
Subject: big new engine feature added to scripts 2.1, Keyboard Hooks

Posted by [jonwil](#) on Wed, 26 Jan 2005 08:57:06 GMT

[View Forum Message](#) <> [Reply to Message](#)

I have jsut completed adding the new keyboard hooks feature to scripts.dll 2.1 as well as the first scripts that will use it (the Jetpack scripts)

How it works is this:

Firstly, there is code that runs on the client.

At startup, the client reads a file called keys.cfg from the data folder (the name and/or location may change).

This file contains lines of the form Jetpack=Y_Key or Deploy=U_Key

The first part can be any string you like and is the "logical key" name which is used later when the script registers the hook.

The second part is one of the standard renegade key names (a full guide/list and possibly a GUI application or custom renegade dialog for editing the config file will be included with 2.1 when its released).

The code takes each line and adds it to a data structure containing key entries.

Each entry contains the logical key name, the number corresponding to the physical key (there is a function in renegade to do the conversion which I have hooked and am using) and the current "key value" (i.e. the value returned from the renegade input code)

Then, it installs a hook into the renegade input processing. This hook iterates through the key list created when the config file was read in.

for each key, it asks the input system for the "state" of the physical key involved.

some extra code is then invoked to make sure that keypresses are only triggered once (I can make it recognize multiple keypresses if a use comes up where recognizing repeat keypresses/key is held down arrises but for now it was easier to make it only recognize one keypress at a time)

After this, if the result was that the key was pressed (and it was in the released state last time the input check ran), a message is sent to the server part of the code.

This contains the logical key name that has been depressed along with the player ID of the player who pressed the key.

A script notifies the server code that it wants to process a given key by handing it 4 things:

The address of the hook procedure

The logical key it is interested in

The player ID it is interested in (obtained e.g. from a GameObject via Get_Player_ID)

and a piece of data that can be whatever you want.

The same data will be given back to the hook procedure when it is called.

There is also a process to unregister a hook when you have finished (e.g. when the infantry unit with JFW_Jetpack on it dies or buys a new character)

When the keypress message is recieved by the server, it loops through all the registered hooks. If it finds a hook where the player ID and logical key name match with the ones sent from the client, the hook is called (passing in the data that was passed in when the hook was reigstered). Then, the hook can do whatever you want it to do.

So far, the hook has been used for the Jetpack script which is to go on the jumpjet infantry for Reborn, it can also be used for the soviet cosmonaut infantry in RA2Ven if required and in fact anything else where you want infantry that can "fly" (including if you wanted spellcasting infantry that could levitate).

It is also planned to be used for the underground digging logic (Subterranean APC/flame tank) and the deploying vehicle logic (e.g. tick tank and lots of other stuff in many games)
