InterWave Studios

Stargate: The Last Stand Game
Source Development Documentation

Stargate: The Last Stand v1.1

Game Details

TITLE: Stargate: The Last Stand Beta

GENRE: Action/Sci-fi

DEVELOPER: InterWave Studios

PUBLISHER: InterWave Games

RELEASE DATE: Jan 18, 2009

LANGUAGES: English*, Goa'uld*

*languages with full audio support

Multi-player

Captions available

Stats

HDR available

Valve Anti-Cheat enabled

Controller enabled

System Requirements

***Product requires a minimum of one Source™ engine game to be installed on a users system, in conjunction with the Steam™ distribution platform. Details follow.*

MINIMAL: 1.7GHz Processor, 512MB RAM, DirectX® 8 level Graphics Card, Windows® Vista/XP/2000, Mouse, Keyboard, Internet Connection

RECOMMENDED: 3.0GHz Processor, 1GB RAM, DirectX® 9 level Graphics Card, Windows® Vista/XP/2000, Mouse, Keyboard, Internet Connection

This is a blank page.

SDK Setup Overview

STEP ONE: Start the SourceSDK application.

STEP TWO: Set the engine version to 'The Orange Box'.

STEP THREE: Open the hammer editor.

STEP FOUR: Visit the 'tools' tab, then 'options', and in the
game configurations tab click the 'edit'
button at the top (right of configuration).

STEP FIVE: We need to give the product a name to start,
so click 'add' and write "SGLTS".

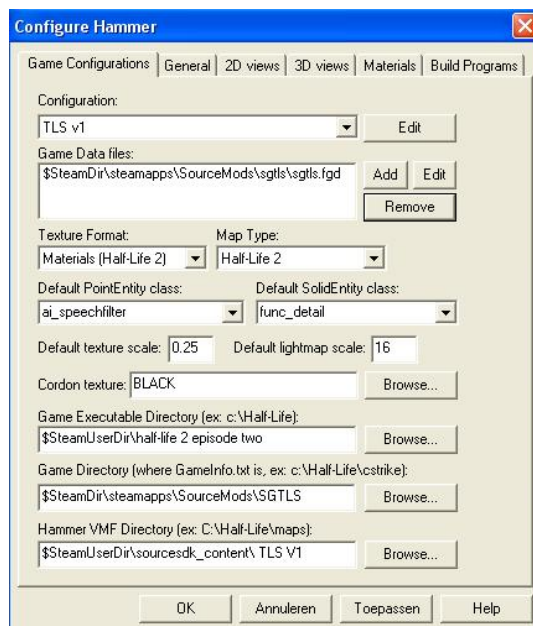**STEP SIX:**   The rest of the settings should match the
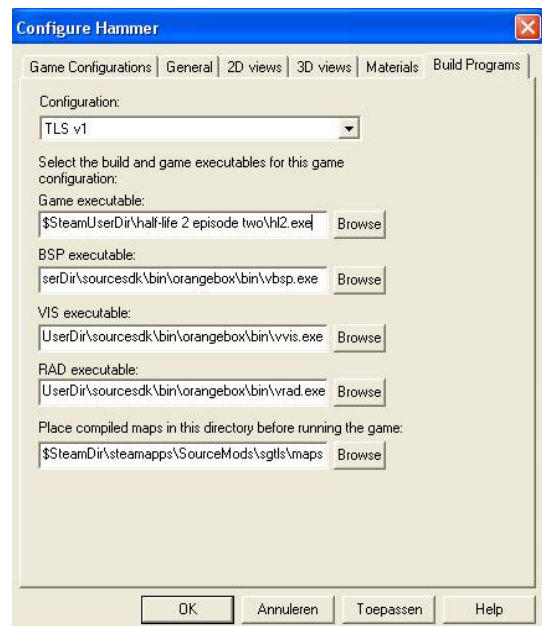following images:

Image One

Image Two

STEP SEVEN: Your level **MUST** contain the following entities. Follow these steps carefully. Add a
'point_viewcontrol' entity, which will control the camera view on level load. Just
leave all of the settings at default, and name the entity "spawn_camera".

STEP EIGHT: Spawn points. The Stargate game uses the entity "info_player_teamspawn". You require at least a single spawn for each team, which can be changed in the properties of the entity. Please note;- it is recommended to have 16 spawn points for each spawning area.

STEP NINE: Game Rules. This entity is used to set up the entire game play structure, and will require many changes to balance your level. Name the entity "gamerules" and set the properties according to your needs.
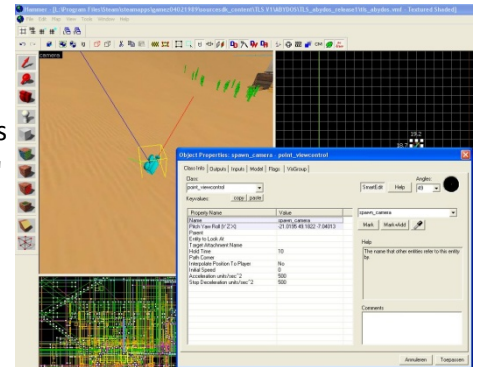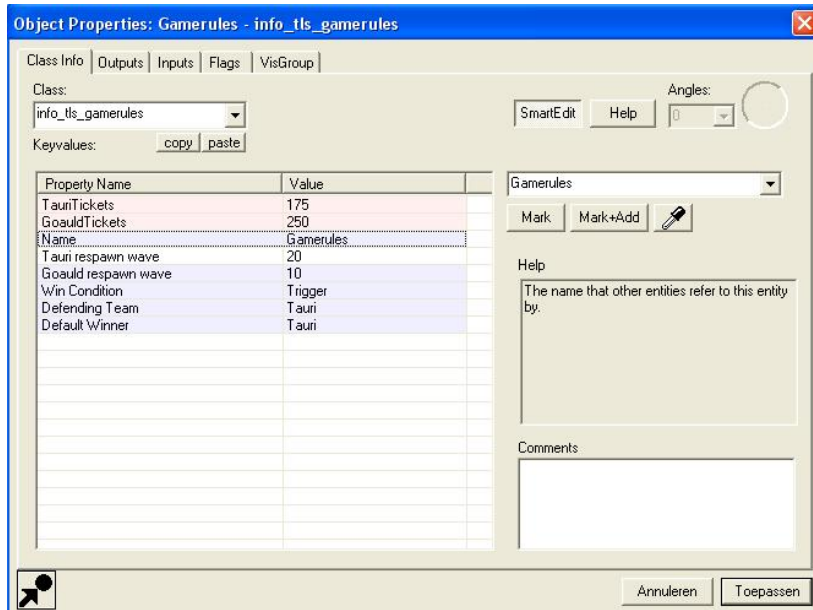
Objective / Game-play Overview
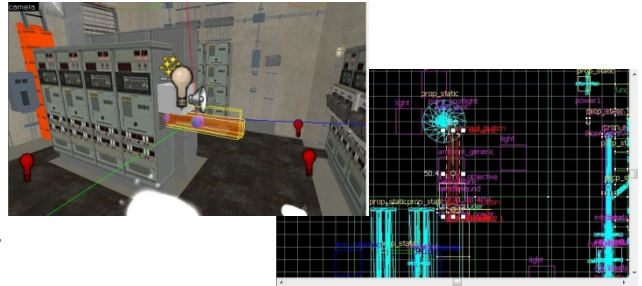
info_mapsize_limit:

PURPOSE: Triggers outputs, depending on how many players the server can hold (maxplayers).

STRUCTURE: OUTPUT: MaxPlayers_8: Sent when there should be no more than 8 players.

OUTPUT: MaxPlayers_16: Sent when there should be no more than 16 players.

OUTPUT: MaxPlayers_32: Sent when there should be no more than 32 players.

NOTES: The outputs will get called once on mapload. The entity will be removed once it has fired the output.

info_player_teamspawn:

PURPOSE: Only spawns players of the same team set in the entity.

STRUCTURE: KEYFIELD: INT - StartDisabled: Should the entity start disabled.

INPUT: Enable: Enables the spawn point.

INPUT: Disable: Disables the spawn point.

OUTPUT: OnPlayerSpawnaxPlayers: Fires when a player spawns on this point.

info_tls_gamerules:

PURPOSE: Entity that sets up the game play logic.

STRUCTURE: KEYFIELD: INT - TauriSpawnTime: Time it takes for Tau'ri spawn windows to pass.

KEYFIELD: INT - GoauldSpawnTime: Time it takes for Goa'uld spawn windows to pass.

KEYFIELD: INT - TauriTickets: Initial Tau'ri team tickets.

KEYFIELD: INT - GoauldTickets: Initial Goa'uld team tickets.

KEYFIELD: CHOICE - WinCondition: The game win conditions.

total_control: Must hold all objectives to win.

trigger: As soon as a specified trigger is activated, a team wins.

KEYFIELD: CHOICE - DefendingTeam: Sets the team that is defending the objectives.

tauri: The Tau'ri must defend, the Goa'uld must attack.

goauld: The Goa'uld must defend, the Tau'ri must attack.

INPUT: TriggerTauriWin: Causes the Tau'ri team to win the round.

INPUT: TriggerGoauldWin: Causes the Goa'uld team to win the round.

info_tls_objective:

PURPOSE: Manages the objective entities.

STRUCTURE: KEYFIELD: CHOICE - TeamFilter: Who can trigger this objective.

tauri: The Tau'ri can trigger this objective.

goauld: The Goa'uld can trigger this objective.

both: Both teams can trigger this objective.

KEYFIELD: CHOICE - StartDisabled: Start so that neither team can trigger.

true: True, neither team can trigger.

false: False, it can be triggered.

INPUT: Trigger: Trigger this objective.

INPUT: Lock: Lock this objective from both teams.

INPUT: Unlock: Unlock this objective.

OUTPUT: OnTauriTriggered: Fires when a Tau'ri triggers the objective.

OUTPUT: OnGoauldTriggered: Fires when a Goa'uld triggers the objective.

func_hold_button:

PURPOSE: A button that needs to be held down before it will fire.

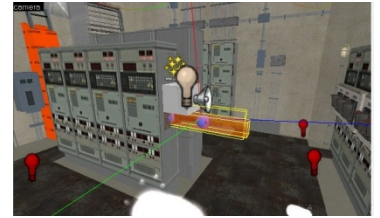STRUCTURE: KEYFIELD: CHOICE - TeamFilter: Who can press this button.

tauri: The Tau'ri can press this button.

goauld: The Goa'uld can press this button.

both: Both teams can press this button.

KEYFIELD: INT - HoldTime: How long the button needs to be held before it will fire OnPressed().

- All inputs, outputs, flags, and fields from the regular func_button will work with this entity.
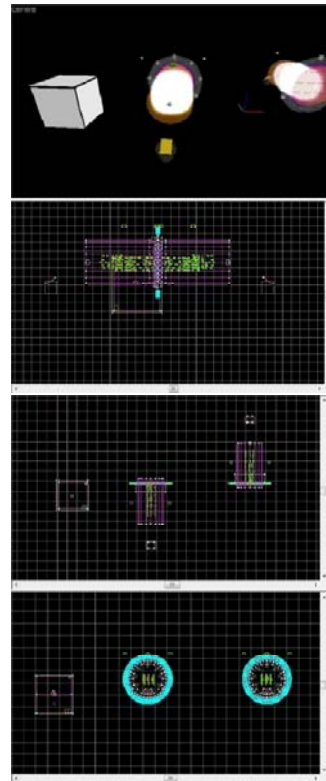
Basic Stargate Prefab Setup

STEP ONE: Copy and paste the prefab into your level, one set at a time.

STEP TWO: Move the Stargate and its DHD to the desired location, one at a time for each set.

NOTE: If you need to rotate the Stargate at all, you must rotate both Stargates by the same amount, so that they are both facing in opposite directions for the teleport system.
You may however, rotate the DHD's as freely as you wish.

STEP THREE: There is a clipping brush in front of each Stargate, which is to be used as a level for the height at which the players feet will teleport to.   *- If this is too low, players may get stuck in the ground!*

STEP FOUR: The black and white textured box should be hidden somewhere outside of the level, where players cannot see it.

Basic Rings Prefab Setup

STEP ONE: Copy and paste the prefab into your level, one set at a time.
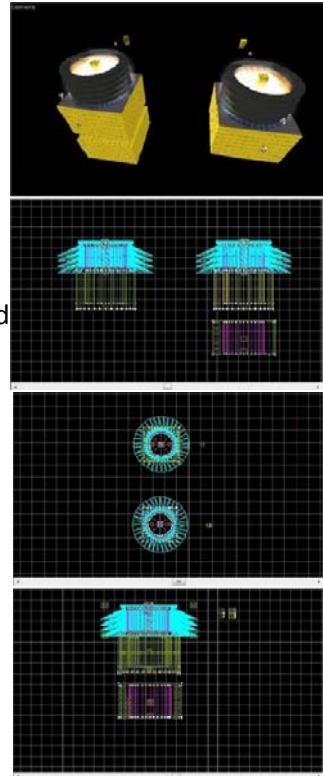
STEP TWO: Move the Rings and their Button to the desired location, one at a time for each set.

NOTE: *Do not attempt to turn the rings upside down, ungroup any part of the prefab, or add the same prefab to the same level twice.*

STEP THREE: Align the rings to the floor, and change the material of the exposed brushes to match your environment.

STEP FOUR: The black and white textured box should be hidden somewhere outside of the level, where players cannot see or access it.

STEP FIVE: Once in your level, you should right click on the rings prefab and choose the option "Centre Origins". This is not always necessary, but to eliminate errors, it is more prudent to do so.