You can use the **AI Debug Panel** (see [Control AI Visualization \(on page 210\)](#)) to see where a detailed navigation mesh is generated.

2. In VBS Editor (Execute Mode) menu, navigate to **Tools > Developer Console**.

3. In the **Code Field**, enter the following SQF code:

```
markNavmeshOutdated [2000, 10000, 4000, 12000];
```

4. Click **Execute**, to execute the code.

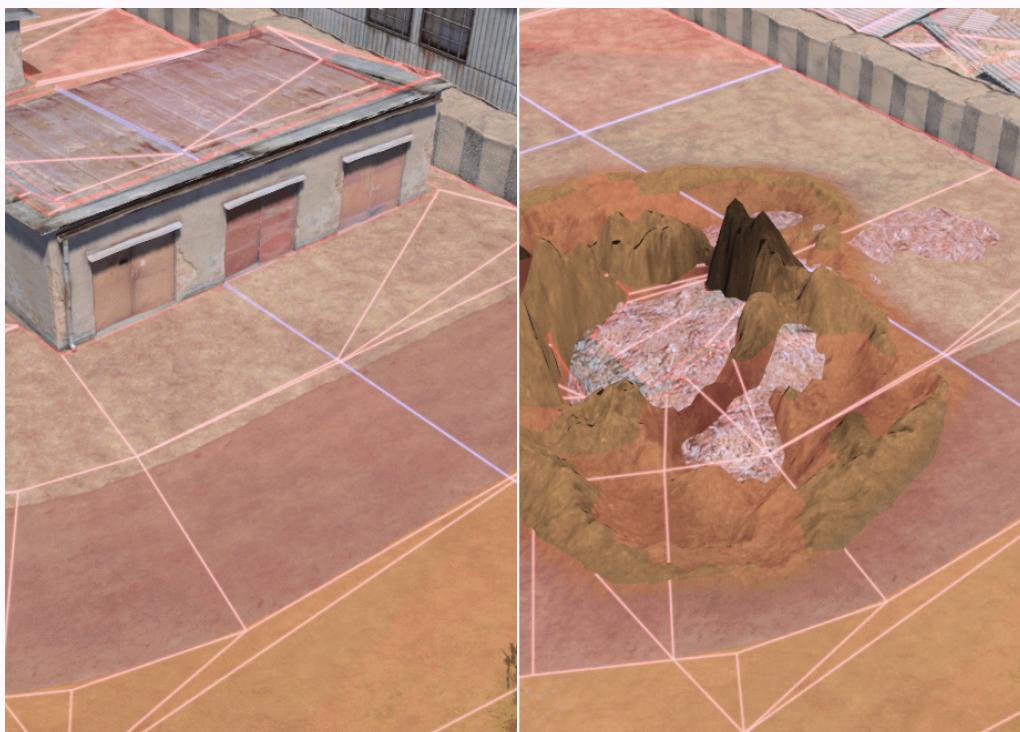
Observe that the detailed navigation mesh is updated the next time a Control AI entity moves through the area.

#### NOTE

Both humanoid and vehicle detailed navigation meshes are updated together.

#### EXAMPLE

Navigation mesh before (left image) and after (right image) marking an inset, containing a destroyed building, as outdated:



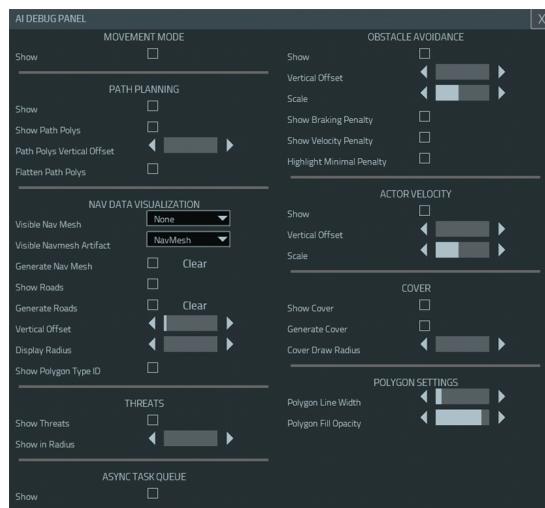
# 12. Control AI Visualization

Control AI has several visualization options available to Control AI users and developers.

For civilian debug visualization, see [Debug Visualization \(on page 168\)](#).

These visualization options, such as path-planning and navigation-mesh visualizations, for both military and civilian Control AI, are available in the AI Debug Panel.

To open the AI Debug Panel, in the VBS4 Editor, open the menu **Control AI > AI Debug Panel**.



To hide all debug visualizations, in the VBS Editor main menu, select **Control AI > Hide AI Debug Panel**. To switch them on again, select **AI Debug Panel**.

You can visualize the following:

- [Movement Mode \(on the next page\)](#) - Shows the Behavior Tree (BT) movement node that is currently running in the entity behavior (if there is any).
- [Path-Planning \(on page 212\)](#) - Visualizes Control AI entities, when they are path planning.
- [Navigation Mesh \(on page 214\)](#) - Visualizes navigation meshes, used by humanoid and vehicle entities, when navigating around the terrain (for more information, see [Navigation Meshes \(on page 205\)](#)).
- [Threats \(on page 220\)](#) - Visualizes various threats to Control AI entities.
- [Cover \(on page 221\)](#) - Visualizes places on the terrain, where Control AI entities can take cover.
- [Async Task Queue \(on page 223\)](#) - Controls asynchronous visualization tasks for path-planning, navigation mesh updates, or cover queries.
- [Obstacle Avoidance \(on page 223\)](#) - Visualizes obstacles, so that they can be avoided by Control AI entities.

- [Actor Velocity \(on page 225\)](#) - Visualizes actor (Control AI entity) velocity, when dismounted from a vehicle, or when in a wheeled / tracked vehicle.

## 12.1 Movement Mode

Shows the movement Behavior Tree (BT) node that is currently running in the entity behavior (if there is any).

For more information on BT nodes, see Behavior Tree Node Reference in the VBS Control Editor Manual.

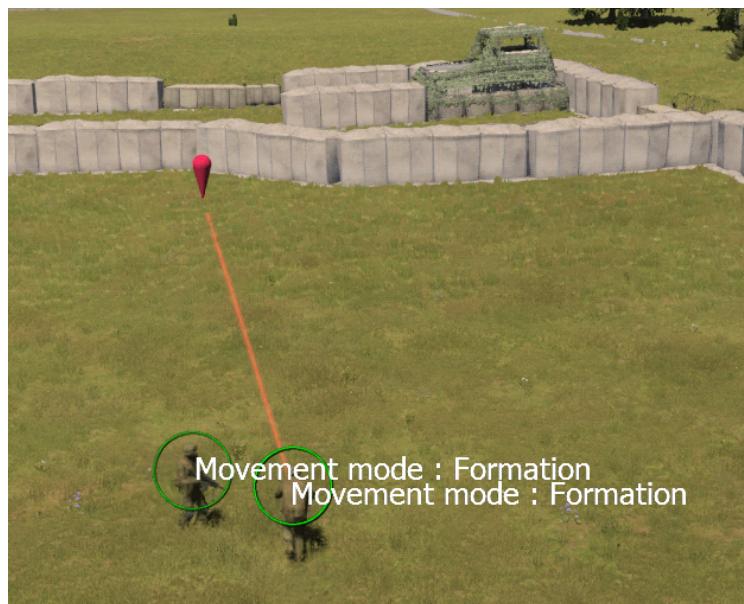
### NOTE

The settings in this section allow you to display the visualizations only for selected Control AI entities. If no Control AI entities are selected, the visualizations are displayed for all Control AI entities.

The BT node visualizations can be:

- **Empty** - If no movement is happening.
- **Position** - When a **Move** BT node is running (when the entity is trying to reach a specific destination).
- **Formation / Follow Entity / Follow Entity With Offset / Vehicle Parking** - If one of these BT nodes is running.

**Image-79: Formation movement mode**



## Follow these steps:

1. Select the menu option **Control AI > AI Debug Panel**.
2. In the **Movement Mode** section of the panel, set the following settings:
  - **Show (below)**

### 12.1.1 Show

Shows / hides the movement-mode visualization.

## 12.2 Path-Planning

An entity path is either a planned path on the navigation mesh, or, if the entity is following a moving formation and there is no obstruction between its desired position and itself, the movement target that is displayed.

### NOTE

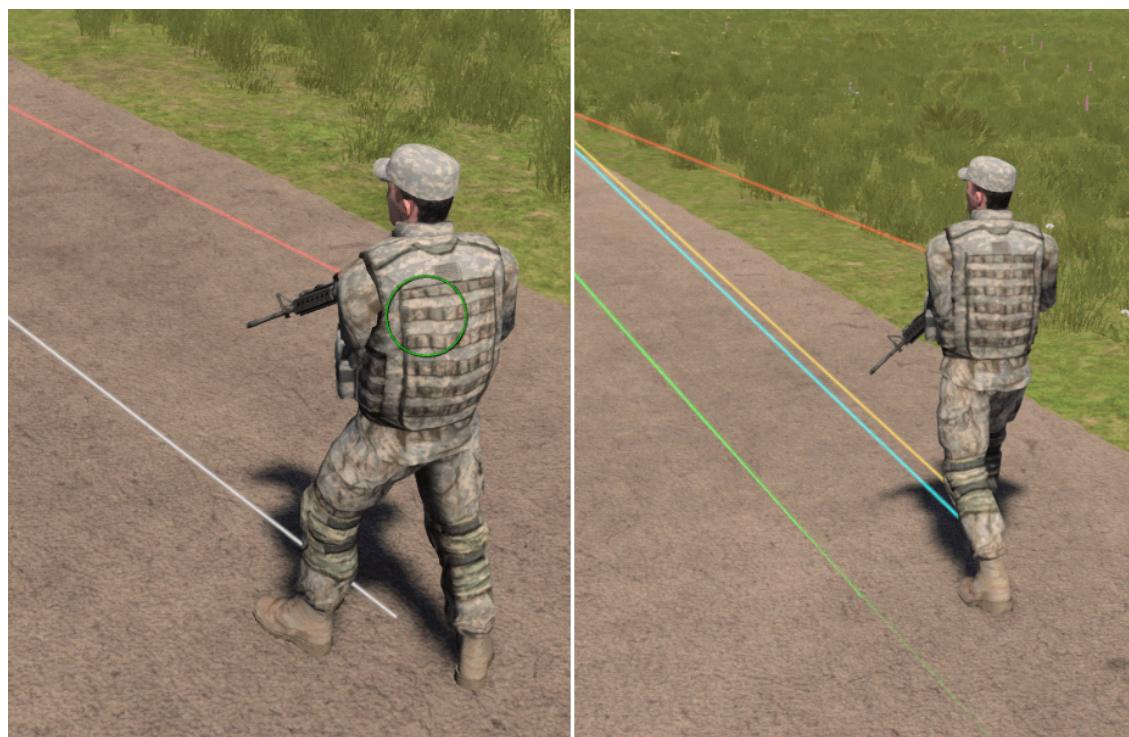
The settings in this section allow you to display the visualizations only for selected Control AI entities. If no Control AI entities are selected, the visualizations are displayed for all Control AI entities.

The path is visualized in the 3D view, using the following lines:

- **White line** - A straight line to the destination, if path-finding is in progress.
- **Green line** - Represents the high-level (coarse) path to the destination, planned on the coarse navigation mesh.
- **Cyan line** - Represents the detailed path, planned on the detailed navigation mesh in the immediate vicinity of the entity.
- **Yellow line** - A straight line to the first corner of the detailed path.

### NOTE

The red line is not part of the path-planning visualization.

**Image-80: White line (left), and green, cyan, and yellow lines (right)****Follow these steps:**

1. Select the menu option **Control AI > AI Debug Panel**.
2. In the **Path Planning** section of the panel, set the following settings:
  - [Show \(below\)](#)
  - [Show Path Polys \(on the next page\)](#)
  - [Path Polys Vertical Offset \(on the next page\)](#)
  - [Flatten Path Polys \(on the next page\)](#)
3. (Optional) Change the polygon line width and opacity for path-planning polygons in the [Polygon Settings \(on page 229\)](#).

### 12.2.1 Show

Shows / hides the path-planning visualization.

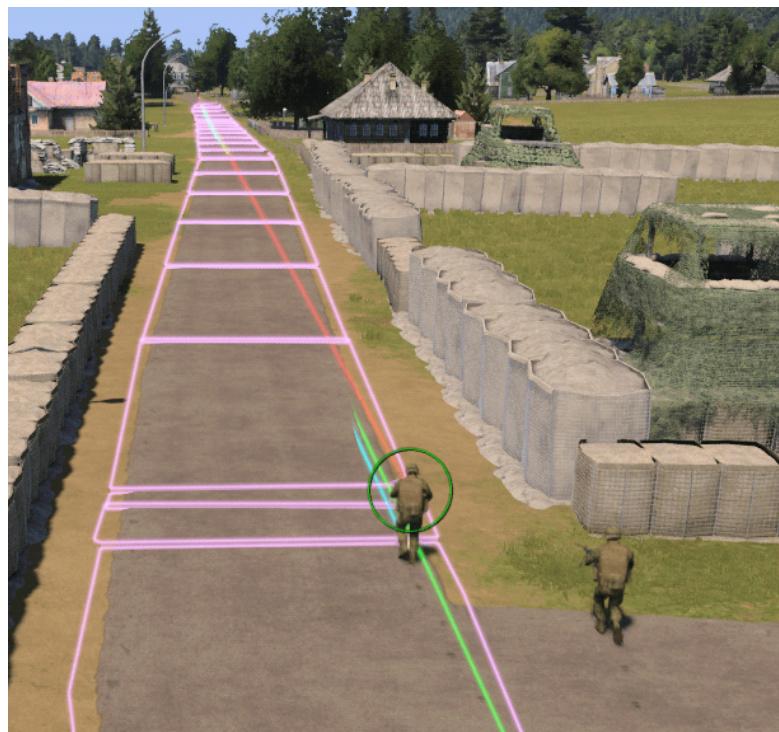
## 12.2.2 Show Path Polys

Shows / hides all the navigation mesh polygons that the path is traversing.

### **NOTE**

The polygons are visible only if the path is found after this checkbox is enabled.

**Image-81: Path polygons and a planned path**



## 12.2.3 Path Polys Vertical Offset

A slider that can move the navigation-mesh polygons upwards in cases where the polygons do not copy the terrain precisely and may clip into it.

## 12.2.4 Flatten Path Polys

Makes sure that neighboring polygons are not slightly vertically offset from each other in an alternating fashion.

## 12.3 Navigation Mesh

Both the coarse and detailed navigation mesh can be displayed in 3D view.

This visualization shows the navigation-mesh polygons in a certain radius from the camera.

- The edges of impassable areas are colored red.
- Tile borders are colored blue.

- Polygon edges are colored pink.

#### Follow these steps:

1. Select the menu option **Control AI > AI Debug Panel**.
2. In the **Nav Data Visualization** section of the panel, set the following settings:
  - [Visible Nav Mesh \(below\)](#)
  - [Visible Nav Mesh Artifact \(on page 217\)](#)
  - [Generate Nav Mesh \(on page 218\)](#)
  - [Show Roads \(on page 218\)](#)
  - [Generate Roads \(on page 219\)](#)
  - [Vertical Offset \(on page 219\)](#)
  - [Display Radius \(on page 219\)](#)
  - [Polygon Type ID \(on page 219\)](#)
3. (Optional) Change the polygon line width and opacity for navigation-mesh polygons in the [Polygon Settings \(on page 229\)](#).

### 12.3.1 Visible Nav Mesh

Shows the navigation mesh. The navigation mesh can be of the following types:



#### WARNING

If the navigation mesh is not yet generated, make sure that [Generate Nav Mesh \(on page 218\)](#) is checked. Otherwise, the visualization does not get displayed.

- **None** - No navigation mesh is shown.
- **Humanoid** - Shows the navigation mesh for humanoid entities.
- **Vehicle** - Shows the navigation mesh for vehicle entities.
- **Coarse** - Shows the coarse navigation mesh.

**Image-82: The navigation mesh designated using color-filled polygons**

When **Visible Nav Mesh** is on, the following information is also displayed, based on the navigation-mesh types:

**NavMesh tile count:**  
**humanoid: 19**  
**vehicle: 4**  
**coarse: 1**

**NOTE**

Unit and vehicle entities use Clearance, indicating which entities can pass through a particular terrain segment. A Clearance level is visualized using the letter **C**, a level number, and a specific color. The higher the level, the bigger the entity that can pass through. Unit entities only have one Clearance level.

**Image-83: C1, C2, and C3 Clearance levels, with different polygon colors**

## 12.3.2 Visible Nav Mesh Artifact

This setting allows you to only visualize a specific part (artifact) of the navigation mesh.

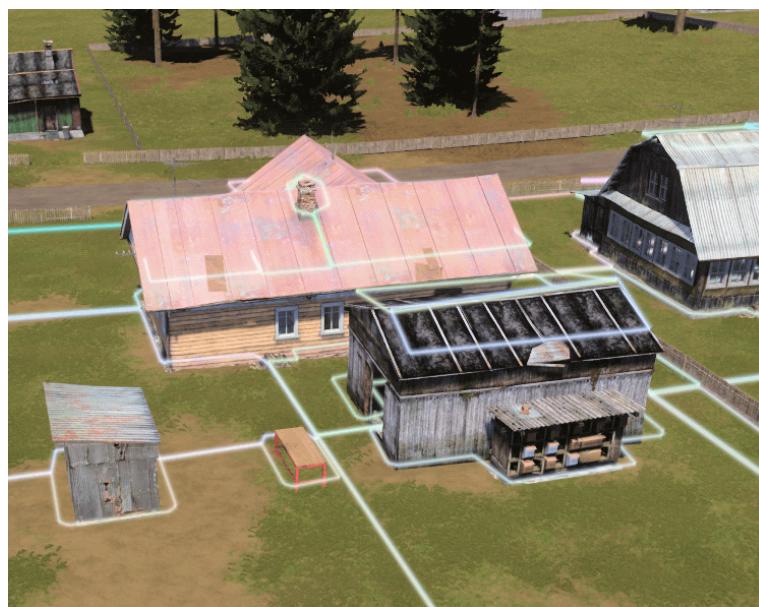
### **WARNING**

If the navigation mesh is not yet generated, make sure that [Generate Nav Mesh \(on the next page\)](#) is checked. Otherwise, the visualization does not get displayed.

The available artifacts are:

- **NavMesh** - The final navigation mesh.
- **Heightfield** - Voxels containing spans that completely encompass the surface of all polygons in the source geometry.
- **Compact Heightfield** - The tops of the heightfield, representing the potential traversable surfaces on top of the solid space.
- **Contours** - Polygons with simplified edges, created from raw contours.
- **Heightfield Walkable** - Represents walkable areas.
- **Compact Heightfield Regions** - Represents compact regions of walkable areas.
- **Raw Contours** - Highly detailed polygons, generated from the regions.
- **Region Connections** - Neighboring regions to which entities can move.
- **Poly Mesh** - Same as **NavMesh**.
- **Poly Mesh Detail** - Triangulated contour polygons.

**Image-84: Contours visualization**



### 12.3.3 Generate Nav Mesh

When this setting is turned on, both detailed and coarse navigation mesh are automatically generated inside the [Display Radius \(on the next page\)](#) around the 3D camera.

When this setting is turned off, only the already generated navigation mesh is displayed.

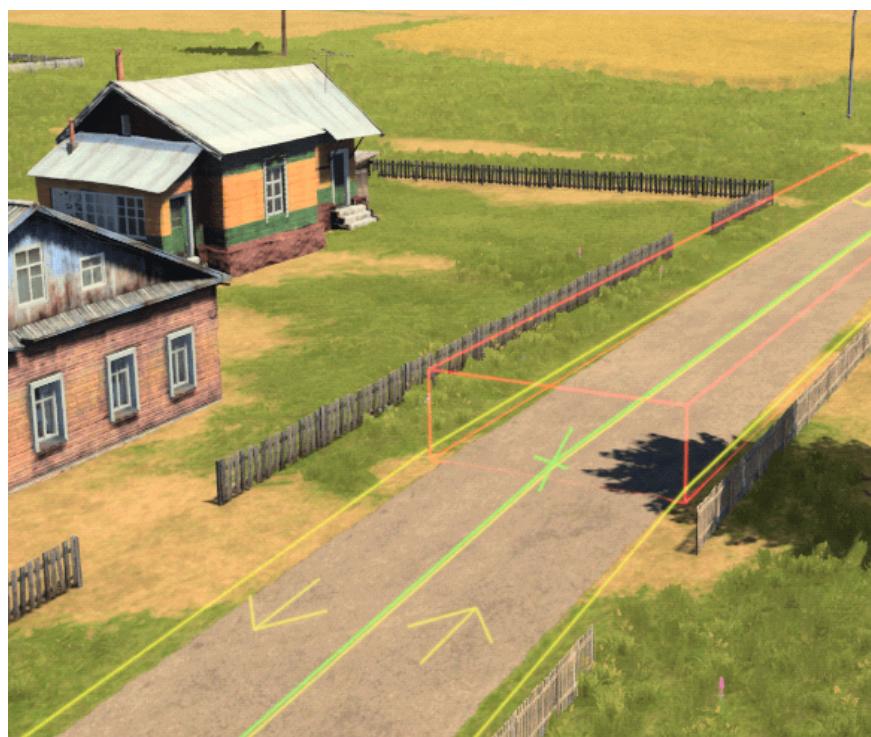
Click **Clear** to manually reset the navigation-mesh data for further navigation-mesh visualizations.

### 12.3.4 Show Roads

Shows / hides the road network data:

- Green and red lines represent the road network. Green lines represent connected segments, and red lines represent unconnected road segments.
- Yellow arrows represent the direction of travel for right-hand traffic.
- Yellow lines represent lane boundaries.
- 3D visualization options used in civilian AI visualizations. For more information, see [Debug Visualization \(on page 168\)](#).

**Image-85: Show Roads enabled**



Use the **Show Roads** and [Generate Roads \(on the next page\)](#) options, when creating vehicle traffic.

For additional traffic visualization options, see [Traffic Debug Visualization \(on page 159\)](#).

## 12.3.5 Generate Roads

When this setting is turned on, the road network is automatically generated inside the [Display Radius \(below\)](#) around the 3D camera.

When this setting is turned off, only the already generated road network is displayed.

Click **Clear** to manually reset the road-network data for further road visualizations.

## 12.3.6 Vertical Offset

Changes the vertical offset of the navigation mesh and road network, if it is clipping into the terrain.

## 12.3.7 Display Radius

Adjusts the display radius (between 50 - 300m) of the visualized navigation mesh and road network.

## 12.3.8 Polygon Type ID

The polygon type. Can be:

- **D** - Default - Lacks safety.
- **S** - Safe - Preferred Polygons.
- **R** - Roads.
- **F** - Footpath. Currently, in combination with the **R** flag, it means a crosswalk. In the future, it is expected the **F** flag will be used for sidewalks, trails, and other surfaces designated for pedestrians.
- **C** - Clearance level, indicating which entities can pass through a particular terrain segment. A Clearance level is visualized using the letter **C**, a level number, and a specific color. The higher the level, the bigger the entity that can pass through.

### **i** NOTE

A polygon can be of more than one type. The only exception is "**DS**", since Dangerous and Safe are mutually exclusive.

## 12.4 Threats

You can display threats in the given radius from the current position in 3D view.

**Follow these steps:**

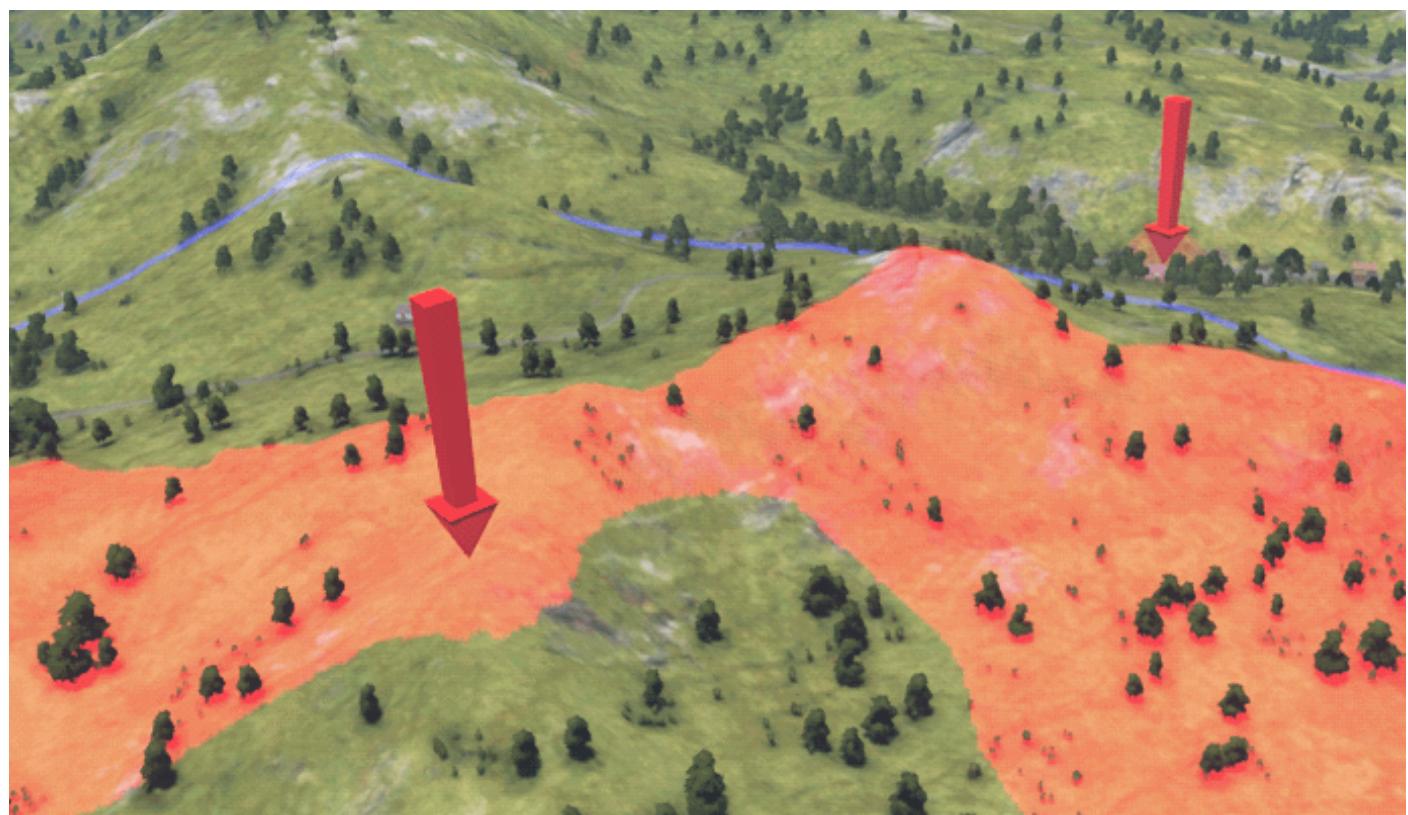
1. Select the menu option **Control AI > AI Debug Panel**.
2. In the **Threats** section of the panel, set the following settings:
  - Show Threats (below)
  - Show in Radius (on the next page)

### 12.4.1 Show Threats

Shows / hides threats.

- The area under threat shows as a grid of red lines.
- The point on the threat grid always consists of at least 2 red lines. The vertical line means that the area is under threat, while the other lines point to the threat (each line points only to one threat).
- Threat radius shows as a blue circle around the threat.

**Image-86: Threats visualized**



## 12.4.2 Show in Radius

The radius (50 - 300m) at which the threat is visualized.

**NOTE**

A bigger radius may affect performance.

## 12.5 Cover

You can display the cover which is available within a certain radius of the current position in 3D view.

**NOTE**

The simulation must be running for the cover visualization to work.

### Follow these steps:

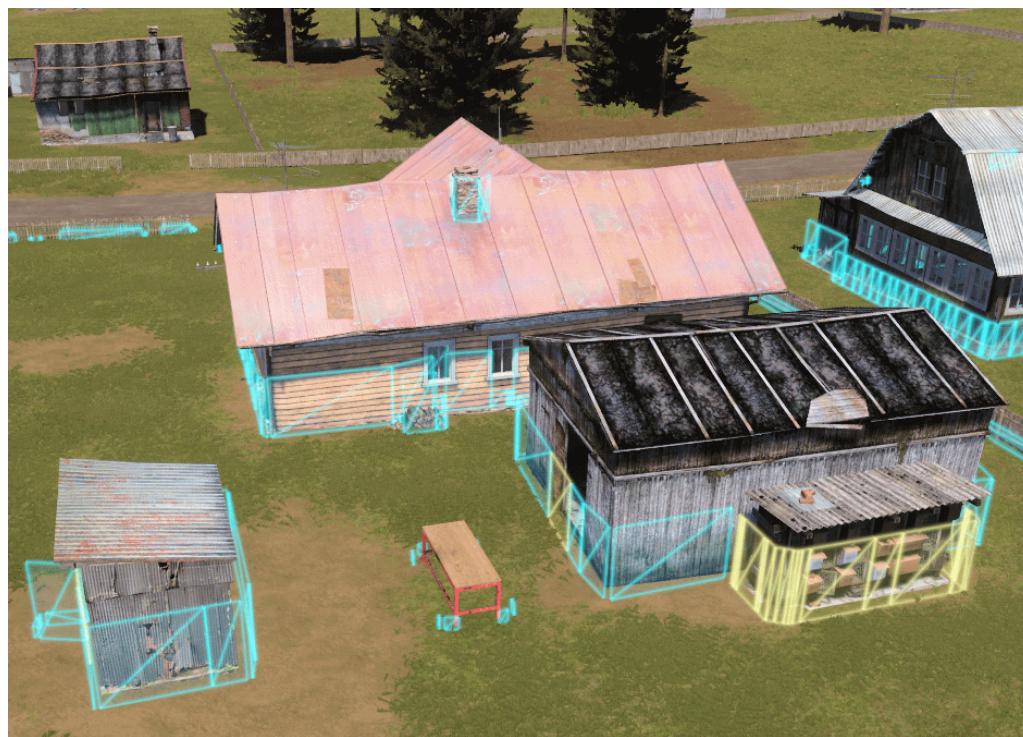
1. Select the menu option **Control AI > AI Debug Panel**.
2. In the **Cover** section of the panel, set the following settings:
  - [Show Cover \(on the next page\)](#)
  - [Generate Cover \(on the next page\)](#)
  - [Cover Draw Radius \(on page 223\)](#)
3. (Optional) Change the polygon line width and opacity for cover polygons in the [Polygon Settings \(on page 229\)](#).

## 12.5.1 Show Cover

Shows / hides the cover visualization.

- The cover is displayed as blue rectangles. Concealment areas are displayed in yellow.
- The cover is not displayed immediately when you move the camera, it appears as it is generated.
- A color-filled cover indicates that you are viewing it from the front (a cover that is not color-filled indicates that you are viewing it from the back).
- A diagonal line from the bottom left corner to the top right corner indicates that you are viewing the cover from the front (a line from the bottom right corner to the top left corner indicates that you are viewing the cover from the back).

**Image-87: The visualized cover (blue) with concealment areas (yellow)**



## 12.5.2 Generate Cover

When this setting is turned on, cover is automatically generated inside the [Cover Draw Radius \(on the next page\)](#) around the 3D camera.

When this setting is turned off, only already generated cover is displayed.

## 12.5.3 Cover Draw Radius

The radius (50 - 300m) at which the cover is visualized.

### NOTE

Setting a high radius and frequently moving the camera may have an impact on performance.

## 12.6 Async Task Queue

You can visualize the number of asynchronous tasks such as path-planning, navigation mesh updates, or cover queries that are still unfinished and pending to be completed at any given time. A non-zero number can indicate how much time it takes for entities to start spawning using the [Control AI - Civilian Editor Object \(on page 20\)](#) in the VBS Control AI Manual.

**Follow these steps:**

1. Select the menu option **Control AI > AI Debug Panel**.
2. In the **Async Task Queue** section of the panel, toggle the **Show** setting, to show / hide the number of pending asynchronous tasks.

## 12.7 Obstacle Avoidance

You can visualize the obstacle avoidance of an actor (Control AI entity).

### NOTE

The settings in this section allow you to display the visualizations only for selected Control AI entities. If no Control AI entities are selected, the visualizations are displayed for all Control AI entities.

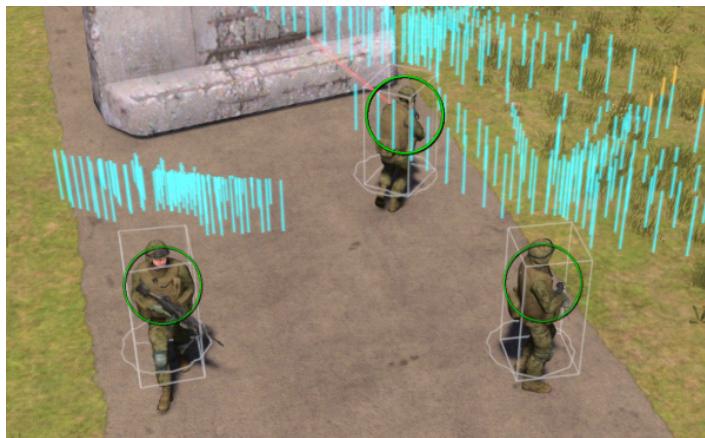
In order to determine the optimal velocity for following a planned path while avoiding obstacles, the following is done:

1. Velocities are sampled.
2. A penalty for each velocity is calculated.
3. The velocity with the lowest penalty is selected.

The penalty is displayed as a vertical bar, consisting of two components:

- **Braking Penalty (orange color)** - Proportional to the deceleration needed to avoid a collision.
- **Velocity Penalty (cyan color)** - Deviance from planned velocity.

### Image-88: Obstacle avoidance visualized



#### Follow these steps:

1. Select the menu option **Control AI > AI Debug Panel**.
2. In the **Obstacle Avoidance** section of the panel, set the following settings:
  - Show (below)
  - Vertical Offset (below)
  - Scale (below)
  - Show Braking Penalty (below)
  - Show Velocity Penalty (on the next page)
  - Highlight Minimal Penalty (on the next page)

#### 12.7.1 Show

Shows / hides the obstacle avoidance visualization.

#### 12.7.2 Vertical Offset

Changes the vertical offset of the displayed penalty bars.

#### 12.7.3 Scale

Changes the scale of the displayed penalty bars.

#### 12.7.4 Show Braking Penalty

Shows / hides an orange bar for each sampled velocity. The height of the bar is proportional to the braking penalty.

## 12.7.5 Show Velocity Penalty

Shows / hides a cyan bar for each sampled velocity. The height of the bar is proportional to the velocity penalty.

## 12.7.6 Highlight Minimal Penalty

Shows / hides a green sphere, highlighting the sampled velocity with the lowest overall penalty.

## 12.8 Actor Velocity

You can visualize actor (Control AI entity) velocity. The visualization can differ, depending on whether an actor is in a vehicle or not.

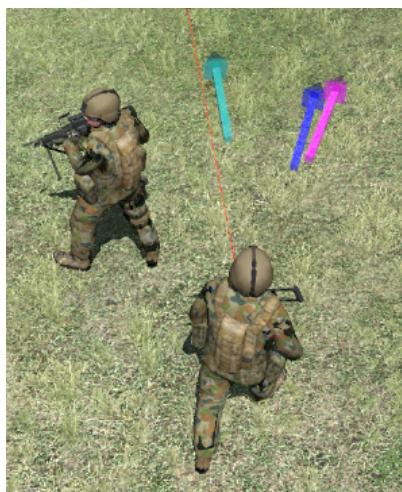
### **i** NOTE

The settings in this section allow you to display the visualizations only for selected Control AI entities. If no Control AI entities are selected, the visualizations are displayed for all Control AI entities.

The different visualizations are:

- **Dismounted** - When dismounted from a vehicle.
  - **Planned Velocity (cyan arrow)** - Velocity for following the planned path on the navigation mesh.
  - **Wanted Velocity (blue arrow)** - Velocity for following the planned path, while avoiding collision.
  - **Current Velocity (magenta arrow)** - Immediate velocity of the actor.

**Image-89: Dismounted actor velocity**



- **Wheeled Vehicle** - When in a wheeled vehicle.

- **Front Indicator** - Indicates steering:

- Yellow indicates the level of steering in a given direction.

- **Left Indicator** - Indicates speed:

When acceleration is wanted:

- Green indicates the current speed.
    - Green + yellow indicate the wanted speed.

When deceleration is wanted:

- Green + red indicate the current speed.
    - Green indicates the desired speed.

- **Middle Indicator** - Indicates throttle:

- Green indicates the level of forward throttling.
    - Cyan indicates the level of backward throttling.

- **Right Indicator** - Indicates braking:

- Red indicates the level of braking.

**Image-90: Wheeled vehicle velocity**



- **Tracked Vehicle** - When in a tracked vehicle.

- **Left Indicator** - Indicates speed:

When acceleration is wanted:

- Green indicates the current speed.
    - Green + yellow indicate the wanted speed.

When deceleration is wanted:

- Green + red indicate the current speed.
    - Green indicates the desired speed.

- **Two Middle Indicators** - Indicate left and right throttle:

- Green indicates the level of forward throttling.
    - Cyan indicates the level of backward throttling.

- **Right Indicator** - Indicates braking:

- Red indicates the level of braking.

**Image-91: Tracked vehicle velocity**



- **Rotary-Wing Vehicle** - When using a rotary-wing vehicle.
  - **Front Indicator** - Indicates rotation:
    - Yellow indicates the level of rotation in a given direction.
  - **Left Indicator** - Indicates speed (without direction):
    - When acceleration is wanted:
      - Green indicates the current speed.
      - Green + yellow indicate the wanted speed.
      - Gray indicates the maximum speed.
    - When deceleration is wanted:
      - Green + red indicate the current speed.
      - Green indicates the wanted speed.
      - Gray indicates the maximum speed.
  - **Middle Indicator** - Indicates velocity (speed with direction):
    - Cyan indicates the current velocity.
    - Blue indicates the wanted velocity.

**Image-92: Rotary-wing vehicle velocity**



#### Follow these steps:

1. Select the menu option **Control AI > AI Debug Panel**.
2. In the **Actor Velocity** section of the panel, set the following settings:
  - Show (on the next page)
  - Vertical Offset (on the next page)
  - Scale (on the next page)

## 12.8.1 Show

Shows / hides the actor velocity visualization.

## 12.8.2 Vertical Offset

Changes the vertical offset of displayed arrows / indicators.

## 12.8.3 Scale

Changes the scale of the displayed arrows.

## 12.9 Polygon Settings

This section contains the Polygon Settings.

### 12.9.1 Polygon Line Width

Control the polygon line width.

### 12.9.2 Polygon Fill Opacity

Controls how opaque the polygons are.

# 13. Example Content

VBS Control AI comes with example content to demonstrate its functionality.

The example scenarios are available at:

`\VBS_Installation\optional\Behavior_Pack\Battlespaces\`

To run the example scenarios, follow these steps:

1. Copy the example scenarios to:

`\Documents\VBS4\Battlespaces\`

2. Start VBS4.
3. Select from the following example scenarios:

- **Civilian Traffic Example (CivilianTrafficExample)** - This example is based on the Sahrani terrain, and demonstrates the use of the [Control AI - Civilian Editor Object \(on page 20\)](#), [Control AI - Activity Editor Object \(on page 20\)](#), and [Control AI - Area Editor Object \(on page 21\)](#) to populate a town with pedestrian and vehicle traffic, including functioning traffic lights.
- **AI on Rails Example (AI\_on\_rails\_example)** - This example is based on the Sahrani terrain, and demonstrates the use of [AI on Rails \(on page 36\)](#) orders ([Individual - Move Order \(on page 50\)](#)) and [Individual - Fire At Order \(on page 47\)](#) in a vehicle ambush, using both radio and presence-activated triggers.
- **Convoy Example (Convoy\_example)** - This example is based on the Sahrani terrain, and demonstrates the use of the [Convoy AI \(on page 93\)](#) with [Branching Waypoints \(on page 25\)](#) - the user can choose between two convoy paths: to go through the city (where the enemy is present) or around the city (safe route), using radio triggers. The user can also order to [Pause / Resume Convoy \(on page 96\)](#) movement.

## 13.1 Additional Example Content

In addition to the example scenarios, the following example content is available in:

`\VBS_Installation\optional\Behavior_Pack\Battlespaces\`

- **Airport (CivilianAI\_example\_mission\_airport)** - An example scenario with a sequence of activities, simulating passengers going through several stages of airport security.
- **Concert (CivilianAI\_example\_mission\_concert)** - An example scenario that demonstrates the use of Generic and Interrupt activities (see [Define Pedestrian Flows \(on page 141\)](#)), to simulate complex crowd behavior. The scenario also uses triggers (see Triggers in the VBS4 Editor Manual) to control the flow of civilian AI actors.

# 14. Known Issues

VBS Control AI has the following issues / limitations, which will be addressed in future releases of VBS4.

## NOTE

Multiplayer issues are marked with (MP).

The issue categories are:

- [Military \(below\)](#)
- [Convoy \(on the next page\)](#)
- [Bridge Laying Convoy \(on page 233\)](#)
- [Aircraft \(on page 234\)](#)
- [Civilian \(on page 234\)](#)
- [Civilian Traffic \(on page 235\)](#)
- [Animal \(on page 236\)](#)
- [General \(on page 237\)](#)

## 14.1 Military

The issues in the Military category are:

Issue	Workaround
Changing an <a href="#">AI on Rails (on page 36)</a> waypoint type (for example, from <a href="#">Individual - Move Order (on page 50)</a> to <a href="#">Individual - Fire At Order (on page 47)</a> ) in the Editor (Prepare Mode) leaves the old Individual Order name above the entity proxy in 3D view.	Save the mission and load it again.
Opening the Editor (Execute Mode) right after mission start results in displaying the wrong waypoints for AI on Rails in the 3D Editor (Execute Mode).	Close the Editor (Execute Mode) and open it again.
Infantry AI units not equipped with launchers (or without launcher ammunition) engage vehicles with their primary weapons.	None, will be improved in future releases.
Infantry AI units only use primary weapons and launchers.	None, will be improved in future releases.
Control AI is sometimes not able to correctly identify urban areas that are composed of generic buildings. As a result, squads cannot correctly switch formation in such areas.	None, will be improved in future releases.

## 14.2 Convoy

The issues in the Convoy category are:

Issue	Workaround
Driving of vehicles with powered trailers (for example, Viking, Bv206) is not supported (regular trailers work as expected).	Use a different vehicle type. Support may be added in future releases.
Non-PhysX vehicles are not supported (such as Motorcycle TT - 650 or Mountain Bike).	Use only PhysX vehicle types.
Spacing is sometimes not maintained correctly, when using tracked vehicles with different speed and acceleration in a Control AI convoy.	Set a smaller movement speed in the Control AI Convoy settings.
A Control AI convoy that is set to drive on roads (set to <b>Use only roads</b> or <b>Use only roads, respect lanes</b> ) cannot drive around obstacles on the road.	Delete or move the obstacle, or switch to <b>Prefer roads</b> .
When using off-road driving in convoys, some wheeled vehicles may not drive carefully enough on steep inclines, and as a result, can crash.	Pick a different convoy route. May be improved in future releases.
Control AI vehicles driving on roads, with <b>Use only roads, respect lanes</b> set, may be unable to get past some boom barriers (bar gates) or road signs.	Choose one of the following options: <ul style="list-style-type: none"> <li>• Delete or move the boom barrier / road sign away from the road.</li> <li>• Switch to a different road usage mode (for example, <b>Use only roads</b> or <b>Prefer roads</b>).</li> </ul>
When a vehicle is placed off-road with a Convoy waypoint (whose road usage is set to <b>Use only roads</b> ) also placed off-road, the vehicle chooses a direct off-road path (it does not move to the closest road first, and uses it to get to the waypoint).	Place the vehicle and the waypoint on the road. In case the off-road placement of the waypoint and / or the vehicle needs to be kept, use multiple waypoints, with the off-road ones set to the <b>Ignore roads</b> road usage.
The convoy behavior counts on waypoints being connected in a way that the convoy does not need to turn around.  Although the convoy can turn around in ideal conditions, it is not recommended to create a situation when this is required, since the convoy can get stuck after vehicle reordering.	Make sure that the next waypoint is always ahead of the lead vehicle, to prevent the convoy from reordering.

Issue	Workaround
The convoy behavior does not have any way of ordering the vehicles, other than assessing the distances of individual vehicles from the waypoint, so that at each turn, the vehicle order may change, if the distance to the first vehicle is not the shortest one.	To preserve the convoy vehicle order, use Follow Callsign.
Wheeled Control AI vehicles require more space than tracked vehicles while driving around obstacles due to their turning radius.	None, vehicles choose a path wide enough for them to fit in if it is possible.

## 14.3 Bridge Laying Convoy

The issues in the Bridge Laying Convoy category are:

Issue	Workaround
Movement of Control AI entities on a Dry Support Bridge (DSB) is not supported.  For more information on Control AI and DSB laying, see <a href="#">Bridge Laying Convoy AI (on page 112)</a> .	Support may be added in future releases.
Driving in reverse for vehicles with hitched trailers is not supported.	Ensure that those vehicles in the bridge-laying convoy with hitched trailers do not have to change direction quickly (which can happen, when there are two <a href="#">Convoy Order (on page 105)</a> waypoints with a sharp turn and / or some obstacles, when driving off the road).  Support may be added in the future releases.

## 14.4 Aircraft

The issues in the rotary-wing and fixed-wing aircraft category are:

Issue	Workaround
<p>The <a href="#">Fly Order (on page 124)</a> has the following limitations:</p> <ul style="list-style-type: none"><li>• The Fly Order cannot be used with omnicopters.</li><li>• The Fly Order can only be used with single aircraft, not aircraft groups.</li><li>• The <b>Weapon Control Status</b> is ignored for fixed-wing aircraft.</li></ul>	None.
<p>The <a href="#">Land Order (on page 127)</a> has the following limitations:</p> <ul style="list-style-type: none"><li>• The Land Order can only be used with rotary-wing aircraft.</li><li>• The Land Order can only be used with single aircraft, not aircraft groups.</li></ul>	None.
<p>The <a href="#">Loiter Order (on page 129)</a> has the following limitations:</p> <ul style="list-style-type: none"><li>• The Loiter Order can only be used with fixed-wing aircraft.</li><li>• The Loiter Order can only be used with single aircraft, not aircraft groups.</li><li>• The <b>Weapon Control Status</b> is ignored for fixed-wing aircraft.</li></ul>	None.

## 14.5 Civilian

The issues in the Civilian category are:

Issue	Workaround
If the Control AI - Civilian Editor Objects are deleted and re-created at runtime too many times, VBS4 may deplete the number of available groups, resulting in no more AI being spawned and a script error thrown.	Avoid frequent deletion and re-creation of Control AI - Civilian Editor Objects.
Civilians form queues across roads, when queue capacity is high near roads.	None, will be improved in future releases.
Restricted areas (see <a href="#">Area (on page 198)</a> activities and the <a href="#">Area Editor Object (on page 199)</a> ) can be moved, but do not get updated while moving.	None, will be improved in future releases.

## Issue

Restricted areas placed in Prepare mode might get ignored by the civilian AI, running on simulation clients.

## Workaround

Move or update the restricted area in any way, to propagate it to the simulation client.



### TIP

To do it automatically, link a trigger to the Restrict Access activity, which has a timeout set to 10 seconds, and a condition set to `true`.

Children do not execute any activities, other than **Start Panic** (see [Interrupt \(on page 196\)](#) activities).

None, will be improved in future releases.

Children do not despawn (see [Spawn / Despawn \(on page 195\)](#) activities).

None, will be improved in future releases.

## 14.6 Civilian Traffic

The issues in the Civilian Traffic category are:

## Issue

Bridge sections removed in an active [Population Editor Object \(on page 189\)](#) area are not updated in the road network.

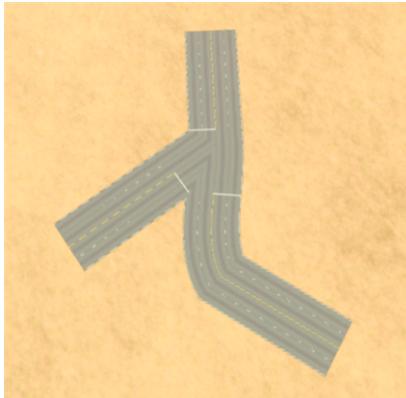
## Workaround

None, will be improved in future releases.

Improperly configured terrains may cause incorrect traffic behavior, often caused by:

- Improperly configured road network.
- Improperly configured traffic lights.
- Complex crossroads, containing multiple junctions.
- Combinations of turns that are too sharp.
- Lanes that are too narrow or too wide.

Some of the wrong configuration can be exposed by using debugging tools for [road networks](#), [traffic lights](#), and [vehicle driving](#). Alternatively, try to use a different location on the terrain.

Issue	Workaround
High density of pedestrians and civilian traffic can result in blocked junctions.	<p>Several solutions are available:</p> <ul style="list-style-type: none"> <li>Decrease the number of vehicles and / or pedestrians.</li> <li>Modify the destinations of the AI by moving / adding / removing activities in the area.</li> <li>Provide more space for the AI by removing obstacles around the narrow area.</li> <li>Use <a href="#">traffic lights</a> to control the flow of the traffic and pedestrians.</li> </ul>
In MP with low FPS, civilian vehicles drive badly, when a player in a vehicle is present nearby.	There is no workaround. However, the occurrence intensity of this problem depends on the performance of the network scenario.
When using civilian traffic on some terrains with simulation clients, vehicles may temporarily swerve, when the simulation is transferred to a simulation client.	Vehicles continue driving as normal, when they are moved back on the road in the Editor (Execute Mode). Will be improved in future releases.
Vehicles can get stuck at Y-junctions.	<p>Offset the Y-junction arms, so that one arm is placed before / after the other arm, using VBS Geo.</p> 

## 14.7 Animal

The issues in the Animal category are:

Issue	Workaround
Animals are only partially supported by Control AI, where they can only move forward or be idle (using the <b>Animal Herd Movement</b> behavior - see <a href="#">Animal AI (on page 202)</a> ). They cannot perform actions like other Control AI units (they cannot move sideways, aim, shoot, roll, change stance, and so on).  Also, only some animal types are supported by Control AI (for example, rabbits are not supported).	None.

## 14.8 General

The issues in the General category are:

Issue	Workaround
An entity may be unable to find a path around obstacles larger than 50m. For more information, see <a href="#">Navigation Mesh Limitations (on page 206)</a> .	Pick an intermediate waypoint, to plan a path around the obstacle. May be improved in future releases.
Loading the coarse navigation mesh for a large terrain may take a few seconds, which can make VBS4 unresponsive. This may happen on mission start, if any Control AI Editor Object (see <a href="#">Control AI Editor Objects (on page 17)</a> ) is in the mission, or during runtime, when the first Control AI Editor Object is placed in the Editor (Execute Mode).	Place the Control AI Editor Object ( <a href="#">(F3)</a> <a href="#">Waypoints Editor Object (on page 18)</a> , <a href="#">Control AI - Civilian Editor Object (on page 20)</a> ) when creating a mission, so that the coarse navigation mesh gets loaded during mission load, and not in runtime.
Control AI units which are killed do not continue performing VBS Control AI behaviors after revival.	None. May be improved in future releases.
Path-finding does not take into account snow compression and road plowing (see the <b>Snow</b> section in Weather Settings in the VBS4 Editor Manual). <ul style="list-style-type: none"><li>• Path-finding does not prefer compressed snow over uncompressed snow.</li><li>• Path-finding does not prefer plowed roads over unplowed roads.</li></ul>	None. May be improved in future releases.
Path-finding does not take surface types, such as ice or mud, into account (see the <b>Surface</b> section in Weather Settings in the VBS4 Editor Manual). Therefore, vehicles may not be able to follow the found path, if the surface is frozen or slippery, and the vehicle tires are unfit for it.	None. May be improved in future releases.
Multi-lane traffic is supported only partially. Vehicles can sometimes get stuck.	Simplify the road network using VBS Geo / TerraTools. Better support may be added in future releases.