



**CONFUS-ED**

**BY RAVEN SOFTWARE**

## 1.0 What Is ConfusEd?

ConfusEd is the editor tool that allows you to create and edit the properties of almost every entity in the SOF2 single player game. The entities created with this tool are, oddly enough, referred to as “ConfusEd entities”. Examples of ConfusEd entities are office chairs, bottles, photocopiers, and toilets.

### 1.1 What Are ConfusEd Entities Good For?

ConfusEd entities provide the user with a huge potential for interaction. You can create entities that will make a noise when the player tries to “use” them, spin when the player shoots them, explode when they are destroyed, etc.

## 2.0 Using ConfusEd

To get started you’ll probably want to open an existing ConfusEd entity file. In the “base” folder where the SOF2 game is installed you’ll find “therest.pk3” – open this like it’s a .zip file. Within it you’ll find “equipment.ent” – extract this file and open it in ConfusEd. What you’ll see is a list of all of the entities defined within equipment.ent.

## 3.0 Events

Click on any entity listed in ConfusEd and you’ll see a list of all possible “events” that the entity can receive. Here are all of the possible events and a brief description of what each one means.

- **Blocked** -- unused
- **Die** – occurs when the entity’s health reaches zero
- **Init** – occurs when the entity is created while running the game
- **Pain** – occurs when the entity takes any damage
- **PIUse** – occurs when the player “uses” the entity
- **Reached** -- unused
- **Think** – occurs twenty times per second
- **Touch** – occurs when something touches the entity
- **Use** – occurs when the entity is “used” by a different entity

## 4.0 Hooks

If you right-click on an event, you’ll be presented with the “Add Hook” option. Hooks are chunks of game code that get called (or “executed”) when a certain event occurs. If you choose to add a hook to an event you are presented with a list of available hooks...from “Activate” to “Was Surface Hit”. When a hook is executed it returns either a True or False value so that following hooks might or not might be executed depending on the results of the previous hook. Here’s the entire list of hooks and a description of what happens when the hook is executed.

- **Activate** – entity will receive events now (might have been deactivated earlier)

- **Add Joint** – defines the connection between two bones as a joint so that part of the entity's model can be rotated
- **Deactivate** – entity no longer receives events (won't register pain, etc.)
- **Disable Standard Pain FX** – standard behavior is for an impact effect to be played at the point of impact based on the material hit. This hook disables that feature.
- **Enable Standard Pain FX** – This hook enables the above feature in case it was turned off
- **Health Check** – returns True or False based on the current health of the entity
- **Jump** – moves the entity a random amount
- **Kill Entity** – forces a Die event to be sent to the entity
- **Play Animation** – plays an animation on the entity's model
- **Play Effect** – plays an effect either at the entity's origin or on a bone
- **Play Sound** – plays a sound centered on the entity
- **Random** -- randomly returns True or False based on a percentage supplied by the user
- **Remove Entity** – removes the entity from the map
- **Rotate** – rotate one of the entity's bones
- **Run Max Times** – executes children with True until it's been called a certain number of times, then it executes children with False any subsequent time it's executed
- **Run Script** – plays a script
- **Surface Damage** – returns True if a surface has taken a certain amount of damage
- **Timer** – executes children with True after a certain amount of time
- **Toggle Bullet Invulnerability** – entity becomes invulnerable to bullets
- **Toggle Surface** – turn on/off one of the entity's surfaces
- **Toggle Total Invulnerability** – entity can't be damaged
- **Was Surface Hit** – only called from Pain event, executes children with True if the pain impact occurred at the prescribed surface

## ***5.0 Hook Execution***

When an entity receives a certain event, all of the hooks listed immediately under that event will get called. Any hooks (we'll call them child hooks for this example) listed immediately under those hooks (we'll call them parent hooks) will get called depending on whether the parent returned a True or False and whether the child is defined as Execute When Parent Succeeds (meaning True), Execute When Parent Fails (meaning False), or Execute Always (regardless of whether parent was True or False). There are a number of examples in the various entities defined in the .ent files in the therest.pk3 file.

## ***6.0 Tips Regarding Bones and Surfaces***

Some of the hooks allow you to specify bones or surfaces on a model. A surface is one or more of the polygons that comprise the model and a bone is just a point inside the model that controls how some of the surfaces move when they animate. All of the bones and surfaces for a given model are listed, and can be viewed with the SOF2 ModView tool.