

AvantGuard: Guides for Users

Guide for:

Players

Overview

In Avant Guard your goal is to keep a convoy safe from attacks by surveying the area, detecting threats, assessing the safest routes, and avoiding the dangerous ones. Based on operator settings, some of these steps may take a lot of user input, while others may be fairly automated.



Here you will see a layout of the interface. In the upper right corner, there is a 2D map. Below that are buttons used to execute different actions which we will go over later.

At the bottom of the screen is a timeline. The white bar is the convoy. The three other bars correspond to the UAV of matching color.

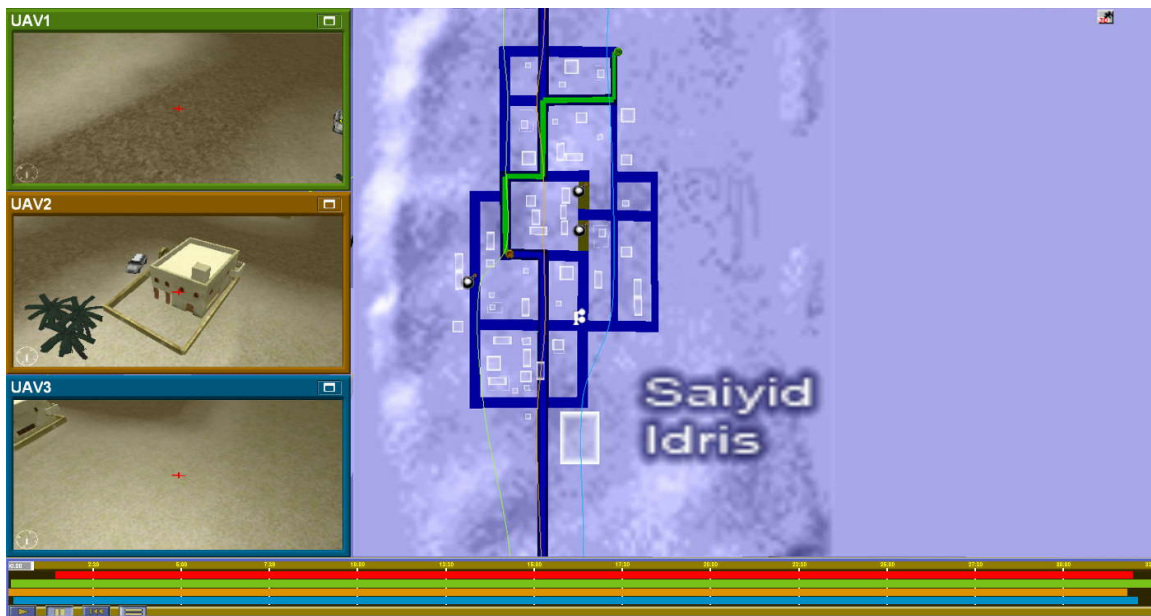
On the left side are video streams of each UAV. In the rest of the screen is a 3D view of the world.

Camera Controls

Use the middle mouse button or WASD to drag the scene in both 2D and 3D.

Scrolling with the middle mouse button zooms in and out. Zooming out all the way goes to the map overhead.

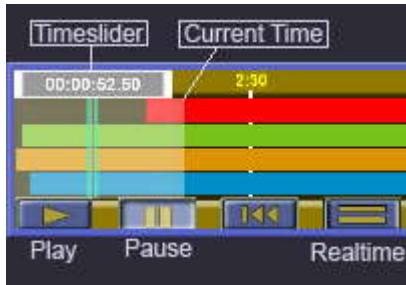
To rotate the scene, hold Shift while dragging with the middle mouse button. This same action can be done on the keyboard by using Shift + A and Shift + D. When you let go of the buttons, the scene will rotate back so that north is up.



Time Controls

The Timeslider shows the videostream, convoy location, and UAV locations at that time. It can be moved to both the past and the future but cannot show video stream when the future.

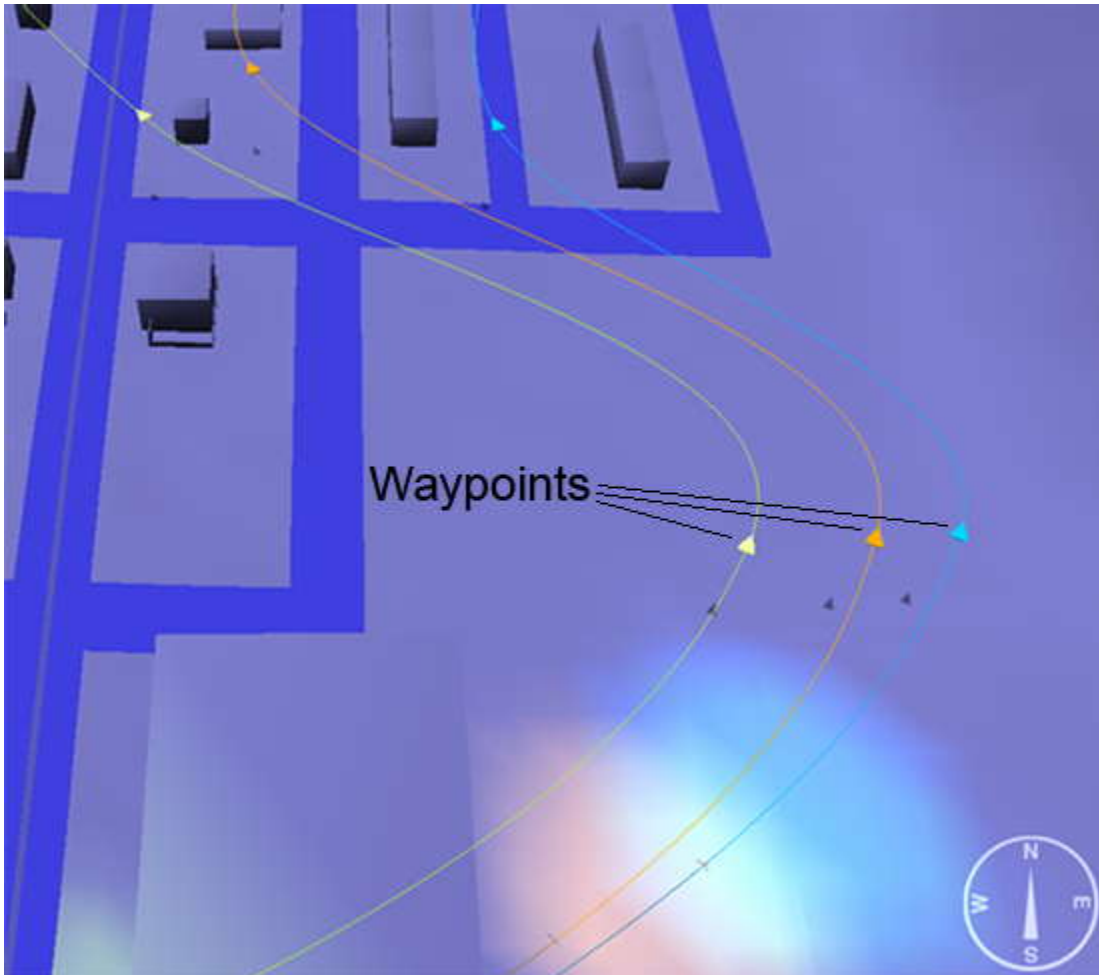
To sync it back to the current time, use the Realtime button in the bottom right.



Threat Surveillance

Drag the waypoints in either the map overlay or the 3D world to designate where the UAVs will survey.

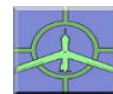
Each UAV has a light on the ground that shows what it will be viewing at that time.



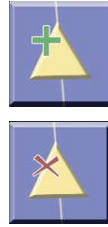
In higher levels of autonomy, use the Points of Interests tool to highlight streets which you want the UAVs to survey.



If at any time you are unsure of where a UAV is, using the Find UAV button will jump to the location and select the flightpath of each plane one by one.



When dealing with waypoints, you can add them to the mission path by using the Insert Waypoint button. Click on a waypoint to select it, click the button, and a new waypoint will be added immediately following the selected one. You can also delete a waypoint by selecting a waypoint and clicking the Delete Waypoint button.



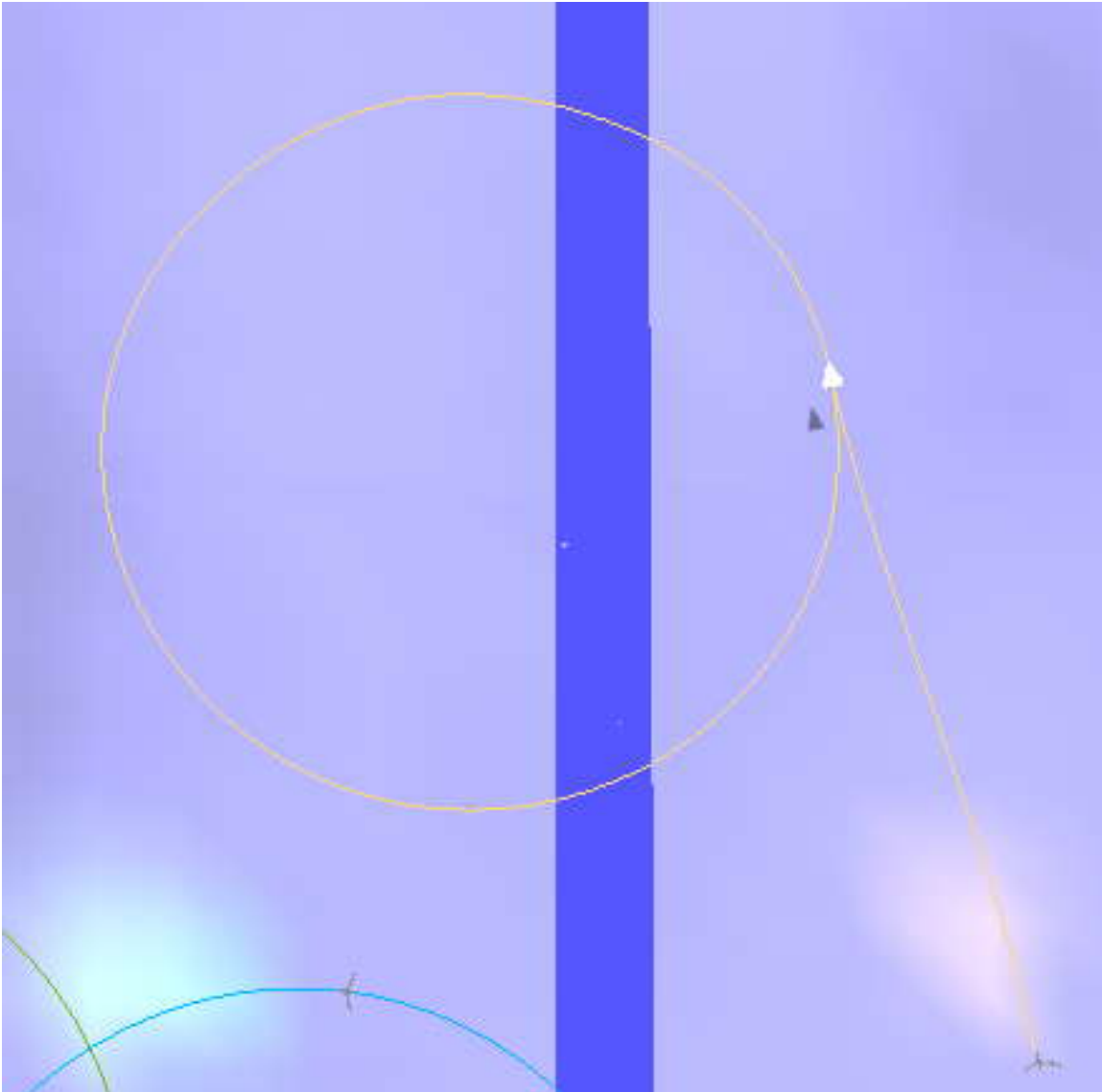
In low levels of autonomy, instead of waypoints you will be in charge of the gimbal camera. Open up any one of the three camera windows by clicking on the top border. This will enlarge the window as well as bring the main scene camera to the UAV's so that they overlap.



When this window is enlarged, you can look around by using the W, A, S, and D keys. Using these keys to look around, you can choose a point in the world with the crosshairs that are in the center of the window around which the camera will rotate by pressing the spacebar.

When a UAV reaches the end of its mission, it will loop until the convoy reaches its destination. To extend the mission at this point, first select the mission by using the Find UAV button. Then, click on the Insert Waypoint, which will add new waypoints to the mission. Or, in higher levels of autonomy, use the points

of Interests tool to direct the UAVs to new destinations.



Threat Detection

Click in the camera windows to mark threats. Hold down the left mouse button when placing a threat to specify the type of threat. This will affect the travel speed calculations of the threat in assessment.

Clicking on the top border of the window will enlarge it. When the window is enlarged, the main scene camera will be synced to the UAV camera. Also, when it is enlarged, you can move the view by using WASD.



Armed men should always be avoided.



Be cautious of trucks carrying men or explosives.

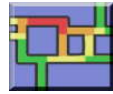




Explosives can be hidden in a variety of ways, including in trash and even in roadkill.

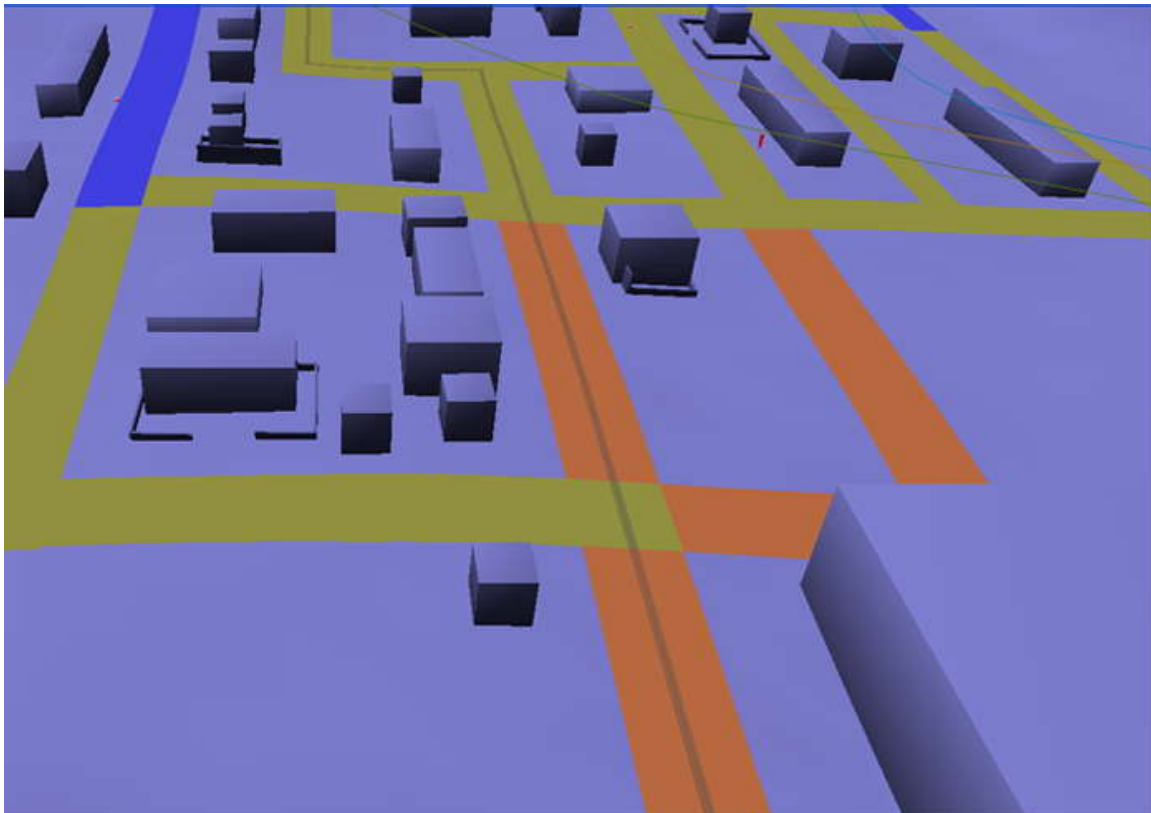
Threat Assessment

The street painter tool can be used to manually assign threat levels to the streets if this is not already automated.



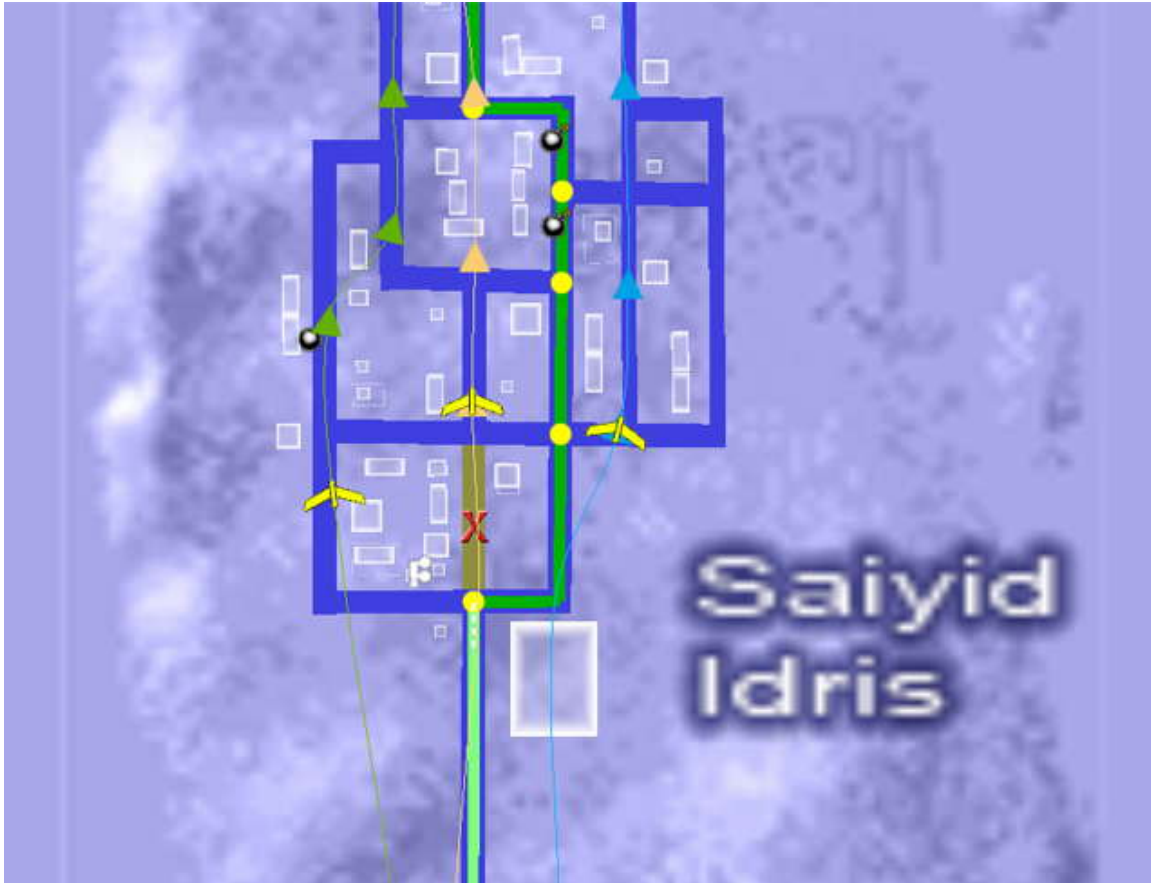
Threat levels on streets show the probability of a threat on that street

- Green – Safe
- Blue – Unknown [Guarded]
- Yellow - Elevated threat
- Orange – High Threat
- Red – Severe Threat

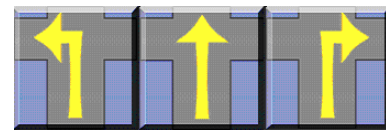


Threat Avoidance

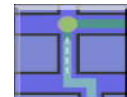
To redirect the convoy, drag the yellow node at the intersection that you want to change to the new direction.



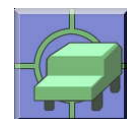
In very low levels of autonomy, you will be required to steer the convoy. These buttons, turn left, go straight, turn right, tell the convoy what to do at the next intersection.



In higher levels of autonomy, using the Recompute Convoy Route button will direct the convoy to automatically plan a new route.



If at any time you are unsure of where the convoy is, using the Find Convoy button will jump to its location.



Guide for:

Researchers

Autonomy

Autonomy is the heart of AvantGuard.

This User's Guide will instruct you how set up a desired Autonomy Profile. It does not teach the theory and practice of Autonomy in AvantGuard. For that, please refer to the Autonomy section in the **AvantGuard Technical Report**.

You are urged to review that material before proceeding.

Step by step

The autonomy system allows for individual autonomy settings for four tasks: Survey Threats, Detect Threats, Assess Threats, and Avoid Threats.

- 1) The autonomy dialog can be accessed through the Context Menu by selecting "Autonomy settings..."
- 2) On the left is a slider to control the autonomy for the selected page, in this case, surveillance. The slider goes from Level 1(manual operation) to Level 10 (fully autonomous operation).
 - a) Not every level is implemented in the current code base.
 - i) Refer to the Technical Report.
 - ii) Level 1 is not available for Surveillance
 - iii) In all stages, Levels 9 and 10 have the same behavior.
- 3) On the right are descriptions of the settings.
The enabled ones show behavior set in place with the current autonomy level.
- 4) Beyond this, there is the **Options** menu, which allows the designer to change details in the AI implementation. While this does affect how the autonomy behaves, it doesn't change the numeric autonomy rating, set by the slider on the left.
- 5) Immediately below this are means of **navigation**.
 - a) The drop down menu on the left allows you to jump to the different pages.
 - b) The previous and next buttons on the right allow you to traverse them in order.
- 6) At the bottom are six buttons for file control.

- a) **Load** – loads a previously saved autonomy file.
 - b) **Save** – saves the current autonomy settings in a file. (This does not automatically associate the autonomy to this scenario.).
 - c) **Defaults** – loads the autonomy settings associated with this scenario, as last saved.
 - d) **Apply** – applies the settings without closing the window.
 - e) **OK** – applies the settings and closes the window.
 - f) **Cancel** – closes the window without applying changes.
- Note that Load, Default, Apply and OK immediately change the autonomy settings, but they will not change the autonomy associated with this scenario. Nor will it be changed by saving the autonomy file. In subsequent sessions, the scenario will continue to load its original autonomy file.
- 7) To make a **permanent change** to the autonomy associated with this scenario, save both the autonomy (.aut) and the scenario (.scn) files.

Session

Before the session, the scenario is prepared. Many experiments can be

Step by step

1) Preparation

- a) Autonomy must be set for the scenario (.SCN) before beginning a session.
- b) The Edit Option for the scenario must be set to “0” for the player.
- c) You will want to be sure the SCN files open with the correct program.
 - (1) Right click on any SCN file and go to “Properties.”
 - (2) Under the “General” tab, it should say “Opens with: Works”.
 - (3) If it does not, click on Change, and select the Works program.
 - (4) From now on, SCN files will open in the Works engine.
- d) As a convenience for you and the Subject, a shortcut to create

2) Performance

- a) PREPARE THE PLAYER
 - i) Orientation of the player will require a verbal description. A standardized script will help make prevent unwanted variation.
 - ii) Training Scenarios – To help prevent a learning curve from affecting the results, make sure the player has successfully completed all the training scenarios. Start the player with Training00.scn. Each successfully completed scenario will automatically move the player to the following one.
- b) When the player opens the scenario, he will start with the Main Menu immediately available.
- c) Second in the menu is the “Start” button, which will allow the user to input his name for the new prior to starting the session.
- d) Once OK is clicked, the game will begin.

3) End

- a) The session will end either when the player reaches the end of that section of streets, or when the player runs into too many threats so that his health depletes to zero. At this point the program will automatically save the results to the Results.csv file.

- b) The same scenario will automatically reload, ready for a new player. Simply click on “Start” and type in the name of the new player. If you want to test a different scenario you will have to close the program in order to load the new scenario.

Data

At the end of the mission, all session data is recorded in a **results file**.

This file accumulates the results of all sessions. Each session produces a new row of data in

Format

This file uses the common '**comma separated value**' format.

It can be processed directly by a variety of data management software

- Spreadsheet (Excel)
- Database (SQL)
- Statistical Analysis (SPSS)

Onscreen Display of Results

Optionally the data results can be posted to the console at the end of the session.

Step by step

- 1) This option is saved per scenario file. To change the option, open the SCN in an XML editor and look for ShowStats under the <Options /> tag. If ShowStats does not appear, it will be set to "1" by default.
- 2) Change this by including 'ShowStats="0"' inside <Options />
`<Options ShowStats="0"/>`
 - a) ShowStats="0" will not show the results at the end of a session (they will still be saved to the results file).
 - b) ShowStats="1" will show the results at the end of a session.

Starting a New Results File

All results are saved to the current results file. You can set the name of this file.

Step by step

- 1) To rename the results file, you will need to open the console inside the program by hitting the ` key.
- 2) In the console type 'setresultsfilename <name here>'
`setresultsfilename NewResults`
- 3) This will not erase any results files; it will only change which one is being recorded to, and if the file does not exist, creates it.

Resetting the Serializer

Every session has a serial number. These numbers are sequential and unique even across multiple results files. However you can reset this number to anything value.

This can be used to assign each machine, each reasearcher or each experiment its own range of unique serial numbers (the maximum is 2 billion)

Alternatively it can be used to ensure that every experiment has a private numbering system that always starts with number 1.

Step by step

- 1) To reset the session number, you will need to open the console inside the program by hitting the ` key.
- 2) In the console type 'setsessionnumber <session number here>'.
`setsessionnumber 1`
- 3) This will not delete or overwrite any previous sessions.
- 4) The next session n this machine will be 1
- 5) The next will be 2, etc

Results

This is an exhaustive list of the data recorded in the Results file''.
(default name: results.csv)

Session Identification

Subject

Subject's name as he typed it in.

Session

Session serial number (on this computer).

Session Timestamp

Time when the session is completed

Scenario

Name of the scenario (SCN) file.

Scenario TimeStamp

Modified date and time of the SCN file.

Autonomy

Name of the Autonomy (AUT) file.

Autonomy TimeStamp

Modified date and time of the AUT file.

Timeline Usage

View Past

View RealTime

View Future

Reports the use of the Timeline.

How many seconds were spent by the subject reviewing the past, examining the future or simply operating in realtime?

Direct Interaction With Autonomy

Survey Approved

Survey Rejected

Survey TimedOut

Survey NotReady

Survey Reviewed

They measure how many autogenerated flightpaths were approved by the subject, rejected by the subject, allowed to time out by a passive subject or were never seen by a subject who was not ready – because he had not yet cleared a previous approval dialog.

It also measures how many times the subject used the view button to alternate the display of the two options he is asked to select among.

These are only valid in levels 5 and 6.

Detect Approved

Detect Rejected

Detect TimedOut

Detect NotReady

Detect Reviewed

These are all equivalent to the corresponding ‘Survey XXXXXXXX’ fields except that they measure the response to tentative threat detections.

In Detect, these metrics are valid in level 4 as well as levels 5 & 6,

There is no NotReady feature in Detect.

Assess Approved

Assess Rejected

Assess TimedOut

Assess NotReady

Assess Reviewed

These are all equivalent to the corresponding ‘Survey XXXXXXXX’ fields except that they measure the response to candidate threat maps.

Avoid Approved

Avoid Rejected

Avoid TimedOut

Avoid NotReady

Avoid Reviewed

These are all equivalent to the corresponding 'Survey XXXXXXXX' fields except that they measure the response to convoy reroutes.

Performance in Threat Detection

Threats Apparent

This is a count of items in the world with a non-zero **threat appearance**.

Threats Real

This counts entities with the **real threat** flag set.

Threats Visible

This is a count of how many apparent threats appeared within the view of the UAVs. It is meaningful only in detection autonomy levels 3,4,5 & 6.

Threats Marked

Unique threats detected and marked. This is the same as Threat Marks minus those identified as secondary sightings of a previously marked threat.

Threat Marks

The number of total sightings registered by the human or automatic threat detector. It includes duplicate sightings.

Performance under Low Autonomy

UAV Reroutes

How many times did the subject manually alter the flightpaths?

AOI Changes

How many times did the subject set new Area of Interest targets for the UAVs?

Convoy Route Changes

How many times did the subject manually change the Convoy route?

Scores

Objectives

How many Objectives exist in this Scenario?

Objectives Achieved

How many of these Objectives are successfully achieved by the Player?

Session Length

How many seconds did the session last?

Health

Measures overall success. 100 health means no threat contact.

In Session Reports

System Messages

This string value concatenates all messages displayed by the system during this session.

Guide for:

Designers

Change Your Point of View

There are two methods of moving your point of view in the world:

Iso Cam is intended especially for the Player. It is designed to be foolproof.

Free Cam serves the developer's needs. It maximizes your freedom of motion.

Step by step

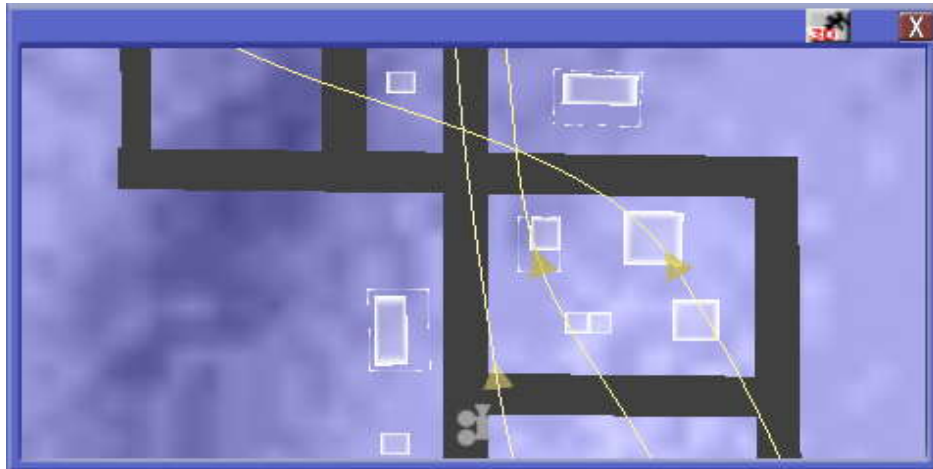
- 1) To switch between the two modes, right click to open the Main Popup. Near the bottom you will see "Camera Mode." Mouse over this option which will open a sub menu to the right. From here, choose "Camera dynamics..." There is a checkbox in the upper right to toggle between the two modes.
- 2) Iso Cam
 - a) In "Isometric Mode" the view will always face north.
 - b) Drag the scene by holding down the middle mouse button and dragging.
 - c) Zoom in and out by scrolling the middle mouse button forward and backward.
 - d) Holding shift while middle mouse dragging will let you temporarily rotate the camera. It will return to facing north when shift is released.
 - e) Currently, you cannot use this mode to add objects at a specific location.
- 3) Free Cam
 - a) In "Free Cam Mode", your movements will be centered around your pick point. Middle clicking on the scene picks a point on the terrain or on an object. This mode is currently required when adding objects.
 - b) Middle mouse drag or use the arrow keys to rotate the camera around this point.
 - c) Zoom in and out on this point by scrolling the middle mouse button forward and backward.
 - d) Freelook by holding "Tab" and moving the mouse. Holding "Tab" and scrolling the middle mouse button will let you fly through the world at very high speeds.

Map Overlay

The map overlay is a 2D representation of the 3D world.

In Isometric Mode, as you move the camera in either the map or the 3D world, the other will move in relation to it.

Any objects that can be dragged in 3D can also be dragged in the map overlay.



Step by step

- 1) The map overlay uses the same controls as Iso Cam
 - a) **Pan** by Middle Click and Drag
 - b) **Zoom** by rolling mouse Fly Wheel.
- 2) The map can be repositioned on the screen
 - a) **Resize** by drag on the border.
 - b) **Move** by drag on the top bar.

Editing Tools

To edit most anything in the scenario, editing must be turned on. Of course, this must be turned off when giving the scenario to the player.

There are two ways of changing this.

Step by step

- 1) With the program already running
 - a) To temporarily change the edit mode of a scenario, hit the ` key while in game, which opens up the console.
 - b) In the console type "set edit 1" to turn on editing
 - c) Type "set edit 0" to turn off editing.
- 2) Before opening the scenario
 - a) To change the edit mode more permanently, open the scenario XML and look for Options.
 - b) Here you should find one of the strings 'edit = "1"' or 'edit = "0"'. Again, 1 turns edit mode on, and 0 turns it off.

A Scenario is made up of many pieces. Any of these can be individually modified to create the scenario you want, or you can change them all to create a completely new scenario. If you would like to start with a completely blank canvas you can open the Empty.scn file.

Context Menu

The most basic way to get to any edit option is through the “Context Menu.” Through this window you can create and access the majority of tools you will be using. These tools will be discussed in detail later. But for now a general overview of this menu will be given.

Step by step

- 1) Right click in the world to open the Context Menu
- 2) There are several functions you can access here. Extended options are available in the sub menus. Simply left click on the item in the menu you would like to use.
- 3) Open .MOB, .SCN, .MIS, .SEM
- 4) Save .MOB, and .SCN files.
- 5) Control display and animation settings.
- 6) Adjust user controls.
- 7) Mission options and scripting.
- 8) Lighting options.
- 9) Creation tools.
- 10) Hierarchy control.
- 11) Camera options.
- 12) Left click in the main scene to cancel.

Animation Parameters

Found in the context menu, the animation parameter menu makes editing easier by giving the designer control over game time and frame rate.

Here, you can toggle display of the frame rate with “Show frame rate.” High frame rates mean the game will run smoothly for the player. Slow frame rates mean you have made the world too complex.

If you need the program to remain open, while you work in another window, you can turn on “Slow frame rate,” so that the program does not hog resources the other might need.

Also in this menu is the “Pause All” selection, which, like it says, pauses everything. This is ideal for getting a closer look at something that would be moving too quickly in the game’s real time.

In addition to these are the “Simulation Speed controls.” Under this menu choose how fast you want the world to run, 1/8X being the slowest, 8X being the fastest, and 1x being real time. Slower speeds, similar to the pause function, allow you to study things that might normally be moving too quickly. Higher speeds allow you to test events that occur late in the scenario. Instead of waiting for an event to happen that needs to be tested, you can speed things up. You must be in edit mode to access these controls.

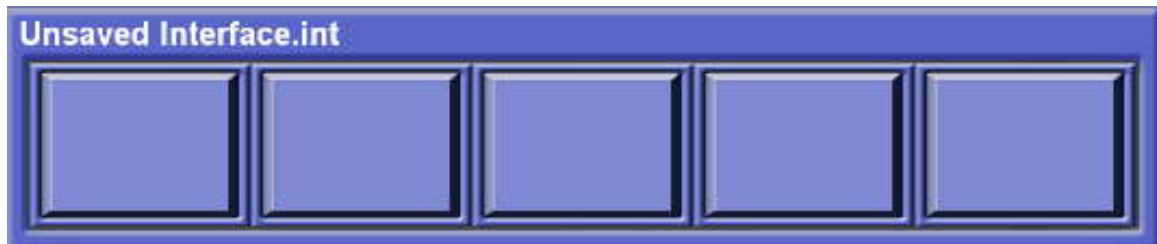
Toolbars

Toolbars are completely customizable to allow the developer to create a toolbar specifically for those tools he will be using the most. Through this toolbar you can access many of the common features that normally require going through the Context Menu, such as adding mesh objects, buildings, animated meshes, models, lights, streets, mission paths, and more.

Step by step

Creating a Toolbar

- 1) If you do not already have a toolbar on the screen, or would like another one, right click to open the Context Menu, and then go down to "Create object." In this sub menu, select "Toolbars."



- 2) This will add a blank toolbar to your screen. By dragging on the border, you can move it, resize it, or reshape it to be vertical.
- 3) Right clicking on the toolbar will open a menu where you can save the current toolbar, load one previously saved, add a button, or adjust settings of the toolbar.
- 4) If you right click on a button you can do these things, as well as edit that button with "Assign Action." There are two types of actions you can assign to a button: execute and create. This gives you access to a lot of the same features in the Context Menu window, but makes them more readily available. Once a button is assigned, it can always be reassigned to perform a different task.
- 5) You can assign a button to create a mesh, animated mesh, building, or model. Because these specific buttons have prefabs from which you can choose, the following options are available.
 - a) If you want to create the same mesh every time, simply choose the mesh from the menu and hit "Create".
 - b) However, if you would rather it ask you each time which mesh you would like to create, hit "Cancel" the first time. Now, every time you

use that toolbar button, it will ask you which mesh you would like to place.



- 6) When you select a tool from the toolbar, you will continue to use that tool until you switch to another. Return to the select tool either by clicking on the select button on the toolbar or by right clicking. If the select tool is not on your toolbar and you would like it to be, add it by looking under "Assign Action" -> "Execute" -> "Select Tool."
- 7) For now, go ahead and load "Designer.tbr," found in the /Game/Interfaces directory.



Static Entities

Streets, buildings, mesh objects, and animated mesh objects are all saved in the mob file. The .scn file in turn, dictates which .mob file to open. All of these can be created either through using the toolbar or through the Context Menu “create object” menu.

Streets

Streets in the Works engine are simple straight line segments. Curved roads and complex intersections will come in later versions.

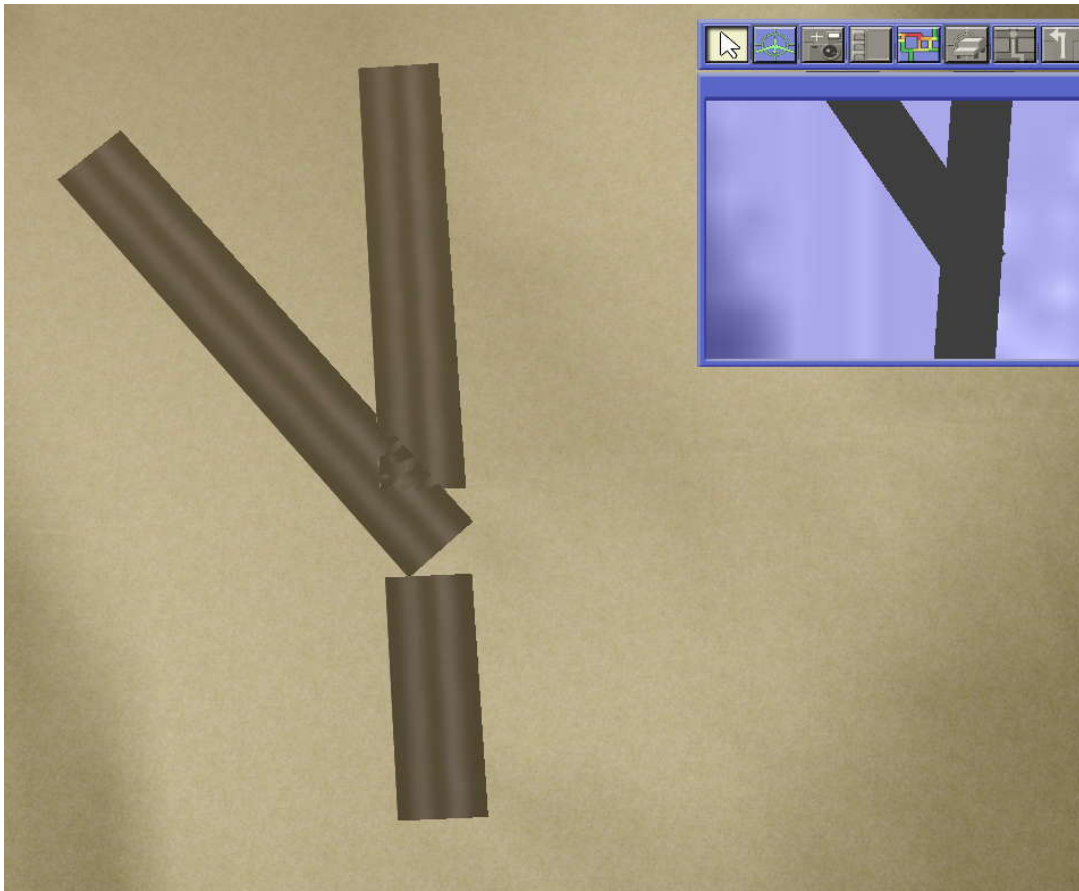
Internal path finding and threat-mapping algorithms treat the streets as a network of intersections and dead-ends connected by single blocks.

As a designer, you do not deal with this detail. Instead, you lay down long street segments and the system will analyze the map. It will discover intersections and connect them, breaking down your long avenues into many streets, each exactly with one block long.

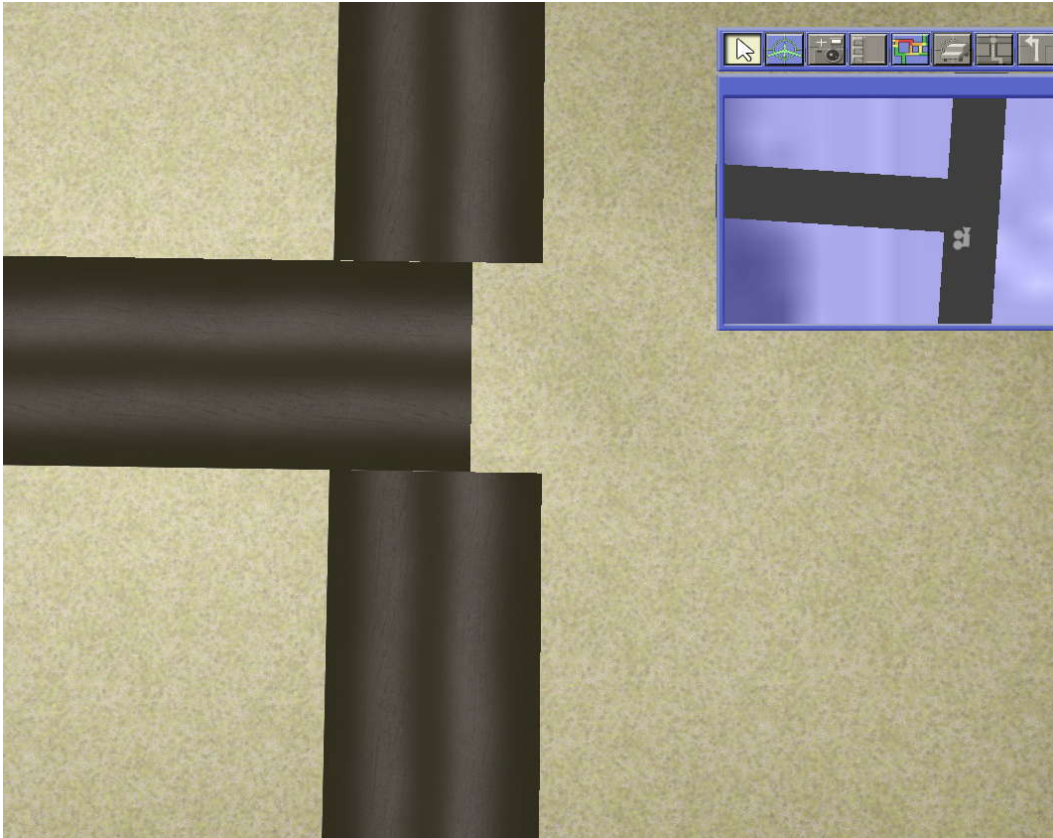
Step by step

- 1) You can create new streets either through the toolbar or Context Menu -> Create object menu.
- 2) With the street tool selected, click where you would like the street(s) created. Any time you want to add a street, use the street tool again.
- 3) To edit the position of the streets, switch to the select tool.
- 4) To drag an entire street, drag the middle section of the street.
- 5) To change the length or direction of a street, drag one end of the street.
- 6) There are certain rules that must be followed when creating streets.
- 7) Streets must intersect at near 90 degree angles. The streets were not made for extreme angles and cause problems as seen in the image below.

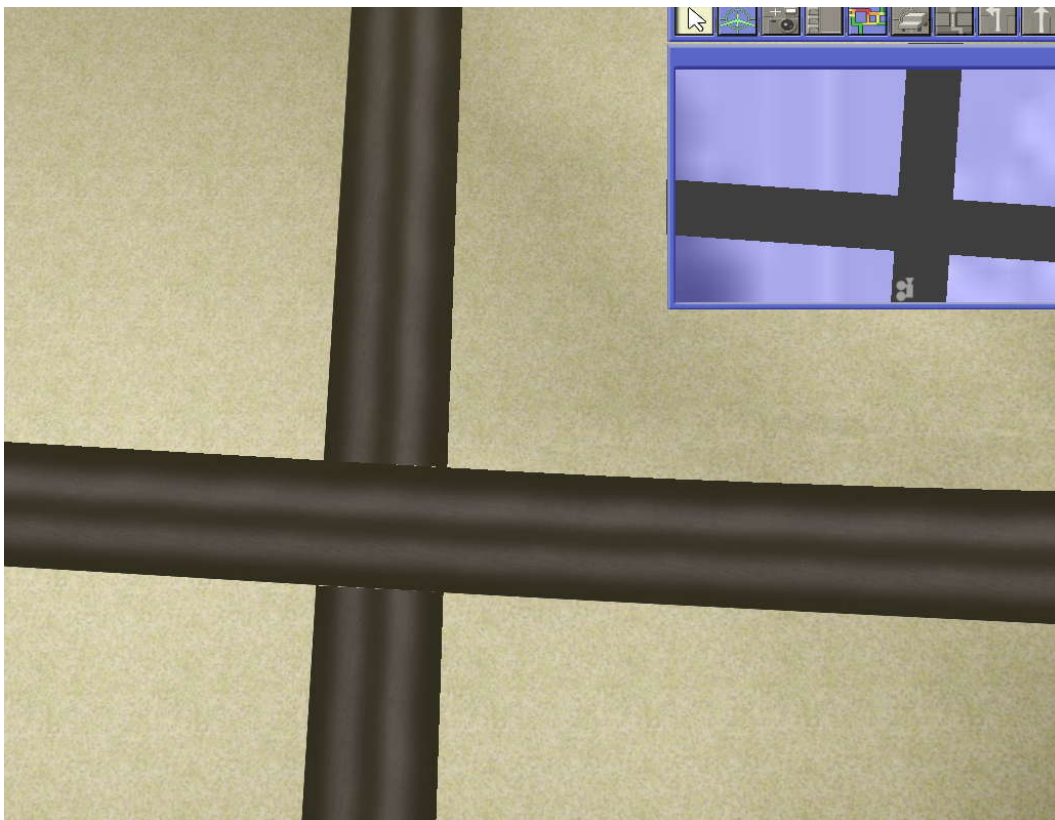
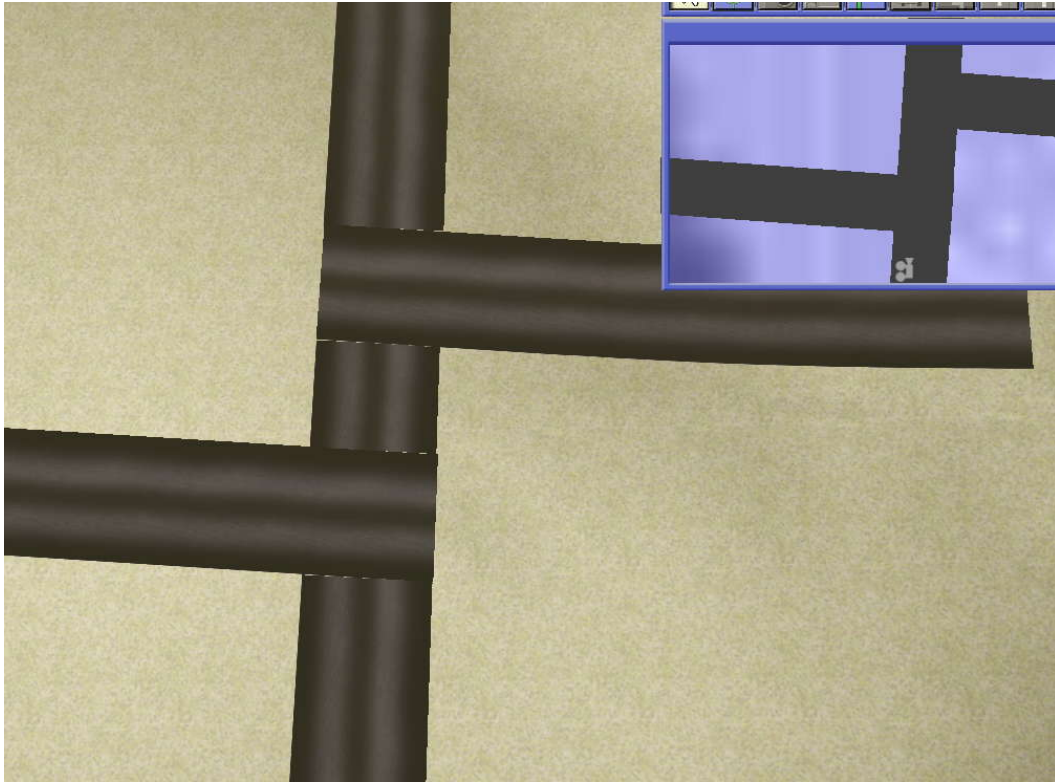




- 8) Streets must extend past the midsections of intersections. If one street does not cross the centerline of the street it is to intersect, no intersection will be recognized, and the convoy will not reroute properly. It is best to align the end of the street with the far edge of the other street. This way the intersection is created avoiding visual problems as seen below.



- 9) Intersections must not be too close to each other. In the next image, you will see three streets and two intersections. If intersections are too close to each other, it may be hard to direct the convoy between them. The example below should either have these streets further apart, or just have two streets with one intersection as seen in the next picture.



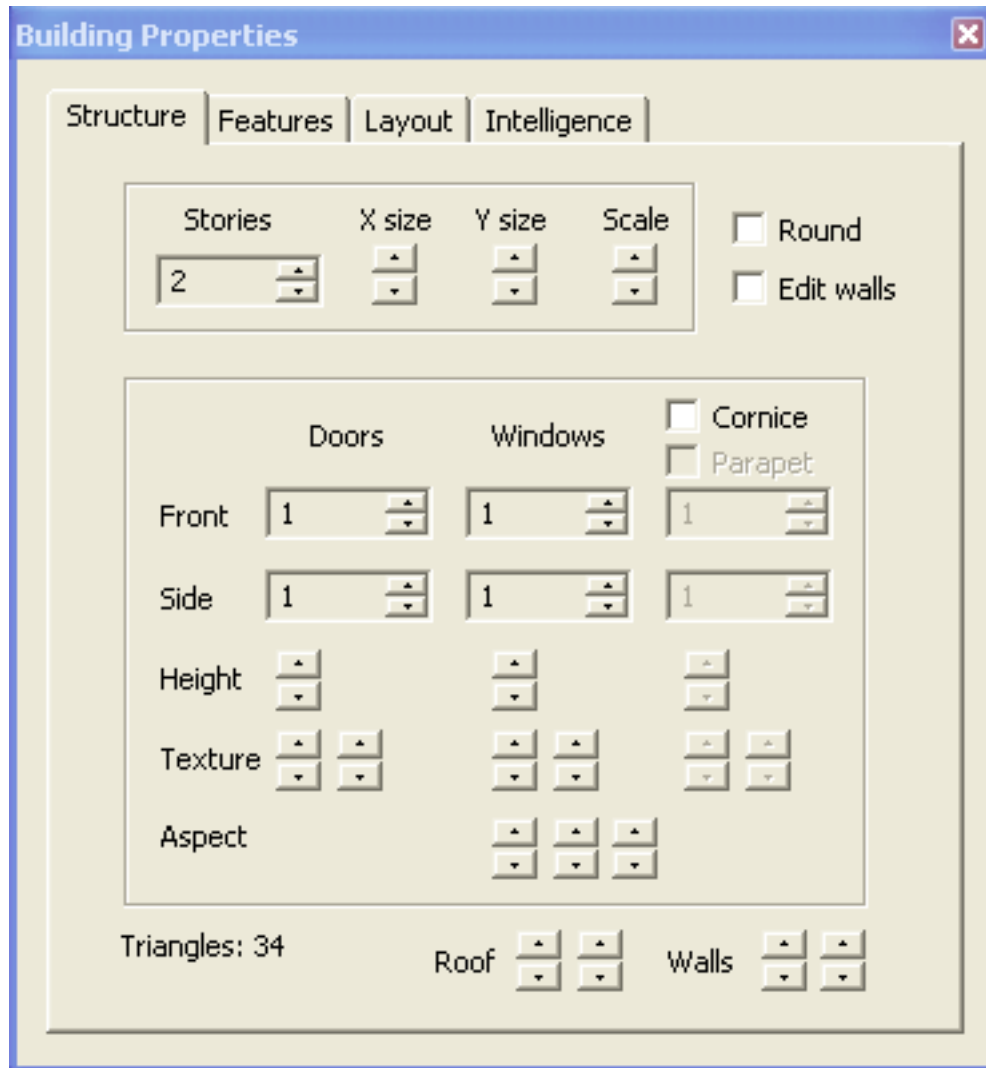
- 10) Double clicking on a street allows you to edit the street's properties, such as texture, color, width, and transparency.
- 11) If you finish editing streets and do not want to edit them accidentally, select "Edit Options" from the Context Menu and uncheck "Streets." You can apply this to anything listed here. Just remember to turn it back on if you want to edit that particular class again.

Buildings

Algorithmic

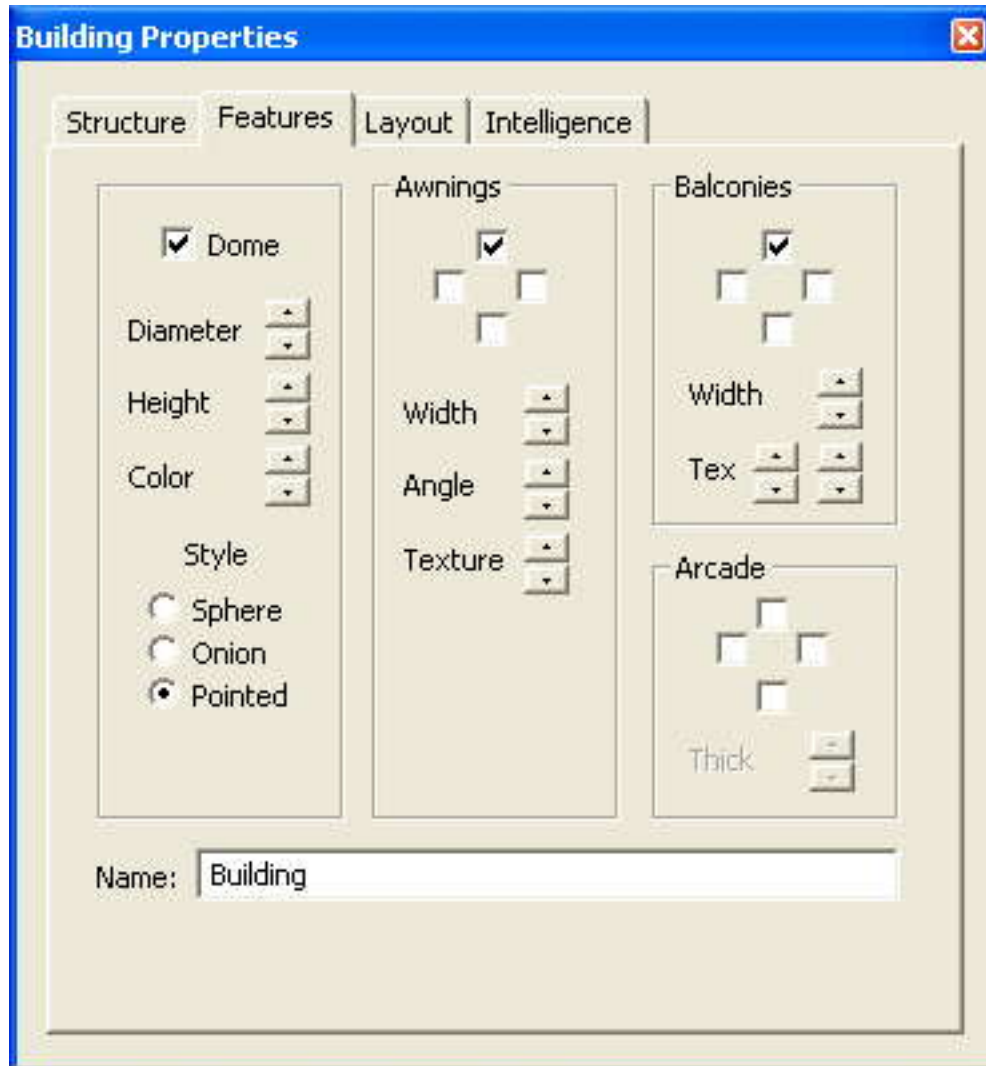
The algorithmic buildings are designed in engine by changing values in the properties dialog.

- 1) When you create a building, either from the toolbar or through the Context Menu, the first option that appears in the list of buildings is *****Algorithmic building*****.
- 2) Select this and click OK. A building will appear along with the properties dialog.
- 3) The first tab deals with the base structure of the building.



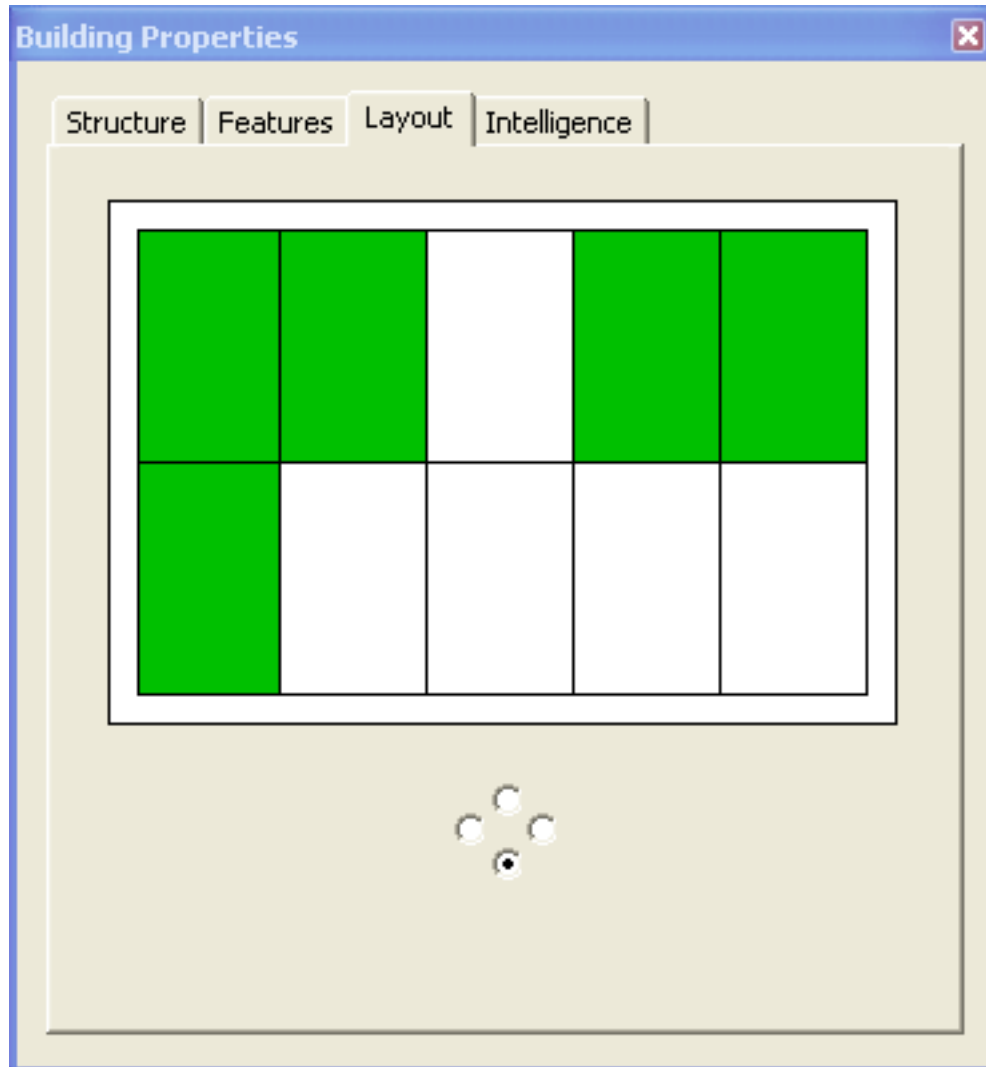
- a) At the top are controls for the scale of the building.
- b) As you may have noticed, changing the scale stretches the textures of the building. Adjust this by adding or subtracting the numbers of doors and windows on the Front and Side. Doors are only on the first story. All the stories above that are where windows will be. Use the first texture up and down arrows to change the texture of the doors and windows. Do not worry if the number of doors and windows are not what you expected them to be. This will be handled under the layout tab, which we will cover soon. The second set of arrows no longer has a functionality and will be removed.
- c) You can add a cornice or parapet to your building, though these will add to the triangle count of the building. You must have a cornice in order to add a parapet. The textures can be adjusted in the same manner as with the walls and doors.

- d) The texture of the roof and walls can be changed in the same fashion at the bottom of this dialog.
- 4) In the features tab, you can control additional structural features of the building. Be aware that these can significantly add to the number of triangles of the building.



- (a) Dome – adds a dome on the roof.
- (b) Awnings – adds an awning on the first story. Each of the four squares represents one side of the building.
- (c) Balconies – adds a balcony on that wall.
- (d) Arcade – opens the doors to reveal inside the building.
- 5) Under “Layout”, you gain more control over how the windows and doors appear on each wall. Earlier when we changed the number of doors and walls in the structure tab, what we were actually doing was adding panels

to the side of the building. Under layout you can determine on which of these panels doors or windows appear.



- a) Use the four radio buttons on the bottom to switch which of the building's walls you are editing.
- b) In the panel above the radio buttons are two rows of rectangles. The top row represents the windows on that wall, and the bottom row represents the doors. Green rectangles will have a door or window texture on them, while white rectangles will have a wall texture. Click on a rectangle to toggle it between green and white.

Freeform

LEVEL OF DETAIL

Scripted Objects

Through the script window you can add events to your scene. This includes events which directly concern the player such as launching the convoy that he must protect and UAVs he must direct. It also includes events in the environment: moving pedestrians and vehicles. Of special concern are the moving threats.

The first three objects listed in the script window should be the three UAVs. Immediately following that should be the protected convoy. Any other scripted objects in the scene should be listed below this convoy.

Step by step:

- 1) Right click to open the Context Menu.
- 2) Mouse over where it says "Scripting." Choose "Author Script..." to open the Script Authoring window. Now that it is open, any time you launch an event, it will appear in the script window.
- 3) Be sure that your three UAV missions are on top, and that your convoy is fourth. This will insure that they appear correctly in the timeline.
- 4) You can drag events to change the timing.
- 5) Scroll with the middle mouse button to zoom in and out.
- 6) "Save..." will save the events in the .scn file.

Convoys

Convoys travel along predetermined roads or mission paths (read the Missions section to learn how to make a convoy travel along a mission path) set by the designer. The convoy can be a friendly convoy that will be tracked and that you want to protect. It can be traffic that is not tracked or even people walking along a mission path, including threats.

FRIENDLY CONVOYS

When editing the convoy manifest, there is an option to make a convoy friendly or not. "Friendly" simply means its location can automatically be tracked. Non-friendly (threats or neutral) convoys must be actually observed. There should only be one friendly convoy in the scene. (Declare all others "hostile").

CREATE A NEW CONVOY

Step by step:

- 1) To create a new convoy route, right click on a toolbar button, choose "Assign Action" -> "Create" -> "Convoy."
- 2) With the "Create a Convoy Route" button selected, the Routing Dialog appears, and yellow dots appear at each intersection in the map overlay.
- 3) Click on the starting and ending intersections. The convoy will use the Dijkstra method to create its path. If you want to define the path yourself, simply click on each intersection, in the order you want the convoy to pass through them.
- 4) Choose which Convoy Manifest will be traveling on the route.
- 5) Make sure the script window is open and press "Launch Convoy."



CONVOY MANIFEST

Step by step:

- 1) With either the "Routing Dialog" or the "Launch convoy on mission" open, you will be given a list of available convoy manifests to send on the path.
- 2) In either of these windows there is an "Edit convoy manifest..." button, with which you can redesign the convoy. Make sure the convoy you want to change is the selected one before you press this button. You can change
 - a) which vehicles are in the convoy.
 - b) how fast the convoy moves.
 - c) the spacing between vehicles.
- 3) whether the convoy is friendly or not. Only friendly vehicles will be tracked. The convoy to be protected must be 'friendly'.
- 4) Now with the "Convoy manifest" dialog open, you can change which vehicles (or people) are in the convoy by dragging from the "Available vehicles" on the left to the "Vehicles in convoy" box on the right. When done, click OK.
- 5) The "Create a new manifest..." button will allow you to create a new manifest from scratch. Enter the name, hit OK, and then edit the convoy vehicles like you would with any other convoy by dragging them from one box to the other.

PREVIEW A CONVOY


Missions

Missions are paths that follow a series of waypoints instead of roads. They can be used for UAVs or for a convoy of vehicles or people that will not be traveling according to roads.

WAYPOINTS


Step by step:

Waypoints are the points through which missions are routed. By dragging the waypoints, one can reshape a flight path as needed. If there are not enough waypoints for the user's needs, it is easy to add them.

- 1) To add waypoints to an already existing mission, right click on a toolbar button, choose "Assign Action" -> "Create" ->  "Waypoint." This will add the "Create a Waypoint" tool to your toolbar.
- 2) Once you have this tool selected, left click in the scene to add a waypoint.
- 3) By default, the waypoint will appear at the end of the most recently selected mission.
- 4) if you want to add waypoints in the middle of the mission, rather than at the end, simply left click near the waypoint next to which you would like to add the new one.

CREATING AND SAVING MISSIONS

Step by step:

- 1) To create a mission from scratch, you will need to use the "Create a Mission" tool. Right click on a toolbar button. Go to "Assign Action" -> "Create" -> "Mission." 
- 2) With this tool selected, simply left click in the scene to start adding waypoints to your new mission.
- 3) If the waypoints do not appear exactly where you would like, switch to the Select Tool to drag them to a new position.
- 4) Also, with the Select Tool, you can double click on a waypoint to open the Mission Properties Dialog.
- 5) Here you can adjust factors such as the altitude or whether or not the mission loops.
- 6) Under the Mission tab, there is a button at the bottom that will allow you to save your mission. Missions are saved as .mis files and can be loaded into any open scenario. If you want the mission to remain in the particular world permanently, you will have to save objects.

FLIGHTS

Step by step:

- 1) To launch a flight, right click to open the Context Menu. In the “Missions” submenu, click on “Flight Manager.”
- 2) If you have other aircraft profiles, you can select which one you want to use under “Aircraft type.” Otherwise, stick with “Default.”
- 3) The mission that you last had selected will initially be selected in the “Available missions” list automatically. Whichever one is selected will be the mission on which the flight will launch once the “Launch Flight” button is clicked.
- 4) If the script window is open, the flight launched will be added to it.

PREVIEW A MISSION

Previewing a mission will take main scene camera along the selected mission path.

USING A UAV MISSION AS A CONVOY ROUTE

Step by step:

- 1) In the Missions Properties dialog (double click a waypoint to open), under the “Mission” tab, change the Mission type to “Convoy route.”
- 2) Now, when you access the flight manager through the Context Menu, instead of opening the flight manager, the program will open the “Launch convoy on a mission” dialog
- 3) In this window, you will be given a list of convoy manifests that you can launch on the convoy route mission. These can consist of any models, including both vehicles and people.
- 4) If you want a convoy manifest that is not listed, follow the steps in the “Convoy Manifest” section for creating a new manifest.

AIRCRAFT

The .apf files contain the set values for the aircraft’s speed, turning radius (which is also its radius of rotation when circling a point in the gimbal cam), roll angle, and climbing and ascending rates, as well as sensor information. The .apf files can be selected in the Flight Manager dialog under “Aircraft type.”

SENSOR

The sensor controls the cameras the UAV cameras the user will see.

- 1) FieldOfViewAngle – controls the field of view of the camera
- 2) Spectrum – controls the render state of the UAV windows.
- 3) “0” sets the cameras to typical real time rendering.
- 4) “1” sets the cameras to data fusion mode rendering.

Environment

Terrain

Works Engine supports large scale, high resolution terrain.

In this context, terrain refers to the topological landform on, in and above which game objects are located. Terrain is represented as a continuous unfolded surface with a variety of properties. For every point on the map, there is one and only one set of terrain properties. The principal properties include geometry and appearance.

Geometry

The geometric properties of terrain are simple. Each point has a geometric location in X, Y, Z space. X and Y are map coordinates corresponding to specification in a lat/long grid. Z is elevation. Additionally the terrain system retains the slope of the surface at each point. However slope information is not independent of the simple XYZ locations. Slopes can be, and are, derived from simple location, so the fact that the terrain system stores these is a detail of implementation. You can usually ignore issues of slope.

You cannot ignore issues of memory requirements. The dense data over very large areas present issues of memory management. The Works engine has a sophisticated system to permit streaming of terrain data. It requires caching active areas of terrain while rolling in new data from slow disk storage.

HEIGHT FIELDS

As in most game engines, the geometric terrain surface in the Works engine is represented as a height field. This is a two dimensional array of elevation values. The array is evenly spaced and all values are present (there are no holes).

The terrain data is divided internally into tiles, each of which contains 100 x 100 data points. These are the units of terrain which are managed in the caching system. Several levels of caching exist: the Works engine caches the 24 most recently active terrain tiles in graphics (GPU) memory that is immediately accessible to the graphics processor. The most active 200 tiles are cached in system (CPU) memory. Additionally, there is usually a cache maintained by the operating system, or the hardware manufacturer, between the disk drive and the CPU memory, and a small cache within the GPU itself which is faster than even the GPU memory.

All of this has implications for the scenario designer.

- 1) Since the amount of data is a function only of the array density and the surface area, you must choose your scale carefully. You must ensure efficient performance with full coverage of your area of interest.
- 2) Since data density is constant, performance is not sensitive to surface complexity. A highly detailed surface renders no less efficiently than a flat plane. You can exploit this fact to deliver more interesting land features.
- 3) There are no caves or overpasses. Every point on the map is represented by exactly one height.
- 4) Vertical surfaces (cliff faces, walls, building fronts) cannot be represented. The distance between postings (elevation points) is fixed everywhere on the map. A vertical surface would require that distance to approach zero. The steepness of the surface is limited by the posting distance. (Extremely steep surfaces look shabby in this terrain format for related reasons, as we will see later.)
- 5) Streaming works best with predictable changes in terrain. Terrain that is overflowed by a plane or traversed by a vehicle or pedestrian will be very easily managed by the caching system. The spooling of data is orderly and easily within the capacity of the system. Huge databases can be streamed with smooth performance.
- 6) When the field of view changes rapidly, large strains are placed on the caching system. A slewed camera will place large demands on the terrain pipeline. A system where several cameras render different parts of the terrain simultaneously can cause terrain 'thrashing' as the cache empties and refills every frame. Multiple fields of view can be simultaneously maintained only if each subtends only a small terrain area. Low or wide cameras consume more terrain than down-pointing cameras with narrow FOV.
- 7) Cache-thrashing and similar problems are eliminated when the entire terrain fits into the CPU or the GPU cache. The CPU holds 2,000,000 data points. The much faster GPU cache holds 240,000 points.

There are two basic sources for terrain height field data: reality and imagination. The former requires conversion of external files.

Opening and saving .SEM files

Step by step

- 1) WORKS terrain files are identified by the .SEM (Streaming Elevation Map) extension on their file names.

- 2) To open a new .SEM, right click to open the Context Menu, and go down to "Lighting controls..." and click to open the Lighting manager.
- 3) Open the tab that says "Layers."
- 4) Click on the button that says "SEM File"
- 5) Select the .SEM file you would like to open.
- 6) Click on Save Profile. Which .SEM file will open is saved in the .TLP file.

Import from Standard files

Step by step

- 1) To create a .sem file, locate DemMaker.exe.
- 2) Drag and drop the foreign terrain data files into DemMaker. Be sure the files conform to a supported format.

DTED IMPORT

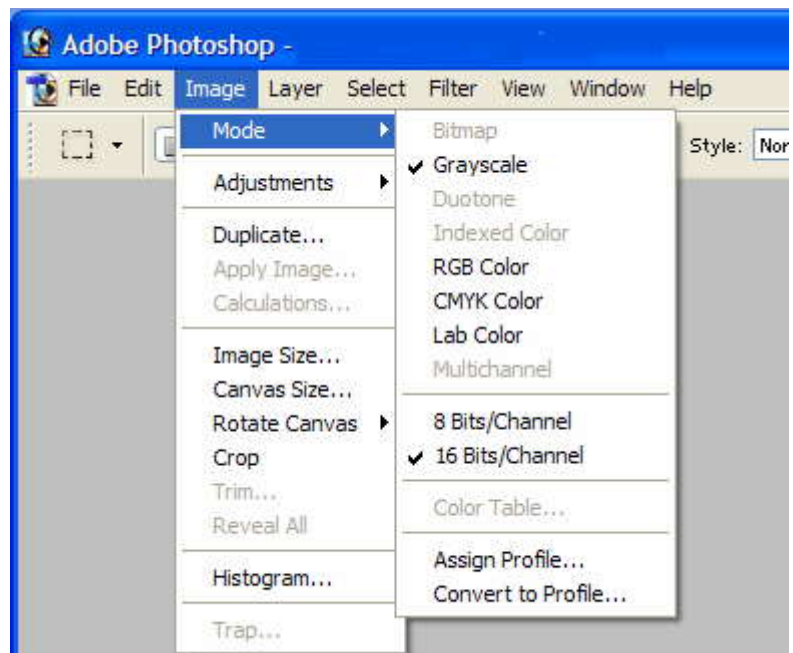
- a) DemMaker will read files conforming to MIL-PRF-89020B, the DTED standard.
- b) DTED 2 resolution is appropriate in most cases, but other resolutions can also be converted.
- c) Although DTED is the best supported, DemMaker also will interface with other GIS standards such as TIGER.

Add files of adjacent areas. DemMaker will stitch these into a single large area streamable file.

Terrain Painting

Step by step

- 1) To create a height field from scratch, you can use a painting program, such as Photoshop. Select **Image / Mode / Grayscale**
- 2) The height field will consist of a grey-scale image in which each pixel represents one elevation data point. White pixels are the peaks, and black pixels are at lowest points.
 - a) 8 bit grayscale are the standard in Photoshop.

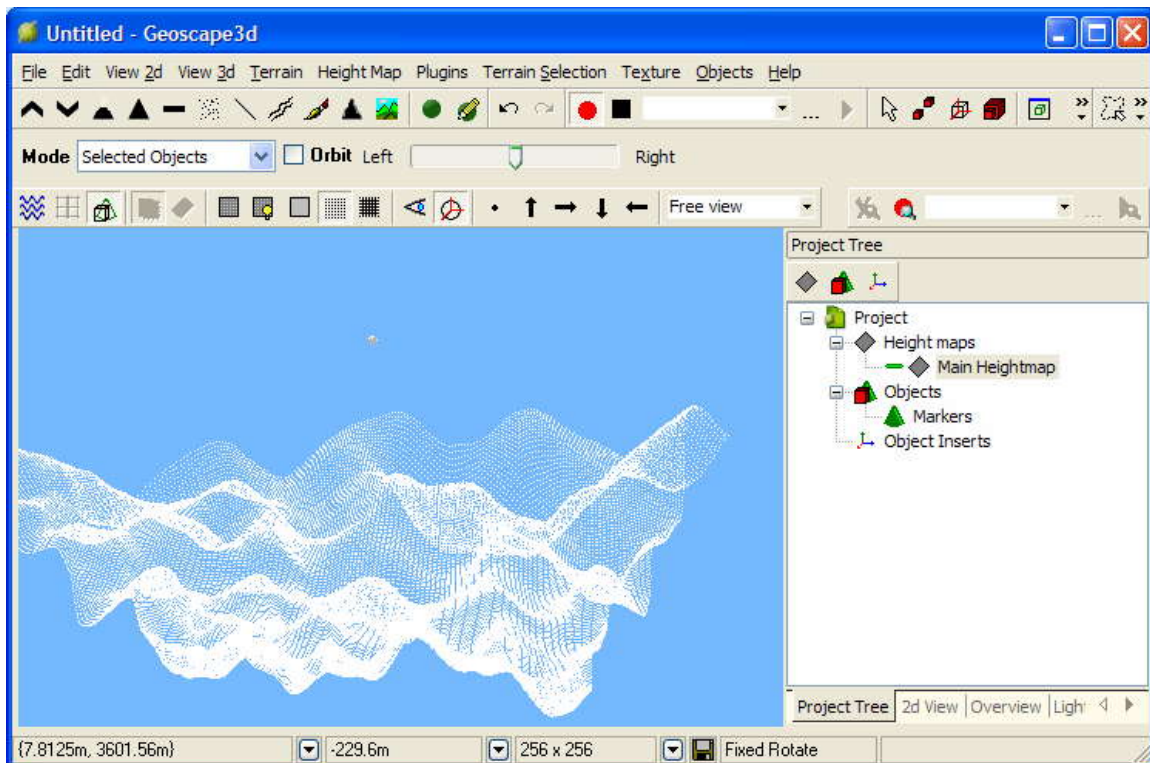


In general these will not yield satisfactory elevations maps. They will only have 256 elevation levels which is not enough dynamic range to avoid conspicuous stair-step artifacts.

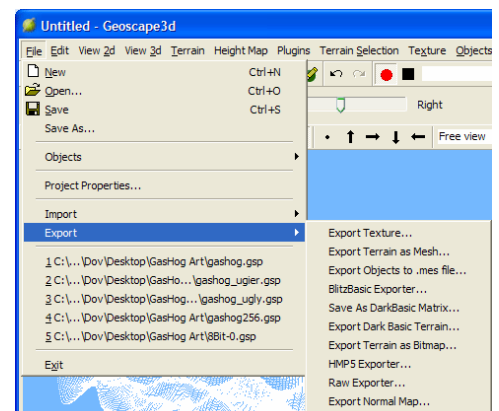
- b) 16 bit grayscale images are perfectly acceptable. Use **Image / Mode / 16 Bits per Channel**
- 3) Use all available painting tools to create a direct height field in which brightness is directly proportionate to elevation.
- 4) Save the file as a RAW file for DemMaker.

Terrain Sculpting

Step by step



- 1) A much more satisfactory tool is a 3D terrain editor. We recommend GeoScape3D (thegamecreators.com).

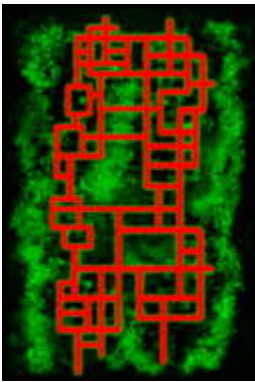


- 2) Use the tools available within GeoScape to create, import, view and edit 3D height fields. Please refer to GeoScape documentation.
- 3) When complete, use the Export Terrain option.
- 4) The **SwarmDem** plug-in for GeoScape allows you to save **.sem** files directly from the GeoScape menu.

Appearance

The geometric form of the landscape will give the terrain an interesting appearance. The contours will be visible and can be enhanced by good lighting. However it is possible to create more meaningful and realistic terrain scenes by applying color and imagery to the surface.

Terrain Textures



Up to three different textures can be applied to a map. These are the Base Texture, Texture One and Texture Two. Each is a separate map.

A fourth map, the Control Texture, determines how the other three blend, based on the intensity of its Red and Green channels.

Where the Control Texture is black, the Base Texture shows through. It can be a 1:1 texture (spread out across the entire terrain), or a detail texture which tiles.

.The Red color defines where Texture One will appear, The more intense the Red channel, the denser Texture One will be. Texture Two will appear wherever Green occurs. All terrain textures are saved in the /Game/Terrain directory.

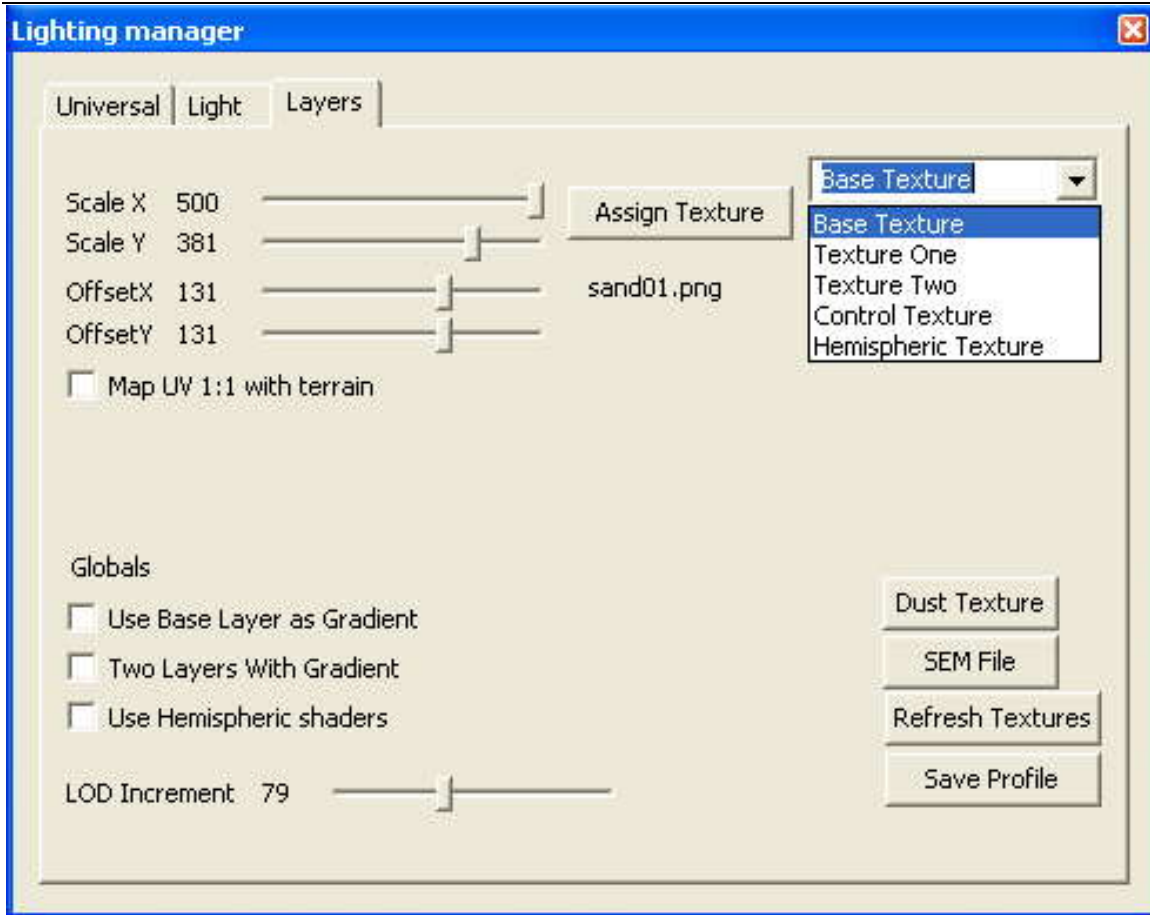
Detail Textures

The detail textures need to be repeating textures unless they are going to be set to 1:1 with the terrain. The alpha channel in the detail textures controls the specular map of the terrain; lighter areas will be more specularly lit. Specular lighting will be covered more in the lighting section.

Applying Textures



Step by step



- 1) Right click to open the Context Menu, and go down to "Lighting controls...", and click to open the Lighting manager.
- 2) Open the tab that says "Layers."
- 3) In the drop down menu on the right, select "Control Texture."
- 4) To the left of this drop down menu is the "Assign Texture" button. Click on this, and find the file saved earlier inside the /Game/Terrain directory.
- 5) You can now choose which textures go where. Go back to the drop down menu and choose Base Texture.
- 6) Assign a texture to the Base Texture the same way you did before by using the "Assign Texture" button. Once you pick a texture, you should see all the parts that were black on the Control Texture change to match the texture you selected just now.
- 7) The scale of the texture should match the size of the world. If the sand on the ground looks very large next to tiny buildings, it will make the world look fake. On the left are Scale X and Y sliders. These control how often your textures repeat across the terrain, and as a result, the scale as well. As you slide them back and forth, you can see the Base Texture adjust

accordingly. If you do not want the textures to repeat at all, check the box that says “Map UV 1:1 with terrain.”

- 8) Offset allows you to slide the texture across the world, though this is not usually needed if it’s just a repeated texture.
- 9) Use these same steps for Texture One (red) and Texture Two (green).
- 10) When you are finished editing the terrain, be sure to save using the “Save Profile” button. The textures are saved in the .TLP file.

Creating New Textures

Step by step

- 1) To create a control texture from scratch, you will have to use a painting program such as Adobe Photoshop.
- 2) Modulate the texture blend by applying Red and Green.
- 3) The map should be the same proportions as the terrain.
- 4) Painting Streets is optional. The Street’s Procedural visibility can be turned on or off.
- 5) When done, save the file in /Game/Terrain.
- 6) In the Works engine, right click to open the Context Menu, and go down to “Lighting controls...,” and click to open the Lighting manager.
- 7) Open the tab that says “Layers.”
- 8) In the drop down menu on the right, select “Control Texture.”
- 9) To the left of this drop down menu is the “Assign Texture” button. Click on this, and find the file saved earlier inside the /Game/Terrain directory.
- 10) You can now choose which textures go where. Go back to the drop down menu and choose Base Texture.
- 11) Assign a texture to the Base Texture the same way you did before by using the “Assign Texture” button. Once you pick a texture, you should see all the parts that were black on the Control Texture change to match the texture you selected just now.
- 12) The scale of the texture should match the size of the world. If the sand on the ground looks very large next to tiny buildings, it will make the world look fake. On the left are Scale X and Y sliders. These control how often your textures repeat across the terrain, and as a result, the scale as well. As you slide them back and forth, you can see the Base Texture adjust accordingly. If you do not want the textures to repeat at all, check the box that says “Map UV 1:1 with terrain.”

- 13) Offset allows you to slide the texture across the world, though this is not usually needed if it's just a repeated texture.
- 14) Use these same steps for Texture One (red) and Texture Two (green).
- 15) When you are finished editing the terrain, be sure to save using the "Save Profile" button.

Brush – is not currently applicable to Avant Guard

Lighting

There are two different types of lights we will discuss. The first one is the scene light, which affects the entire scene no matter where you are. There are many different components to it, which we will discuss later. The other types of lights are individual lights which only affect the area in their immediate proximity. Only the scene light affects shadows.

Lighting Components

- 1) Open the lighting manager by right clicking to open the Context Menu and selecting "Lighting controls..."
- 2) In the first tab, Universal, are the various lighting components that can be adjusted for the main scene light.

Here you will find properties you can change for the scene light. In the top part are the actual light properties. Below that are fog properties. The fog allows the scene to gradually fade into the distance so that only things near the camera will be rendered. With the fog properties, you can control how near this fading occurs. The list of color schemes below that is specifically for older machines that cannot take advantage of the terrain shaders. Note that fog color is in this section as well. On the top right is a gizmo where you can control the direction of the main scene light.

AMBIENT

Ambient intensity controls the non-directional 'fill' light. The higher this is, the lighter the shaded surfaces will be, and the lower, the darker they will be. Low ambience increases scene contrast and modeling. Too much ambient light can make the scene look "washed out."

DIRECTIONAL

Directional intensity controls the amount of direct light cast by the scene light.

SPECULAR

The specular light setting determines the terrain's glossiness. This does not affect objects other than the terrain. All parts of the terrain will be equally specular unless it is otherwise defined in the alpha channel of the detail texture maps.

Intensity

The specular intensity controls how bright the specular highlight on the terrain is.

Power

The specular power controls how concentrated the specular highlight on the terrain is. If the power is low, the specular highlights will be spread widely across the terrain – and the effect is satiny. If it is set high, the highlights will be concentrated and the effect is glassy.

DAYLIGHT

The Daylight slider controls the total amount of light given by the scene light for ambient, directional, and specular.

Light Source

The second tab, the one labeled “Light” can only be accessed if a light has already been placed in the scene. Unlike the main scene light, these lights only illuminate the area in their immediate vicinity.

Step by step

- 1) To add a light, add a “Create Light” button on the toolbar through “Assign Action” -> “Create” -> “Light.” Alternatively, go into the Context Menu, go down to “Create object” and click Light in the sub menu.
- 2) To edit the light’s position, you will need to temporarily turn on the appearance of lights. Go back into the Context Menu, go down to “Appearance” and in the sub menu choose “Lights.” You should now see a light bulb symbol where the light was placed. As long as “Light” is checked on under appearance, these bulbs will appear, so you will want to turn it off when you are done editing.
- 3) There is a limit of four lights per terrain tile. If you are unsure where the terrain tiles are, go back to Appearance in the Context Menu and turn on “Wireframe.” This will reveal a grid across the world divided into large and small squares. Each large square is one terrain tile and can only be affected by four lights. Remember, if there are moving lights, (e.g.: headlights) above a tile, these count toward the limit of four.
- 4) Make sure the Select tool is in use, and then double click on the light to open the lighting manager. In the top left you will see types of light you can choose, including
 - a) a point source, which distributes light equally in all directions from a single point
 - b) a spotlight, which shines light in the Z axis

- 5) To change the color of the light, click on “Choose color...” just below type. This will bring up a typical windows color picker. The color in the box above where it says “Color/Solid” is the color you have selected.
- 6) There are a few ways to control the brightness of your light.
 - a) The brighter the color of the light, the brighter the light will be.
 - b) The higher the “Range” is, the brighter the light will be.
 - c) “Angle” only works with spotlights and affects how wide of an area the spotlight covers.
 - d) The type of fall off affects the brightness of your light at a close proximity to the source versus the brightness of its edges.
- 7) The direction only matters for spotlights, as point lights distribute their light evenly in all directions. The Z axis is the direction in which your light will point. Change the direction by dragging the axis in the white box to rotate it.
- 8) Checking on “Animate Range” will cause your light to pulse slightly.

Hemispheric

AZIMUTH

Light coming from the direction of the sun will be very bright, and slightly yellow. Light coming from other directions will be bouncing from the sky, and will be bluer as it is further from the sun. A circle of color describes this transition, with bright, yellow-white at one point and dim blue light diametrically opposed to it.

ELEVATION

In considering the azimuth, sunlight is assumed to always be bright and yellow-white. But this is not always true. As the sun ascends in the sky, the light it casts changes from “rosy-fingered” dawn to the yellow-white glare of noon, and then it again becomes redder as the sun begins to set again.

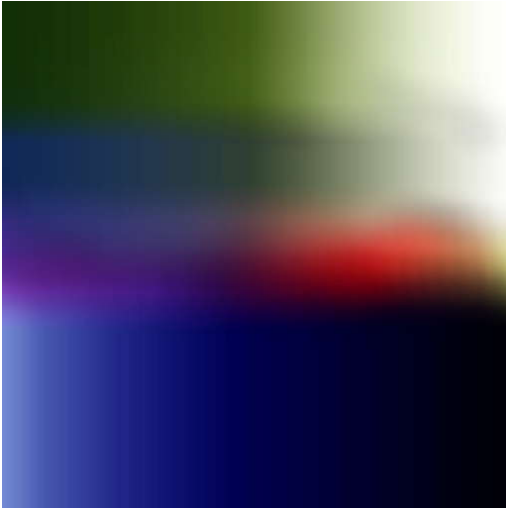
This consideration suggests that the azimuthal ring that associates light color and direction is only one of a series of rings each of which corresponds to a different time of day (different sun elevation.)

MAP

How-to

- 1) Simply enable hemispheric lighting by click on the **Use Hemispheric Shaders** checkbox in the **Layers** tab of the **Lighting Manager** dialogue.
- 2) Usually you are done at this point. However you might want to create your own light map. If so, proceed.

- 3) Using Photoshop or the equivalent paint a map. High resolution is not necessary. 128x128 is plenty of data.



Hemispheric Light Map

- 4) The right edge represents the sun position (azimuthal) and the left is purely skybounce.
- 5) The top of the map is noon, and the bottom is midnight, with the band in the center representing the light at both dawn and dusk.
- 6) The bottom half of the map allows for a moonlight illumination.

Select this file in the lighting manager, by using **Assign Texture** with **Hemispheric Texture** showing in the dropdown selector.

Atmospherics

Shadows

Shadows add great realism to the scene but they must be used with caution, since their computation requirements impose a large cost on frame rate.

In fact, the performance cost of shadow rendering is so high that its use is ultimately determined by the Player who is best able to judge the capabilities of the current hardware and the quality of the resulting display.

For shadows to be visible they must both be selected by the designer and they must not be disabled by the Player's video quality settings.

Two shadow techniques are supplied.

SHADOW VOLUMES

In this system, volumes of darkness are geometrically created behind each shadow-casting object by extruding its silhouette into a prism. A sphere creates a cylindrical shadow volume. A cube might create a rectangular or hexagonal prism depending on the light angle.

It should be clear that this is a very costly operation, especially for animated objects, since the profile of the shape must be determined and the volume extruded before the complex shadowing can occur.

This technique can yield very precise shadows but it is only appropriate where the scene is very simple and has few shadow casting objects.

SHADOW MAPS

This technique casts the light into the scene to create a separate rendering in 'light space' before the normal rendering in 'camera space' is performed. The time required to rendering this shadow map and then refer to it during the normal render can easily double the total render cost of the scene. It has an additional cost in graphic memory, since a large surface must be reserved for the shadow map.

However this technique is fast and reliable and produces fewer anomalies than the volume technique. It can render shadows from detail in the texture-maps (e.g.: alphatest maps) as well as geometric detail. In most scenes it is faster and superior to shadow volumes. Its major defect is that small shadow maps (chosen to conserve memory) can exhibit 'pixilated' low- resolution shadows.



Shadow Volume



(b) 512 Shadow Map



(c) 2048 Shadow Map

Step by step:

- 1) The objects must have shadows enabled.
 - a) Objects which will cast shadows must be flagged. Most already are.
 - b) For objects that are not casting shadows. Double click on each distinct object (but not each instance) to open the mesh properties dialog.
 - c) Select **Casts Shadows**.
- 2) The scene must have shadows enabled.
 - a) Right Click in the scene itself to open the **Main Edit Menu**.
 - b) Select **Appearance / Shadows / Use Shadow Mapping** (or, in rare circumstances, **Volumes**)
 - c) Select **Open Shadow Mapping Dialog**
 - d) Choose smallest map with acceptable results.
- 3) The machine must have shadows enabled.
 - a) Hit **Escape** to open the **Main Play Menu**.
 - b) Select **Video Options**

- a) Check **Shadows**
- 4) The machine must have Hemispheric shaders enabled.
 - a) Open the lighting manager through the Context Menu.
 - b) Go to the Layers tab.
 - c) Check on “Use Hemispheric shaders.”

Sky

The Works engine allows the designer to place a “sky dome” behind the furthest visible objects and the most distant terrain. This sky dome is created by wrapping an image of the sky around 360 degrees of horizon, and extending it upwards from the horizon to the zenith.

Of course, fitting a rectangular picture to the inside of a hemisphere introduces distortion of the image. The supplied sky picture must compensate for these. It must have the familiar anamorphic distortion seen in a Mercator map, where Greenland and northern Canada are monstrously wide. The Mercator projection is a simple unwrapping of a hemisphere onto a rectangle. Just the same the sky image stretches more horizontally as you approach the zenith. The bottom edge is the circle of the equator while the top edge is the stretched-out singularity of the North Pole.



A typical Sky Dome source image

Step by step:

- 1) Assemble your sky imagery into a single picture in an image editor such as Photoshop. An image size of 1024 x 256 is appropriate.
- 2) Use paint tools to be certain that the sun is not obvious in the sky, and the direction of light is as ambiguous as possible, for best results.

- 3) Match the right and left edges perfectly smoothly. A useful tool in this process is Photoshop Filter / Other / Offset with WrapAround selected and Horizontal set to 100 or so.
- 4) Introduce the Mercator distortion, using a tool such as Photoshop's Spherize filter and considerable manual manipulation.
- 5) Be certain that the right and left edges still meet smoothly. You do not want a seam in the sky.
- 6) Adding blur near the horizon will improve results.
- 7) Save the file in bmp format as in the Game/Horizons directory. Name the file carefully.
- 8) Select this file in the Lighting Manager.
- 9) Your changes are recorded in the .MOB (Multiple Object) file that records all the scene contents.

Fog

Fog serves both an aesthetic and a functional purpose in a game engine. The aesthetics are self explanatory, but the functionality requires an explanation of practical geometry.

From any point where the camera is positioned, the field of view extends forward as a pyramid, with its tip at the lens of the camera. The further one goes from the camera, the more the content is within the field of view.

In the real world, this pyramid (called the viewing frustum) often extends infinitely: you can see the mountains miles away, and stars light years further. In a synthetic world, this causes problems. Everything that will appear in the view must participate in a variety of mathematical operations before it is rendered. These operations use CPU cycles.

As distance from the lens increase, objects become less distinct and less significant, but they actually impose a greater load on the rendering pipeline. This is because objects in the distance are more numerous – they are in the wide part of the frustum, where hundreds of objects can fall within the field of view.

It is necessary to choose a “far clip plane” beyond which it is not worthwhile to render objects because they are rather costly and fairly worthless.

To ease the sharp falloff at this edge, it is feathered with fog. The fog blends in: it is slight at its near edge and solid at its far edge, which corresponds to the far clip plane.

You can set these to match the requirements of your scenario.

How-to

- 1) It is advisable to use a frame-rate measuring system.

- a) Use a third-party frame rate monitor. FRAPS is, deservedly, the industry standard.
 - b) Alternatively, you can use the built-in frame rate monitor by selecting the **Main Edit Menu** with the right-click, and selecting **Animation / Show Frame Rate**
- 2) Line up the scene camera in a complex part of the world, looking far into the distance where there is plenty of geometry and terrain.
 - 3) Optionally, pause the simulation with the **P** key. This will allow for very precise comparisons, undistorted by changes in the actual scene content.
 - 4) Use the Fog Start and Fog End sliders to adjust the fog distance and falloff.
 - 5) Balance performance and aesthetics.
 - 6) Save the .TLP file.

Rain

The weather system is rather experimental at this point.

- 1) Press the ` key to open the console
- 2) Type in "rain"
- 3) Hit enter to make rain appear.

Lens flare

The lens flare in the scene can be turned on and off in the Context Menu under Appearance -> Lens flare.

Model Framework

Models are systems made up of components (physical objects), and their actions (how the objects act and react within the system). Models are designed to give the designer more control and allow more complex relationships in the world. Ultimately the components of a model will be able to interact with each other to create a system of gestalt.

Model Editor

Model Editor basics



The Editor consists of the title bar across the top and menu buttons below that. Under these are the component and action drop down menus, where your currently selected component and action are displayed. Beneath these is the timeline. Below the timeline will be graphs of the action, which we will cover later.

Creating a new model

Step by step

- 1) You can create a model by either using the "Create a Model" button from the toolbar, or by going through the Context Menu "Create object" menu.
- 2) When the model selection menu appears, hit "New" on the right. An empty model will appear along with the model editor dialog. However, before it can be very useful, you will need to add components and actions.

Editing a model

Step by step

- 1) To edit a model that already exists open the model selection menu as if creating a new model.
- 2) Instead of clicking New, choose the model you would like to edit from the list.

3) Click “Create.”

The Model Framework

Components

The components of a model are simply the objects in it. These objects generally have a physical location and an appearance: the player and others can see or hear or feel them.

There are several classes of components, each with different purpose and different characteristics.

Animated Mesh

The Animated Mesh is a component that includes mesh geometry and textures, and also includes a system of bones and a repertoire of bone animations.

The framework editor does not manipulate bones directly, but instead manipulates the bone animations, which are included as actions. The animations can be stretched, modulated and blended together within the framework editor.

Emitter

Emitters are spawn points. They can spawn particle systems, to create special effects such as fire and smoke. Alternatively, they can spawn entire new models from within this model framework

Light

The model framework allows you to animate a light's color, brightness, direction and position. You can also set the scale of its influence (allowing control over its rendering cost).

Mesh Object

The mesh is the most basic of the components. You can alter their position and proportions, but their shape does not change, and they do not have animations of its own.

MODELS

Models can have entire other models as components. Many times the designer will have more control by having creating a separate framework within this framework, just as the earth/moon framework is within that of the solar system.

MORPH MESH OBJECT

Morph Mesh Objects can change shape but have no bones. Instead, they have up to five alternate versions of the mesh object, each one presenting an extreme pose. Using the framework editor, animation is created by blending these extremes over time.

Sound Object

Sound objects allow your model to play sounds. You can adjust the frequency and the volume as the sound plays.

Triggers

Triggers are invisible but sensitive volumes within the framework. They are represented in the editor by a cage that cannot be seen during game play.

Triggers can react to collision with other models (or mouse clicks) by sending messages through the framework system.

Actions

Actions define how the components of a model act and react. They are composed of keys and channels. Keys specify the time that a change happens. Within each key, channels specify the property that changes, such as position, size, or even more complicated events, such as a message to end the game. Each component has channels that correspond to its attribute. Many of these are common (i.e.: position) while others are more specialized (only the sound object has a “volume” channel.)

Keys

Keys specify the time that a change happens. Each key contains the information of all active channels linked at that time. When you have a key selected, clicking on “Key” will show you all the current active channels for that key. If you do not have a key selected and click on “Key” and select a channel you will add a new key containing that channel.

Channels

Channels are the attributes of a component.

- 1) Translation – the location of the component in model space
- 2) Rotation – the orientation of the component rotating around its own center in model space
- 3) Scale – the size of the component, with X, Y, and Z independent or locked.
- 4) Morphs – blends the weight of morph mesh targets [Morph Mesh]
- 5) Volume – controls sound volume [Sound]
- 6) Frequency – controls sound frequency [Sound]
- 7) Color – adjust the color of the light [Light]
- 8) Spawn – controls what an emitter will spawn [Emitter]
- 9) Message – sends a message to models or components

- 10) Controlled Action – blends the weight and manipulates the flow of an action. Controlled actions must be pre-assigned under the “Action” menu.
- 11) Threat – determines the type of threat, the apparent threat value (used by Automated Threat Detection) and whether it is an actual threat or not (used for scoring) [Mesh and Animated Mesh]

The New Action

The New action is a special action that defines the properties of a component when it is first created. The New action is a special form of action called a state. States only have one key, and they do not become invalid when another action takes place. At this time the only state is the New state, and it is in always in effect. In the future, user-defined states (such as “Destroyed” or “Hidden”) may be possible and transitions between these will be defined in the framework.

Step by step:

- 1) Click on the “Component” button.
- 2) Hover the cursor over “New” to bring up a menu of available components and add a new component. The simplest component to add is the mesh object. It is a simple geometric shape that does not have any animations of its own. We’ll start by adding one of these.
- 3) Click on Mesh Object and choose “Abdul Pose Argue Back” from the list. Click “Create”.



- 4) In the component list on the left is “Abdul Pose Argue Back 00.” This is your selected component. In the action list on the right is “New.” This is your currently selected action. There will not be more options in either of these lists until we add them.
- 5) If at any time you want to change the properties of your component, click on the Component button. The option “Mesh properties” will bring up the Mesh Properties dialog for the selected component. This will not affect other versions of this mesh. It only affects the mesh within the model.

- 6) Let's adjust the location where the mesh will begin in relation to the model. Click on Key, and select Translation. This adds the translation channel to the New action. The model editor window will expand down.



- 7) There will be three bars, or sliders, in the window (though you may only see one at first). Each bar represents an axis of translation, blue for Z, green for Y, and red for X. You can toggle which are visible by using the corresponding buttons on the left. Note that this will adjust the values within the model framework space, not in world space.
- 8) There are a few ways we can adjust the values.
- a) Slider.
Toggle the buttons on the left until you can only see the X value, or the red bar. Move this bar gradually up and down, and you will see the mesh move forward and backward in the X axis.
 - b) Gizmo.
At the origin of the model, you should see the gizmo. It has tools for translate, rotate, and scale. If you only want to affect one channel, you can turn off parts of the gizmo by clicking on "Tools" and toggling the tool button to change the visibility.
The axes of each tool are color coded. The red arrow on the gizmo affects the value of the X axis of position. Drag the arrow to drag the mesh.
 - c) Keyboard.
Upon the creation of the translation key, a window headed "Key Info at Time 0" appears. This window contains all the values of any channel stored in the selected key. Here you can type in specific values for each channel. Be sure to hit "Enter" after typing in a value.

If this window is not visible on your screen, click on one of the sliders to make it reappear.

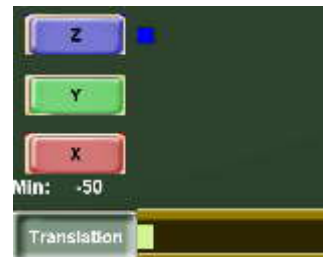
- 9) Other channels can be added and edited to the New action in this same way.

Animating Through Action

Step by step

- 1) Let's add animation to our model through actions now. Add a light component to the model the same way you added the mesh, but this time choose Light under Component -> New.
- 2) In the New action, adjust the Z axis translation so that the light is in front of Abdul's face.
- 3) Now, click on Action at the top and select "New," not to be confused with the specialized New Action that we just went over. This adds a new action which you can name. Name it "Default." "Default" is a special action that plays if the component has no other action to play.
- 4) This action is faster than we want it to be. Click on Action. Go down to "Set Total Time." Clicking on this will bring up a box with a 1. Change that 1 second to 10.

- 5) Click on Key and add the Translation channel. Unlike the 'New' state, this action will change over time, so it can have many keys. You will notice the value slider is not as wide, and there is a draggable key at the bottom of the channel window. This will allow you to add many keys to the timeline and to change their tempo.



If this key is selected and you click on the Key menu at the top to add a channel, 1 the channel will be added to this key. If no key is selected, it will 2 add a new key with the channel you pick. You can also 3 add a channel to this key by right clicking on it and going through that menu.



- 6) Now click on an empty spot just to the right of your new key. From this menu choose "Add Key."
- 7) Change the X value of the second key so that the light moves forward and backward from Abdul. Notice it wraps back to the first key automatically. To turn off this ability, go to the Animation menu, and uncheck "Loop." You can drag the keys to change the timing. If at any time you need to

pause the animation to make it easier to adjust keys, there is a play/pause button in the upper left.

- 8) Now try to do the same thing but with the color channel. See if you can make the light change colors as it travels back and forth. Do this in the Default action. You have the red, green and blue channels as well as the alpha channel. Alpha controls the brightness.
- 9) When done save the model. Go to File -> Save As and name it AbdulWithLight.

Models as Components

Step by step

- 1) You are familiar with adding components to models, as we did in the previous tutorials. Now we will add a model as a component. You can add any existing model as a component of this model framework. In this example, we will use a pre-existing model as our component.
- 2) Add the model, "AbdulWithLight" from the previous tutorial to the scene, either through the Context Menu or through the toolbar. We will be expanding on what we have already created.
- 3) Click on Component -> New -> Model. Find AppleOffset in the list, select it, and click Create. If you want to edit the component model as a model, you can do that by going to Component -> Model properties, however, this will close the parent model that is open now.
- 4) Add a translation key to the new action and move AppleOffset in the Z axis so that it is above the ground. Notice that the apple mesh is not in the center where the gizmo is. This is because it is offset within the AppleOffset model. The gizmo is the actual center of AppleOffset. This is useful because it will allow us to rotate it around Abdul easily.
- 5) Let's add Default animation to this component. We already have defined the Default animation for other components in this model framework, so rather than type it in again, select "Action" -> "Add Sibling's." This will bring up a list of all the actions defined so far in this model. Choose "Default."
- 6) Now that Default is the selected action instead of New, all components are using their Default animation. Notice that this includes the components inside the AppleOffset model, as you can see the apple rotating and scaling.
- 7) To make the apple rotate in a full circle, add three rotation keys to AppleOffset's Default action. Put one key at 0, another at 3.33 and the third at 6.66. Set the Yaw (around the Z axis) to 0, 120 and 240 degrees

- respectively. It will return to 0 between 6.66 and 10. Note that the rotation always takes the shortest path, so if we used only two keys (0 and 180) the results would be unpredictable. Now the apple should rotate in a full circle around Abdul. (Right now you have to 'reset' it by going back to new first.)
- 8) By having the apple in a model, rather than being a direct component, it can continue to perform the actions within its own model system. In this particular example, this includes the scale, rotation, and the translation from its own origin.
 - 9) Save this as AbdulWithApple

Emitters as Components

Step by step

- 1) If you have not read the section on creating particles, you may want to do that now. Remember that it is dangerous to emit cyclical particle systems within a model framework.
- 2) Create a new model either through the Context Menu or toolbar.
- 3) Use Component -> New -> Emitter to add an emitter to the model.
- 4) Add a Default animation through Action -> New.
- 5) Go ahead and add translation keys. Raise it up in the Z axis so that the emitter is above the ground. We want to be able to see the results. With the Default animation selected, use Key -> Translation.
- 6) Let's add a spawn key now. Go to Key -> Spawn. If you hit play, you'll notice that at that point, it emits a default particle.
- 7) To change which particle is emitted, select the spawn key. Inside its key info dialog is a drop down menu. Click on this and you can choose "Model" or "Particle". These are the two things a model can spawn.
- 8) For now, click on particle. This will open a particle selection menu. Find SmokeNonCyclical and click Create.
The particles now have changed to resemble a puff of smoke.
- 9) If you would rather have continuous smoke, simply right click on the key and choose "Copy Key." Paste the key five times and drag them across the time line so they are spread out. The smoke will update automatically, and you can adjust the keys until its appearance is to your liking.
- 10) If you want to adjust the particle system, open the particle editor and load the particle with which you are working. This can be done while the model editor is still open.

Animated Meshes as Components

When given animations, meshes become inherently more complicated. Within the model framework, the mesh's animations are added to the list of actions i actions which you defined in the framework editor. These can be blended together within new actions. Models can be created with mission path following in mind. Also, though, the animations must have their timing calibrated for the model, which we will cover.

Animations as Actions

Step by step

- 1) Create a new model.
- 2) Go through Component -> New -> Animated Mesh and select from the list "Jabbar_Multi_Anim."
- 3) If you look through the actions list, you will notice that many actions have already been included. These are the animations that were originally imported with the animated mesh. Select "conv01."
- 4) The animation conv01 now plays, but it is too fast. Go to the window titled "Boned Animation." Here you will see two fields: Repetitions and Natural Repetitions.
- 5) "Repetitions" is how many times the animation plays within the model's time frame.
- 6) "Natural Repetitions" is how many times the animation would play during this time if it were playing at its natural speed, the speed at which it was originally exported.
- 7) Make sure Natural Repetitions is set to a whole number and hit "Enter."

The Message Channel

Step by step

- 1) Open the model, AbdulWithApple, from the previous tutorial.
- 2) In the component drop down menu, choose the component, AppleOffset. We will first have the model send a message to tell AppleOffset to switch from its Default animation to a different animation through the message system.
- 3) Add a new action through Action -> New. Name it AppleUp.
- 4) In AppleOffset's AppleUp, add a rotation key through Key -> Rotation.

- 5) Change the pitch (green) to -45, and switch back to the Default animation for AppleOffset.
- 6) Add a message channel by going to Key -> Message.
- 7) Drag the message key near end of the timeline. Now we will make the message change AppleOffset's animation from Default to AppleUp.
- 8) In the Key Info window set Message to "AppleUp," Message: Type to "ChangeAnimation," and Message: Target to "None." Check Message: Self on.



- 9) Save the model as AbdulWithApple2, close it, and watch what happens. The light is still playing its Default animation. The apple itself is still playing its Default animation. These did not have AppleOffset animations to change to, so they stay the same. However, AppleOffset now plays its AppleUp animation.

Animation Blending Groups

Step by step

- 1) Open AbdulWithApple2 from the previous tutorial. In this model, we will be working with the component, AppleOffset, which has two actions, Default and AppleUp and blend them together.
- 2) Currently, the Default animation has a message key that tells AppleOffset to switch to AppleUp. Now we will look at how animation blending groups affect how this switch occurs. If at any time you would like to watch the change replay, simply click on the drop down action menu and reselect "Default."

- 3) Notice that once AppleOffset gets to the end of Default, it moves up 45 degrees and no longer rotates around Abdul. It completely switches to AppleUp. Now, with the default action selected, open the Action menu, and go down to "Set Action Blending Group." Notice that Default is set to Group 1. Actions that exist in the same group cannot play at the same time. Actions set in different groups will attempt to blend together.
- 4) Select the AppleUp from the action drop down menu. No go to Action -> Set Action Blending Group. You will see that AppleUp is also set to Group 1. Change it to Group 2. Switch back to "Default" in the action drop down menu, and watch what happens.
- 5) AppleOffset still moves up 45 degrees, but it is also still rotating around Abdul. With each action in two different blending groups, it will blend them together as you see here.

Particle Editor

Particle systems are special effects that can add a lot to the scene while being super efficient. Each system is composed of dozens or hundreds of independent particles. These particles move in 3D space, but they are typically not 3D objects. They are 2D sprites – speedy little images aligned to always face the screen.

Each particle follows its own trajectory through space, and changes size, color and opacity. Its path is predetermined by the parameters that define its laws of motion. In one system, all the particles follow the same laws of motion. One might then expect all the particles to be superimposed on each other. It is by adding random factors that you spread out the trajectories and makes the particle system interesting.

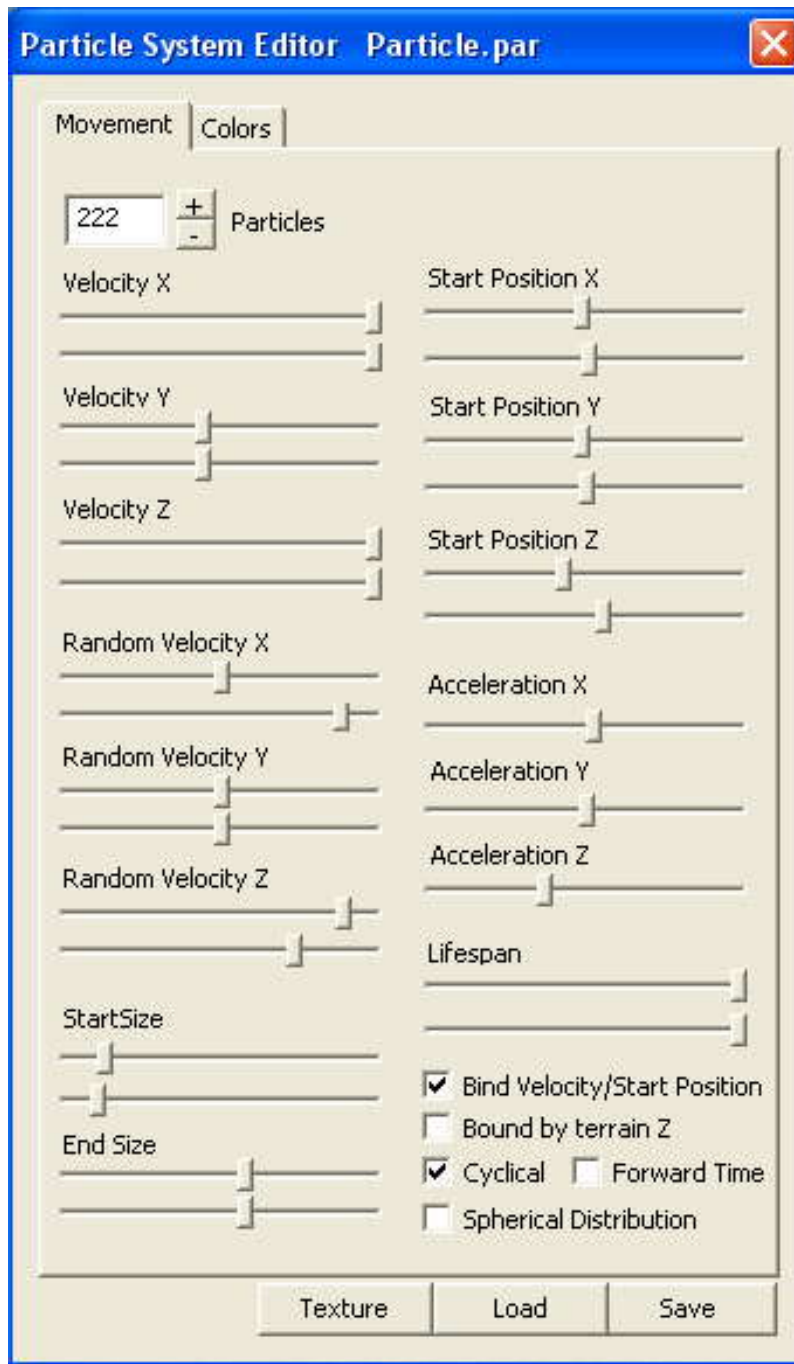
The actual appearance of each particle is determined by several factors. Each has a basic appearance defined by a texture map. This image file defines the color and alpha density of each pixel of the particle.

These colors and opacity are modulated by the particle system itself over the lifetime of the particle according to the particle system's parameters.

The particle blending method is also accessible, making possible special effects.

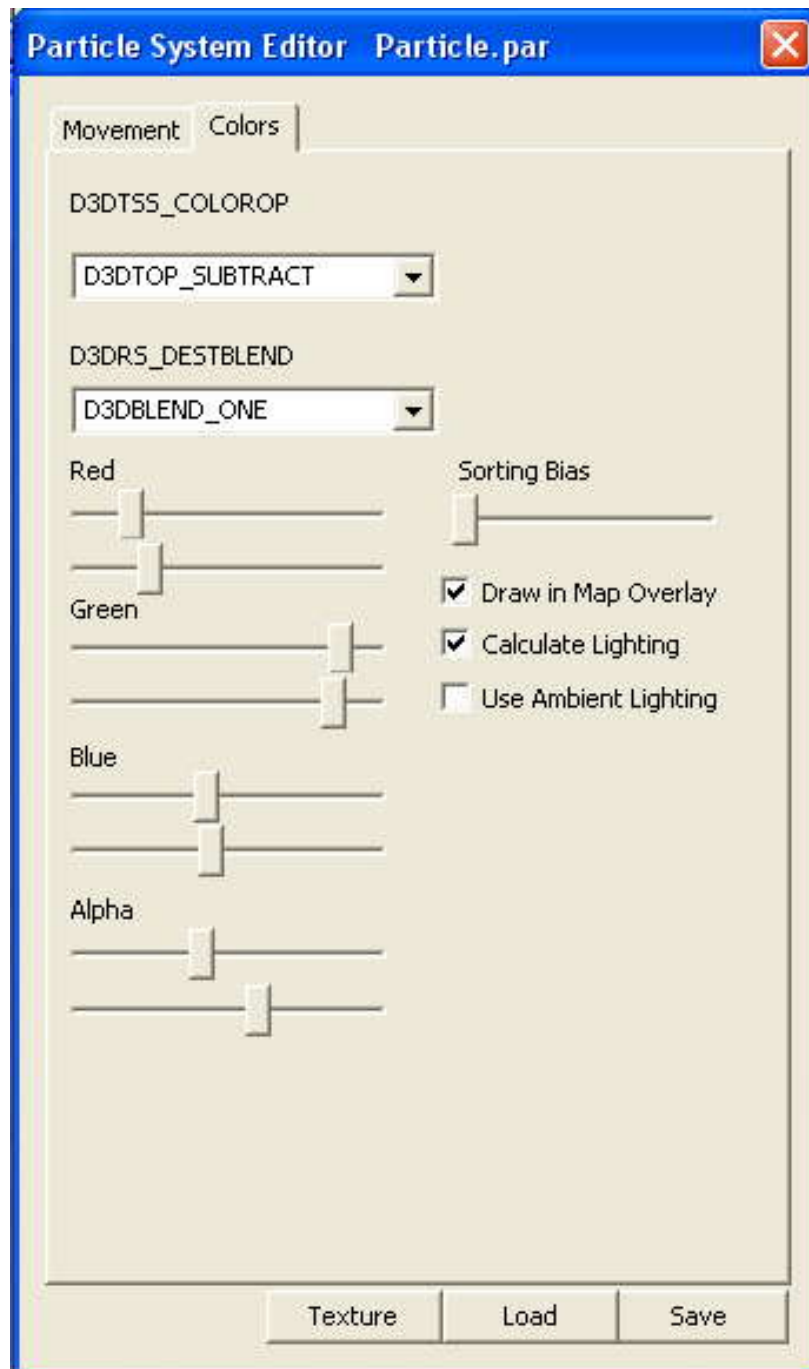


- 1) You can add a particle system either through the Context Menu or through the toolbar editor.
- 2) A particle system will appear on the screen as well as the editor dialog. This dialog has two tabs: Movement and Colors.



- 3) The box at the top allows you to change the total number of particles in the system. Change it to 100, then to 10, then to 1000 and watch the difference.
- 4) Jump down to the lower right part of the window. These checkboxes change the way the system distributes its particles.
 - a) Bind Velocity/Start Position – causes the particle to calculate its velocity based on its spawn point relative to the center of the system.
 - i) Off: The particle spawns at a random point, and then picks a random velocity (speed and direction).
 - ii) On: The particle spawns at a random point, and then picks and moves directly away from the system's center at a speed determined by the distance from center.
 - b) Bound by terrain Z – prevents the particles from penetrating the terrain.
 - c) Cyclical – distinguishes one-shot from continuous effects.
 - i) Off: The system executes only once. When the particles expire, the system expires. Note that during the design process, this editor may replay the effect repeatedly. Do not let this confuse you into thinking the system is cyclical – the looping is just for your design convenience. In the game, the system will execute only once.
 - ii) On: The system is continuous. As particles expire, more are spawned, The system itself never expires, so be very cautious about creating these effects. For example, these are especially dangerous in a model framework. In a framework, an emitter is used to spawn a system, and the system spawns particles. If the systems are cyclical, they will never die, and the repeating emitters will fill the world with particles.
 - d) Forward Time – makes cyclical systems start in the middle of their cycle. Instead of taking time to build up, they begin fully formed.
 - e) Spherical Distribution – modifies the random distribution of the particles so they are evenly distributed around center rather than independently distributed in x, y, and z.
- 5) The rest of Movement tab has sliders to change the particles' behavior. When there are two sliders, it is for min and max values.
- 6) Velocity X, Y, Z – control how much speed each particle has from each axis.
- 7) Random Velocity X, Y, Z – add randomness to the velocity to prevent it the particles from looking identical.

- 8) Start Size and End Size – make the particles grow or shrink over time.
- 9) Start Position X, Y, Z – changes how close to the center of the system the particles start.
- 10) Acceleration X, Y, Z – changes the associated velocity. This makes the particles move more quickly or more slowly in that direction or even reverse directions.
- 11) Lifespan – controls how long the particles last before they expire (and, in cyclical systems, respawn). This determines, among other things, how far they travel.
- 12) At the very bottom of this window are three buttons: Texture, Load, and Save. The texture is an image file (with alpha) that defines the shape of each particle. The “Texture” button will load a different texture map. “Load” and “Save” will respectively load and save particle systems.



- 13) In the colors tab, you can change the colors and rendering of the particles. At the top in the drop down menus, D3DTSS_COLOROP, you control whether the particles blend into the scene using multiplicative or subtractive blending. Use subtractive for dark particles, like smoke.
- 14) Below that in the D3DRS_DESTBLEND drop down menu, choose to invert alpha if you are making dark particles.

- 15) The red, blue, and green sliders control the min and max values for that color channel in the particles. The alpha controls the transparency of the particles.
- 16) The “Sorting Bias” slider determines controls if this particle system will appear in front or behind other particle systems when they share the same world space.
- 17) Use “Show in Map Overlay” to make this system visible on the map.
- 18) Set “Calculate Lighting” and/or “Use Ambient Light” to approximate the daylight effect (try both). Otherwise the particles are emissive, which is not responsive to any lighting conditions.

Property Sheets

Mesh objects, animated meshes, and buildings all have a properties dialog menu where many details of the object can be adjusted. These details include things such as scale and position, lighting and rendering options, and levels of detail. Objects included in the packaging have already had their properties set. Note that if you change any of these, the changes will apply to all instances of that object.

Mesh Properties

A mesh object is a basic 3D object familiar to computer graphics professionals. It describes the surface of an object with a network (mesh) of vertices. Each vertex has a geometric location and can refer to “texture” files. These files, (together with the materials) define the color and other surface characteristics for every point on the model. Mesh objects are imported from an .x file and their properties are defined in an .SMP file (Static Mesh Properties).

Instances of the mesh are placed the scene using either a toolbar or the Context Menu -> Create object menu. These placed instances are saved in the .MOB file (Multiple Objects).

Properties The Tabs...

Appearance:

Under the appearance tab, you can control the scale of the object and how the object is rendered.

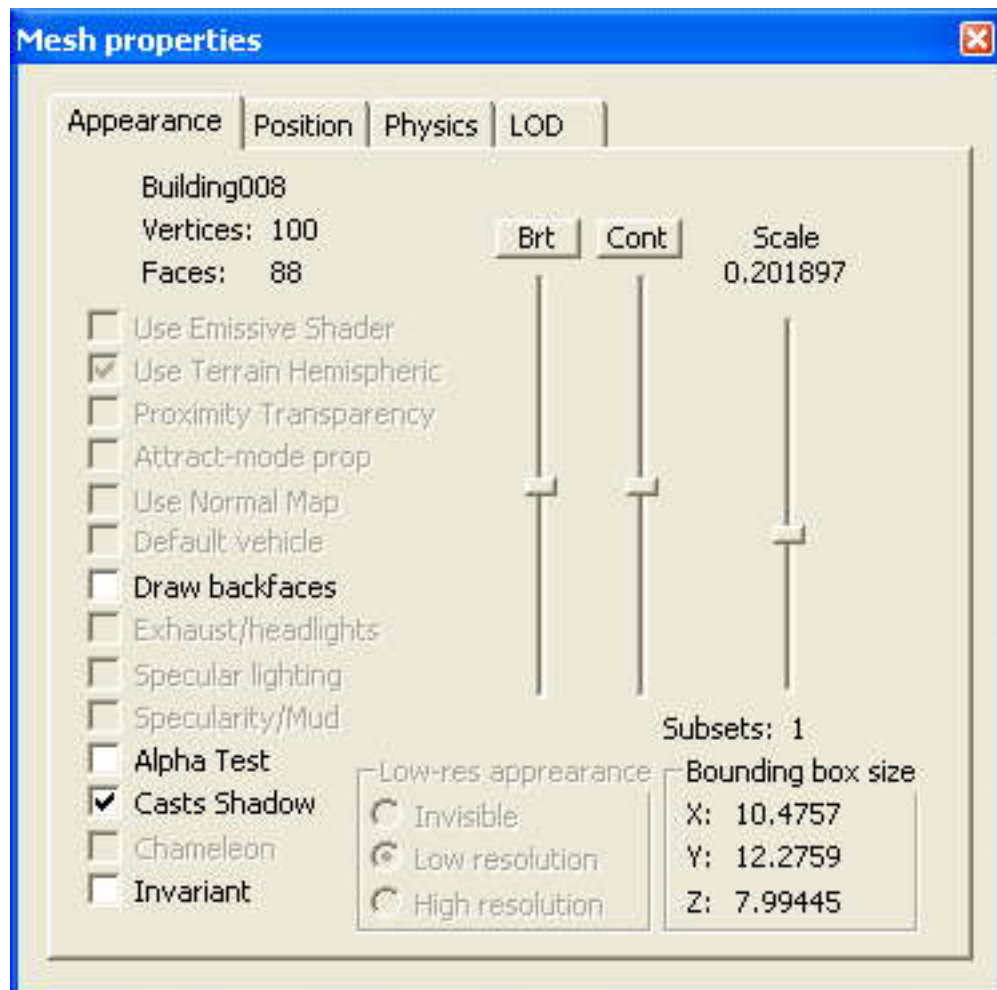
Position: Here you control how the object sits in the world.

Physics: The Physics tab deals with how the object reacts with other objects.

LOD: The LOD, or “Levels of Detail,” tab lets you set up transition meshes that look like the selected mesh but with less detail. The mesh will swap to the lower detail version at a distance that you select. This allows the program to run faster because it is not rendering as much detail without taking away from the visuals.

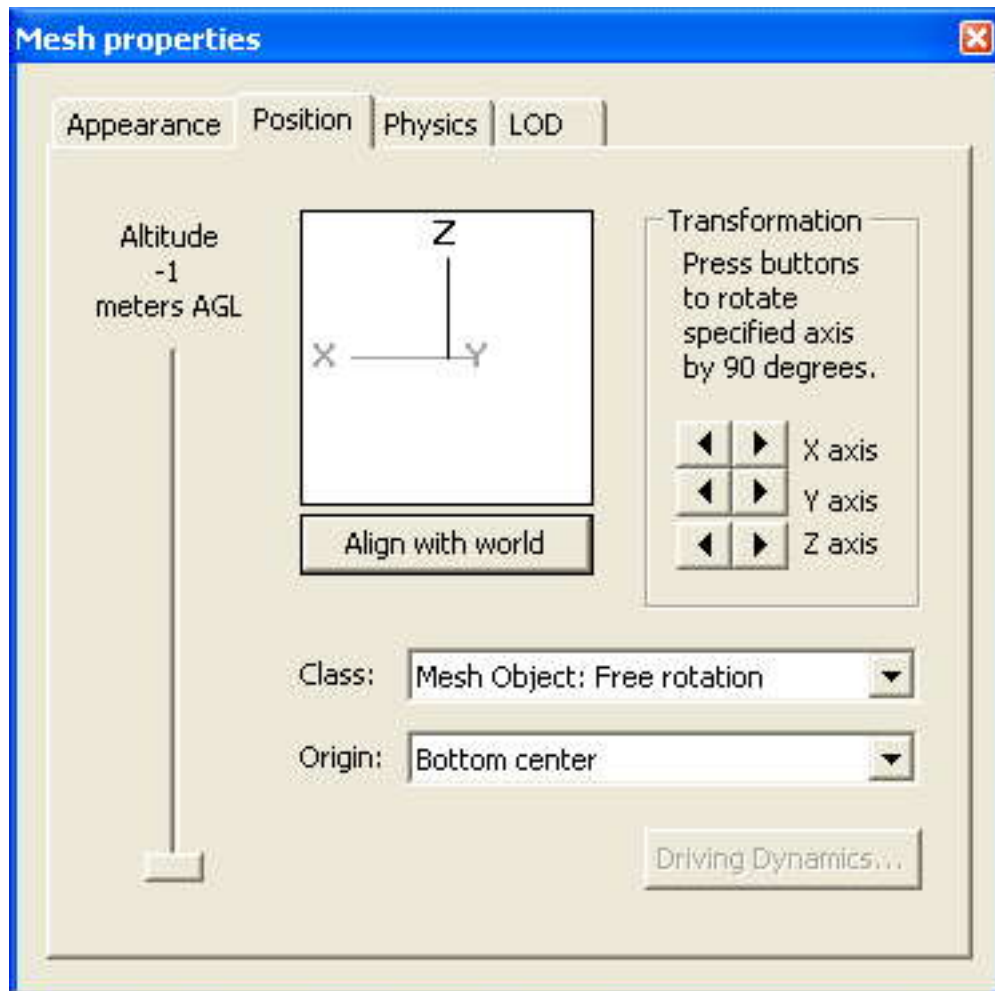
Double click on any mesh object that is not part of a model to open the Mesh Properties window.

Appearance:



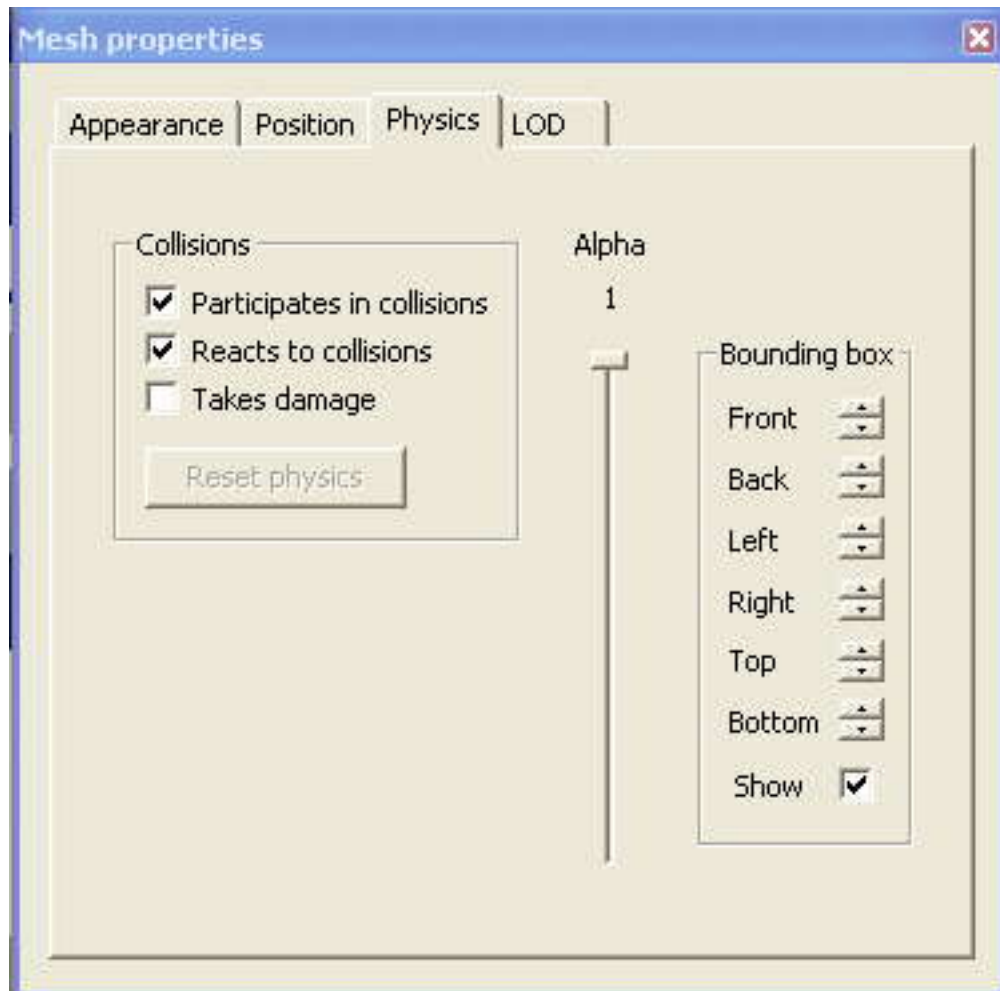
- 1) On the left starting at the top is basic property information about the mesh, such as the name and geometry size. The fewer vertices it has, the more quickly it will be rendered.
- 2) Below that is a list of checkboxes that control certain render properties involving lighting and shaders.
- 3) Low-res appearance at the bottom controls how the object will render in Data Fusion Mode.
- 4) On the right side is a scale slider, which makes the object larger or smaller.

Position:



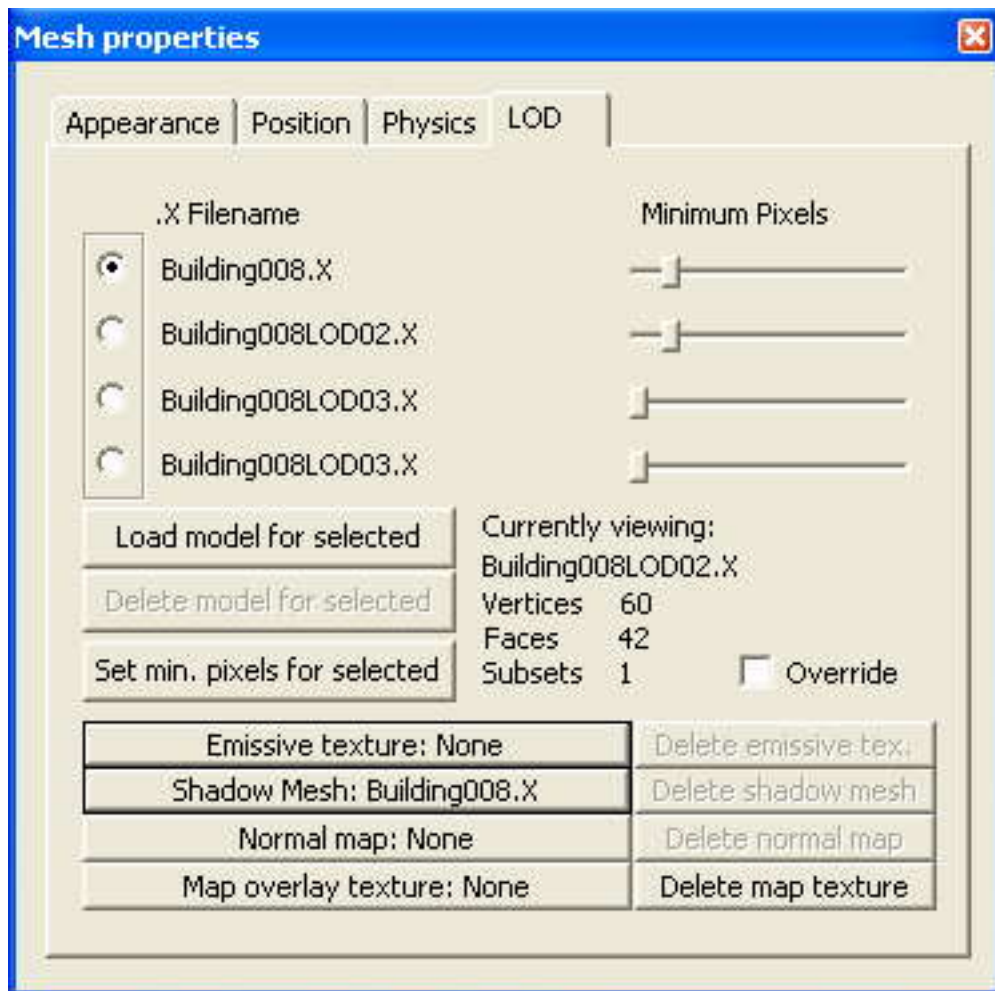
- 1) The “Transformation” section in the top right lets you rotate the object in 90 degree increments. The box in the top middle lets you rotate the object freely.
- 2) In the “Class” drop down menu, you can change an object to be a vehicle or building to allow for special properties that come with these.
- 3) Under the “Origin” drop down menu, you can change where the object’s center is. This will affect the point around which it rotates, as well as where it sits, should the class be set as “Locked to Terrain.”

Physics:



- 1) The "Collisions" box lets you control whether or not the selected mesh collides with objects.
- 2) "Participates in collisions" means other meshes will collide with it.
- 3) "Reacts to collisions" only applies to objects in motion when they collide and will cause them to bounce back after colliding.
- 4) The "Bounding" box lets you establish where the object will collide on each side.

LOD:



- 5) On the top left is the list of LODs. There can only be up to four different LODs. The first one should be your most detailed version, and the last one your least detailed. Use the radial button to select which you would currently like to edit.
- 6) Below this, you will see three buttons.
- 7) "Load model..." will let you load a mesh to an LOD if none exist, or will switch to a different one.
- 8) "Delete model..." will remove the selected file for the selected LOD.
- 9) To use "Set min. pixels for selected," scroll in or out to the point where you would like that mesh to switch to the lower LOD level. When you are happy with the switch point, hit the "Set min. pixels for selected" button. You will see the sliders in the top right adjust according to your actions.
- 10) The Override checkbox on the right lets you view the selected LOD mesh, no matter how close you are to the object.

- 11) The buttons at the bottom allow you to add image maps or alternative geometries that improve rendering. Some of these help increase apparent detail, while others improve efficiency. Note that all but “Map overlay texture” must be flagged under appearance in order to be seen.
- 12) “Emissive textures” allows you to import a texture that defines which areas of the object are emissive. These areas always render in full color and brightness, irrespective of the scene lighting.
- 13) “Shadow Mesh” allows you to import an alternative, simplified mesh which will speed the calculation of shadows..
- 14) A “Normal map” is a specialized texture that describes 3D surface detail.
- 15) The “Map Symbol” (or “Map overlay texture”) is the image that will represent this object in the map overlay.

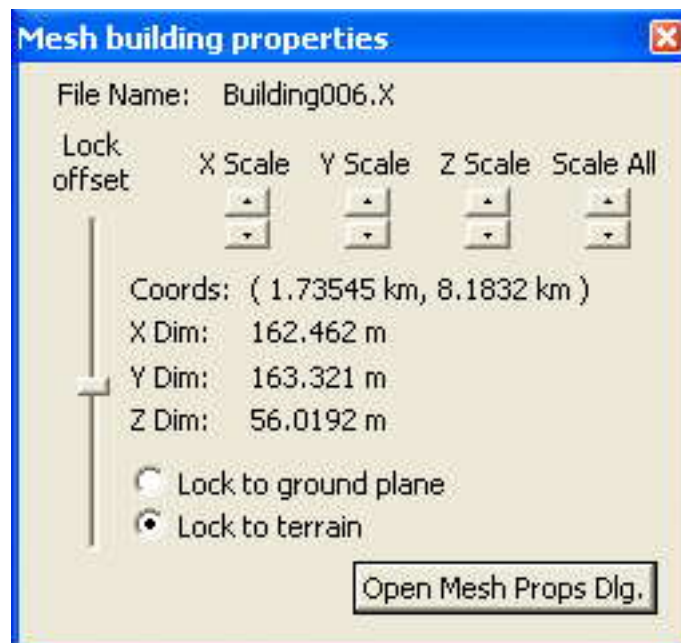
Mesh Building Properties

Buildings are similar to mesh objects but are a special class and have extra properties because of this.

Unlike typical static meshes, they are locked to the terrain automatically and they can be scaled as individual instances.

Step by step

- 1) Double click on any building to open the Mesh Properties window.



- 2) The lock offset on the left controls how high the building's origin will be in relation to either the ground plane or terrain, depending on which radial button is selected at the bottom. "Lock to terrain" will position it relative to the lowest terrain point within its footprint. The offset applies only to this particular instance, and will not affect other buildings.
- 3) The scaling buttons at the top allow you to scale the building uniformly or in one particular axis. Unlike other mesh property settings, these only modify this particular instance, and will not affect other buildings.
- 4) The "Open Mesh Props Dlg." at the bottom will open the general Mesh Properties Dialog that is associated with all instances of this particular static mesh.

Animated Mesh Properties

Using the FrameWork, all objects can animate from one position to another, or animate in size or various other properties.

Animated mesh objects are differentiated from generic (static) meshes by their ability to change shape.

The system supports two different technologies.

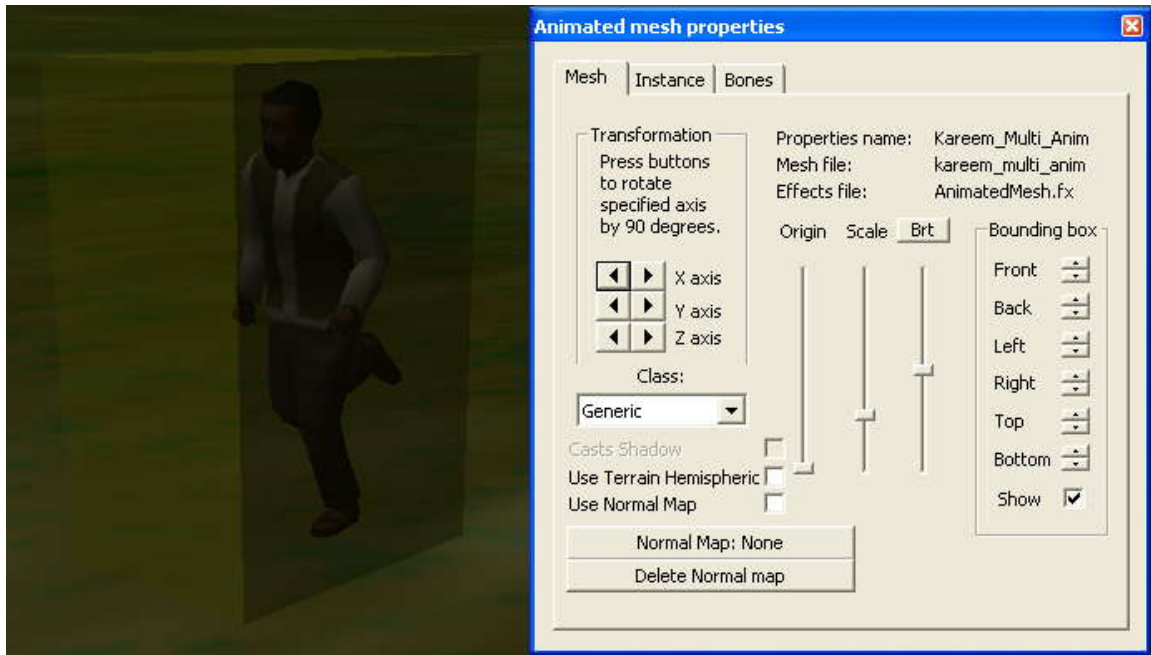
Arbitrary, freeform change might be best handled with "Morph" animation.

Most practical animation requires changing the pose of a human or other figure whose shape maintains integrity even as it deforms.

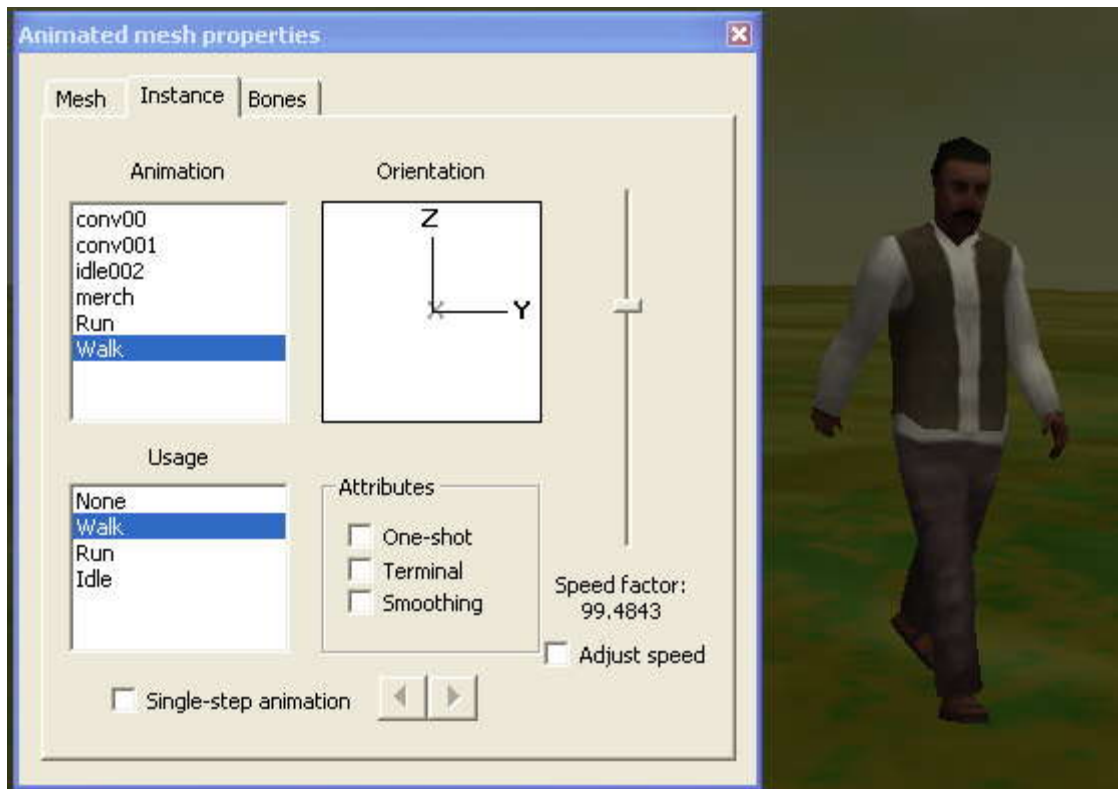
This animation can best be described using an articulated armature within the mesh. This bone system and the animations attached to it characterize the Animated Mesh Object. Although by definition this is the only difference between mesh objects and animated mesh objects, it is a major difference that requires many details to be in place.

Step by step

- 1) Double click on any animated mesh object that is not part of a model to open the Mesh Properties window.
- 2) Mesh



- a) In the top left is Transformation. Rotate the animated mesh in 90 degree increments.
 - b) Below that is Class. Usually “Generic” will suffice. Other classes are designed for special cases.
 - c) At the bottom left are rendering options for lighting.
 - d) Origin and Scale are in the middle. Origin controls how high off the ground the animated mesh sits. Scale controls the size.
 - e) On the right, you can control the bounding box of the mesh. This defines the boundaries where it will collide with objects.
- 3) Instance



- a) Animation: Choose which of the mesh's animations will play
- b) Orientation: Adjust the position in less than 90 degree increments
- 4) Bones
 - a) The Bone list allows you to see the location of any bone within the animated mesh.
 - b) The property list allows you to assign a bone to the selected property for properties dependent on specific locations inside an animated mesh.

Import Meshes

New models can be imported into system and added to your scenario. The engine accepts models in the .x format. Most popular software can create an .x file natively or with an export plug-in. Third party tools exist to convert many other formats to .x or they can be processed through 3D Studio as described below.

File System:

All static meshes and their textures are saved in the /Game/Models directory, and all animated meshes and their textures are saved in the /Game/AnimatedMeshes directory.

Software needed:

- 1) Autodesk 3D Studio Max
- 2) Panda DirectX Export Plug-in
- 3) DirectX Mesh Viewer (for testing and previewing animated meshes)

Meshes

Any file that is not already an .x file can be converted with the Panda exporter as long as it is a file type that 3D Studio Max can read.

Textures

The preferred texture file is a .png. This file must be saved in the same folder where the mesh will be.

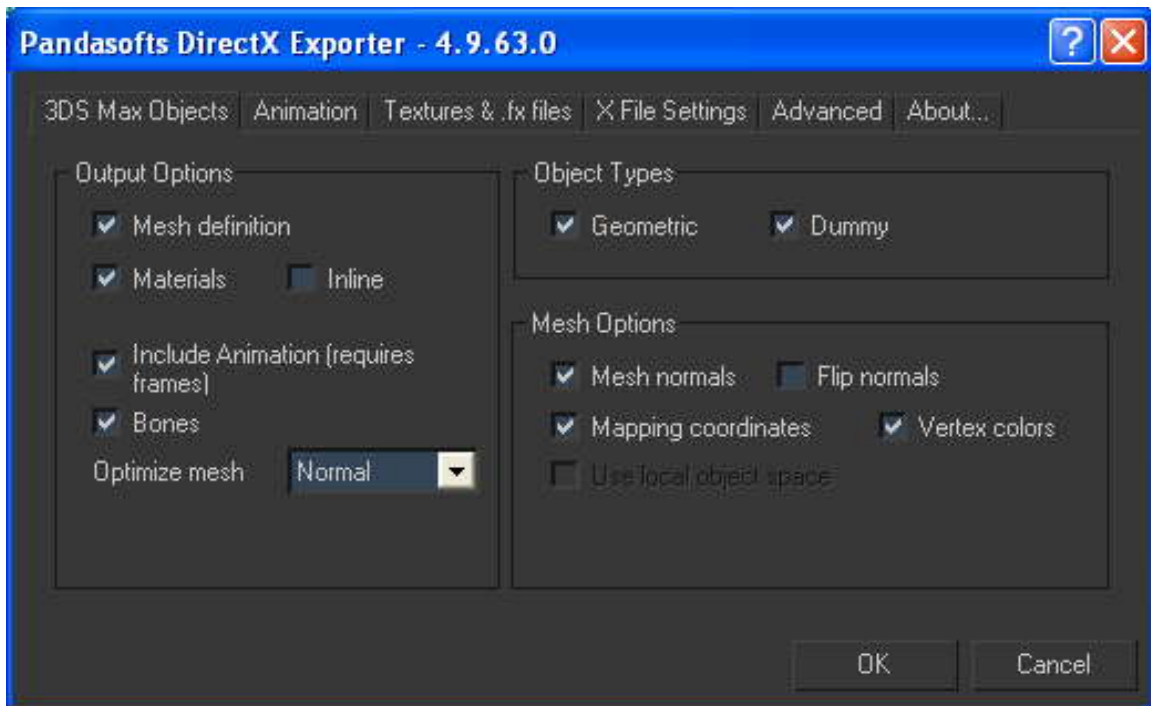
Animations

Animated meshes can present many complications. Please see the animated mesh troubleshooting section at the end if there are issues importing the animated mesh.

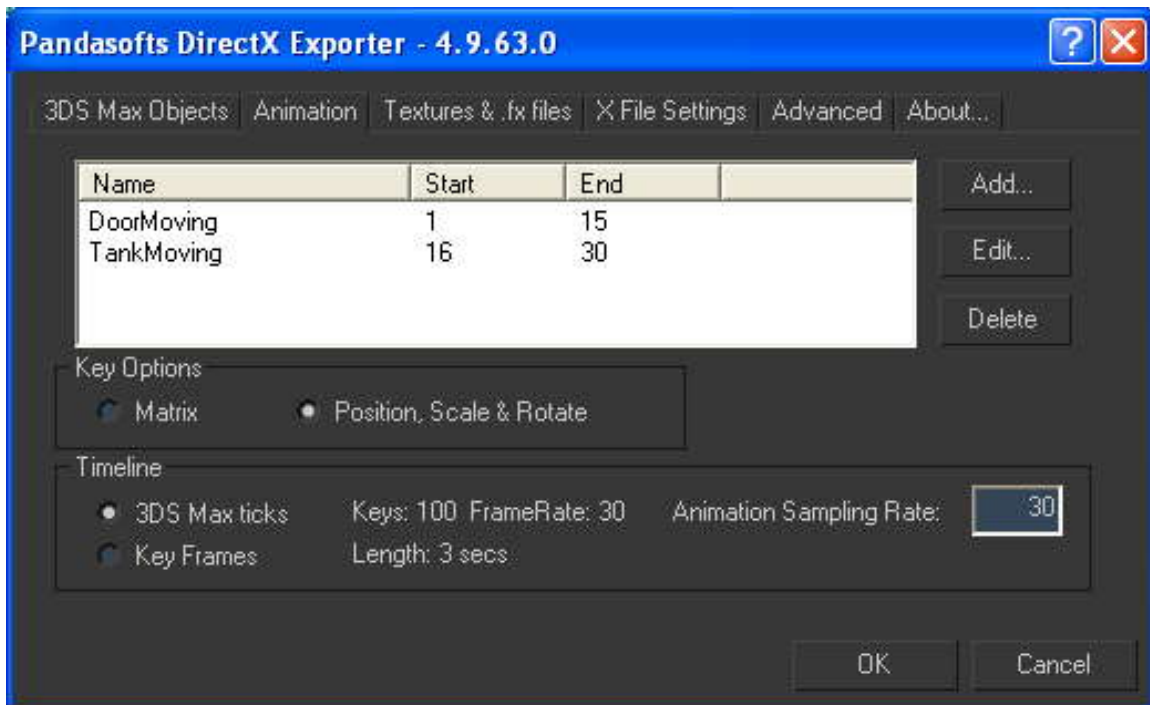
Step by step

- 1) Either open or import the file to be converted to an .x file into 3D Studio Max.
- 2) File -> Export (or Export Selected).

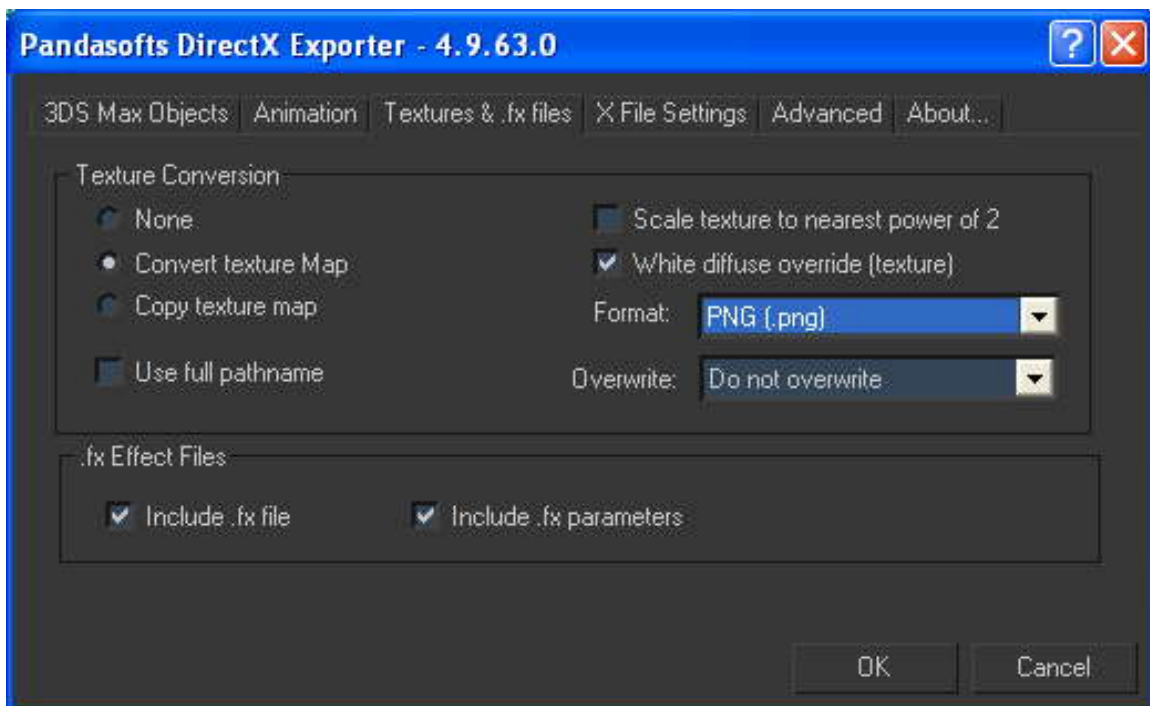
- 3) Browse to the /Game/Models folder if saving a static mesh, or the /Game/AnimatedMeshes folder if saving an animated mesh, and change the Save as type: field to "Panda DirectX (*.X)."
- 4) If exporting an animated mesh, be sure that all meshes are connected to the root bone in the schematic view: Graph Editors -> Saved Schematic View (or New Schematic View if none exist).
- 5) A dialog box will follow.
- 6) 3DS Max Objects - Animation and Bones must be checked if exporting boned animation, but checking them when you do not have animation or bones has no ill effects. If there is animation or bones in the scene and you do not want to include them, uncheck these boxes to avoid unnecessary file size. Everything else should be selected as in the example below.



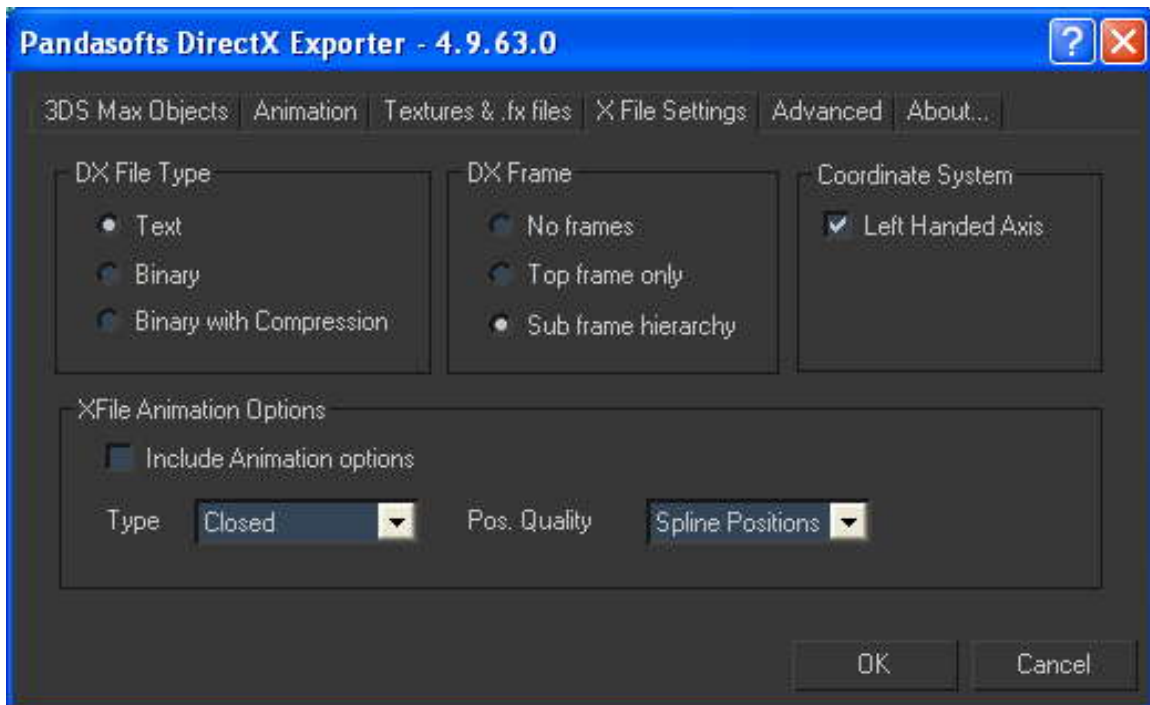
- 7) Animation - The animations listed will show up in the Animation list under the Instance tab in the animated mesh properties dialog. Be sure to use Position, Scale & Rotate, and 3DS Max ticks.



- 8) Textures & .fx files – This will ensure that the texture is saved in the png format, in the correct folder, and that this information is in the .x file itself.



- 9) X File Settings – Be sure to turn on Text, Sub frame hierarchy and Left Handed Axis.



10) Hit OK.

11) Now you should be able to import the mesh like you would add any new mesh, going through either the Context Menu Create menu or through the toolbar. When the Mesh Selection dialog appears, choose “New” to browse and select the new mesh.



Animated Mesh Troubleshooting Guide

Load Error

Game Load Error Message: Frame Count != Animation Count

- Dummy objects being exported as subsets in animated meshes has caused this.

Rogue Vertex

Vertices do not animate correctly with the bones they are attached to, go off to strange spots in space, don't move at all when they should be.

- Make sure all polygons have shading groups.
- Make sure that there are no bones scaled. Select all bones. Animation -> Bone Tools -> Reset Stretch, Reset Scale
- Make sure there are no isolated vertices.
- Make sure no bones were mirrored when creating the rig [not positive on this].

Skewed Subset

Subsets are rotated or skewed from their apparent position in max in comparison to the other subsets in the scene.

- Try connecting the broken subset to a working one in the hierarchy.

Mesh Viewer Failure

Error when importing into Microsoft's Direct X Mesh Viewer

- Make sure there are no bones sharing the same name.
- Make sure you did not change bone names after setting up IK chains.
- Make sure IK chains it expects to find are being exported.

Dummy / Mesh Misalignment

Dummy Objects and Mesh Object are not aligned

If the mesh is a child of a bone system, and the dummy objects are not, they will not transform in the same way

Guide for:

Technicians

Help System

Players can access the Help system at any time. There are three ways to reach it. The toolbar has an In Game Help button. Players can use the “Escape” key to bring up the Main Menu, and through this, access the Help Menu. Or they can simply press “F1”, the universal help shortcut key.



The buttons across the top represent major topics and are used for easy navigation. The two windows in the middle contain the Help content. The left window contains text and the right illustrates the material with an image. The arrow buttons on the bottom navigate pages within the selected topic.

Editing the Help Screen XML

The contents and appearance of Help system is defined in a single XML file. The scenario defines which help file to use. Typically you will use the XML help file, `AvantGuard.hlp`, found in the `/Game/HelpScreens` directory.

Step by step

- 1) HelpSystem – Control the page’s overall layout through the following:
(Be sure that all interface image files are saved in `/Game/GUITextures`.)

```
<HelpSystem Title="Avant Guard Help" Image="GUIHelpScreenAGBackground.png"
LeftBorderTexture="GUIBorderNeutral.png" LeftBackgroundTexture="GUIDialogBackground.png"
RightBorderTexture="GUIBorderNeutral.png"
PreviousButtonTexture="GUIButtonAGArrowLeft.png"
NextButtonTexture="GUIButtonAGArrowRight.png" CloseButtonTexture="GUIButtonClose.png">
```


- a) Title – Title text that appears on the top border
 - b) Image – Background image;
 - c) LeftBorderTexture – Border on the middle left window;
 - d) LeftBackgroundTexture – Image appearing behind the text in the middle left window;
 - e) RightBorderTexture – Border on the middle right window
 - f) PreviousButtonTexture – Bottom left button that takes the user to the previous screen
 - g) NextButtonTexture – Bottom right button that takes the user to the following screen;
 - h) CloseButtonTexture – Button in upper right that closes the menu
- 2) Topics and Content – Next in the XML are topics and their content starting with the right most topic button and going left. Note that the Image files named as part of a Page are found in the /Game/Preview directory.

- a) Topic – Starts a new major topic

```
<Topic Title="Avoid" Button="GUIButtonHelpAGAvoid.png">
```

- i) Title – The title that appears when mousing over the topic button
- ii) Button – The png file that defines the button states that appear in the top section. Clicking on this button opens the corresponding topic

- b) Page – Starts a new page within the topic

```
<Page Title="Assess Threat Levels">[NEWLINE] Threat levels on streets show the possibility of a threat on that street. [NEWLINE] [NEWLINE] Green - Safe [NEWLINE] [NEWLINE] Blue - Unknown [NEWLINE] [NEWLINE] Yellow - One known threat [NEWLINE] [NEWLINE] Orange - Two known threats [NEWLINE] [NEWLINE] Red - Three or more known threats
```

- i) Title – The title that appears above the middle windows
- ii) Text – The text that will appear in the left screen. Use [NEWLINE] to start a new line. There must be a space before and after [NEWLINE].
- c) View – Controls the right panel associated with the <Page/> in which it's included.

```
<View Image="AGAssessStreets.png" Sound="Filename" CameraPosition="0,0,0"/>
</Page>
```

- i) Image – The image that will show in the right screen,
- ii) Sound – Plays this sound when the page is selected

- iii) CameraPosition – Shows the world at this camera position. View Image must not be in use for this to work.

Load Screens

Images appear on the screen while the game is loading, to provide information, and make load time more interesting. Images will be as large as the game window.

Step by step

- 1) Save the image that you would like to use as a load screen in the /Game/Preview directory.
- 2) Load files are specified in a LoadingScreens node in the .scn file.
 - a) This node contains a series of LoadingScreen nodes
 - b) In this example, a sequence of two loading files is defined:

```
<LoadingScreens>  
  <LoadingScreen File="AVGload00.png" WaitTime="4.0" TransitionTime="0.5"/>  
  <LoadingScreen File="AVGload01.png" WaitTime="4.0" TransitionTime="1.0"/>  
</LoadingScreens>
```

- 3) The LoadingScreen node has three attributes
 - a) Actual game load time depends on the size of the specific .scn file as well as the system loading it. It may last longer than the LoadingScreens sequence. In that case, the last LoadingScreen will transition back to the first File – the name of the graphics file to be displayed while the game is loaded
 - b) WaitTime – says how long (in seconds) that image will display before transitioning to the next file
 - c) TransitionTime – the time it takes to blend away LoadingScreen and repeat.

File Types

These are the file types unique to the Works engine and how they are used within the structure.

Individual File Types

AUT – autonomy settings; /Game/
AMP – animated mesh properties; /Game/ Animated Meshes/
APF – aircraft profiles; /Game/
CFG – speech configuration file; /Game/Speech/
CSV – data results; /Game/
CTL – user controls; /Game/
CVY – convoy manifests; /Game/Convoy/
DMP – crash dump file; /Game/
DOC – documentation; /Game/Documentation/
EVT – event playback aircraft; /Game/
FXO – shaders; /Game/CompiledShaders/
GUP – GUI properties; /Game/GUIProperties/
HLP – help screen templates; /Game/HelpScreens/
LOG – console saved files; /Game/Logs/
MDL – models; /Game/Models/
MIS – contains waypoint information for a mission path; /Game/
MOB – scene file containing saved objects; /Game/
/Game/Autosave/
PAR – particle effects; /Game/Particle/
PNG – texture file; found many places
REG – registry files; /Game/Utilities/
RTF – documentation; /Game/Documentation/
SEM – terrain geometry file; /Game/Terrain/
SMP – static mesh properties; /Game/Models/

TBR – contains saved toolbar settings; /Game/Toolbars/

TGA – texture file; found many places

TXT – documentation; /Game/

/Game/Documentation

WAV – sound file; /Game/Sounds/

X – animated and static mesh geometry, rigs and animation;

/Game/AnimatedMeshes/

/Game/Models/

Composite File Types

TLP – contains several types of terrain files: SEM, terrain and hemispheric lighting textures

SCN – contains several files regarding the behavior of the level: MOB, TBR, TLP, MIS, AUT, APF, WAV, CVY, TLP, CTL

contains properties regarding the behavior of the level: options, scripted events

Scenario (SCN) Options

Edit Functions

Edit

FALSE

TRUE

Allows the scenario to be edited (and other super-user functions)

AutoSaveMOB

FALSE

TRUE

Autosave your work during scenario editing

BuildAndShowGlowMeshes

FALSE

TRUE

Visualization technique for Selected Object (incompletely supported)

DragFormations

n/a

TRUE

UAVs can be selected and moved

DragConvoys

n/a

TRUE

Convoys can be selected and moved

DragCASMissions

n/a

TRUE

Close Air Support Missions can be selected and moved

DragThreatClouds

n/a

TRUE

can be selected and moved

EditPhotoOps

n/a

TRUE

Points of Interest (for UAV flightpaths) can be selected and moved

EditTriggers

n/a

TRUE

Model's invisible but responsive geometry can be seen, selected and moved

Game Behavior

ShowStats

FALSE

TRUE

After each session (Game Over) open the console and display metrics

RequireNetworking

FALSE

FALSE

AvantGuard as (exactly) two-player game

Control System

DefaultJoystickControl

FALSE

FALSE

Expect Joystick (Game Controller) rather than Keyboard and Mouse

UseVoiceRecognition

FALSE

FALSE

Not currently supported in AvantGuard

Interface

UseGUIMenus

TRUE

TRUE

Use custom interface components (when available) instead of OS (Teeded for fullscreen)

ShowAvantGuardMenus

TRUE

TRUE

Does the game present the AVANTGUARD interface?

ShowWheelchairHUD

FALSE

FALSE

Does the game present the Pete Armstrong interface?

ShowGasHogsMenus

FALSE

FALSE

Does the game present the Gas Hogs interface?

ShowCompass

TRUE

TRUE

HUD includes Compass

ShowFuelGauge

FALSE

FALSE

n/a in AvantGuard

ShowSpeedometer

FALSE

FALSE

n/a in AvantGuard

ShowFinderArrow

FALSE

FALSE

n/a in AvantGuard

ShowMapOverlay

TRUE

TRUE

Presents a mini map of the scenario

Visibility

ShowMissionSplines

FALSE

TRUE

Does the player see the flightpaths in Map and 3D?

ShowWaypoints

FALSE

TRUE

UAV Waypoints are visible in Map and 3D

Aesthetics

ShowLensFlare

TRUE

TRUE

Simulate cinematic effects of sun

ShowShadowMapping

FALSE

FALSE

Render realistic scenes using shadows (Not yet implemented for UAV Cameras)

ShowShadowVolumes

FALSE

FALSE

Alternative shadow technology