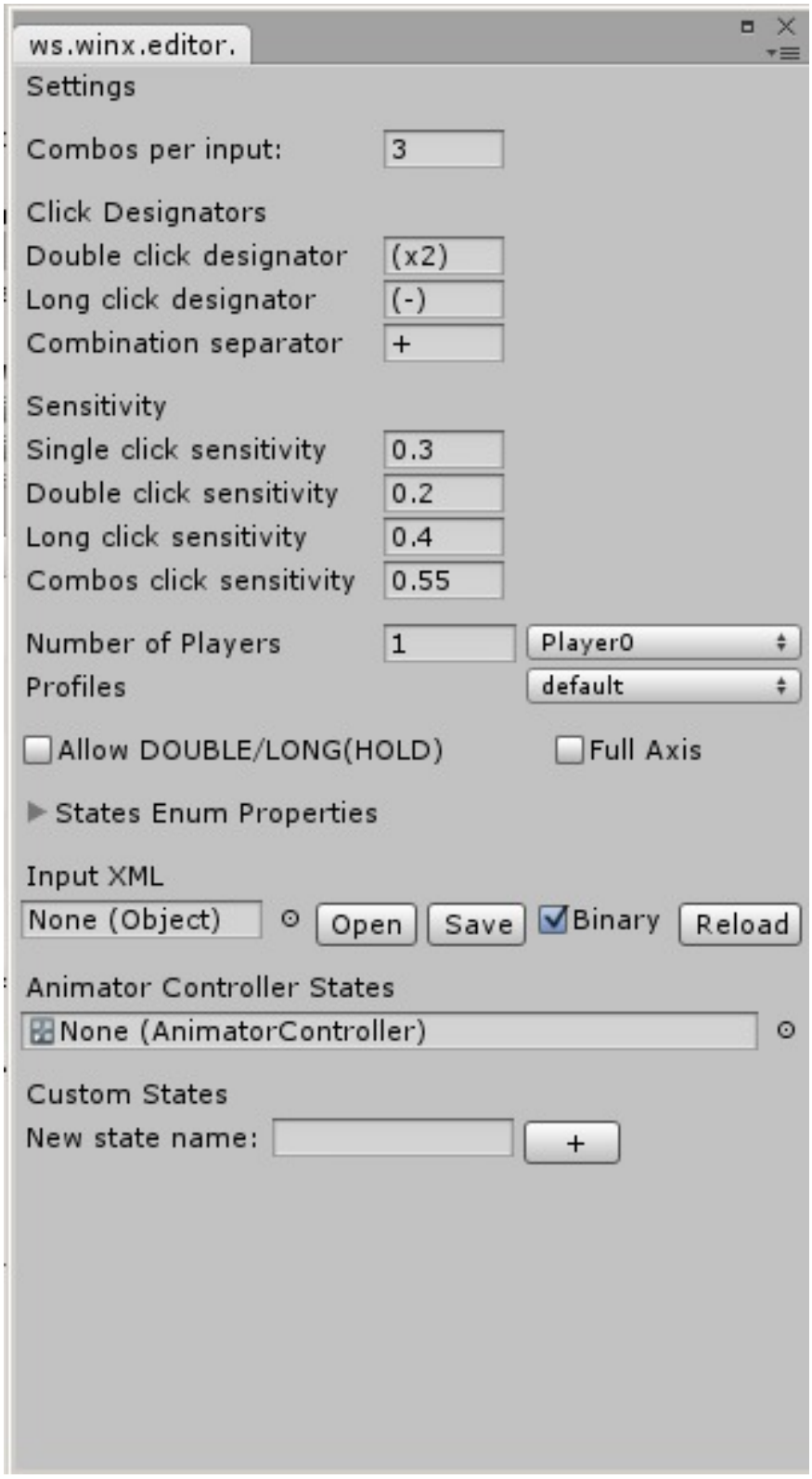
1) Install the package
2) Put gmcs.rsp and smcs.rsp in root
3) Put „InputSettings.xml/bin“ in StreamingAssets folder from package if you want to hit play on PlayerVsPlayer demo
4) Connect your devices
5) Open Window->InputMapper->InputMapper editor

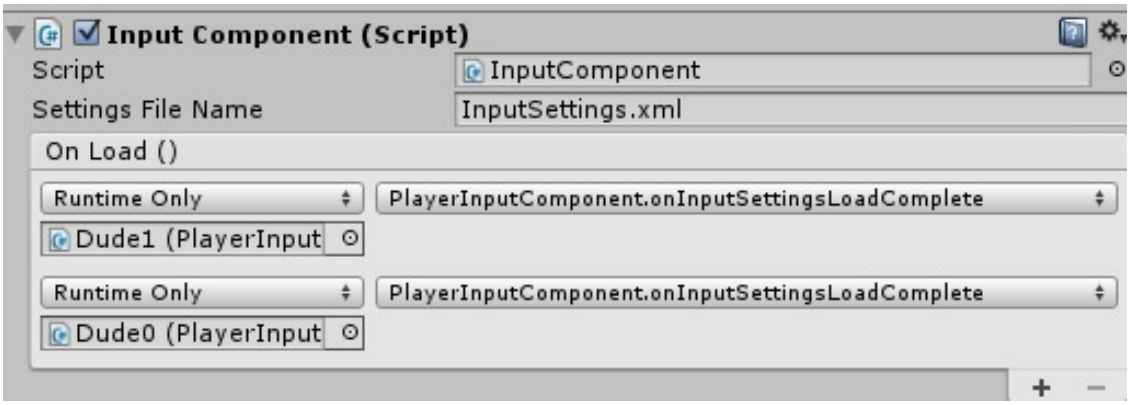


6) Create custom states

Advanced:

```
InputManager.MapStateToInput("Click_W
+C_State",KeyCodeExtension.W.DOUBLE,KeyCodeExtension.C.LONG);
```

7) Map your input into combination by clicking buttons or moving joysticks. (single,double,long(hold) action)
8) Save your settings as .xml or .bin (States.cs would be generated for you so you can access your states as States.MyState)
9) Drag and drop InputComponent to some GameObject

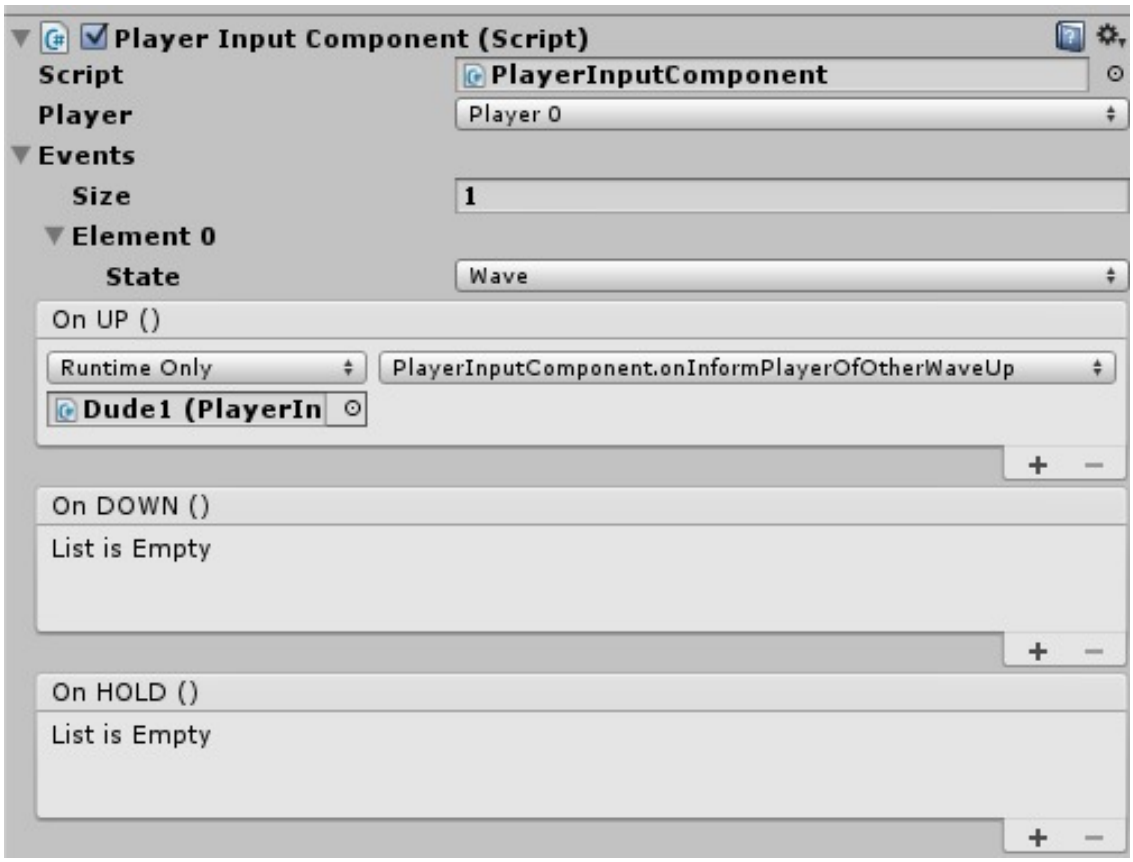


Component would handle loading and raise event when loading is complete.

Advanced:

```
InputManager.loadSettings(Path.Combine(Application.streamingAssetsPath,"InputSettings.xml"));
```

10) Drag and drop PlayerComponent to character's game objects



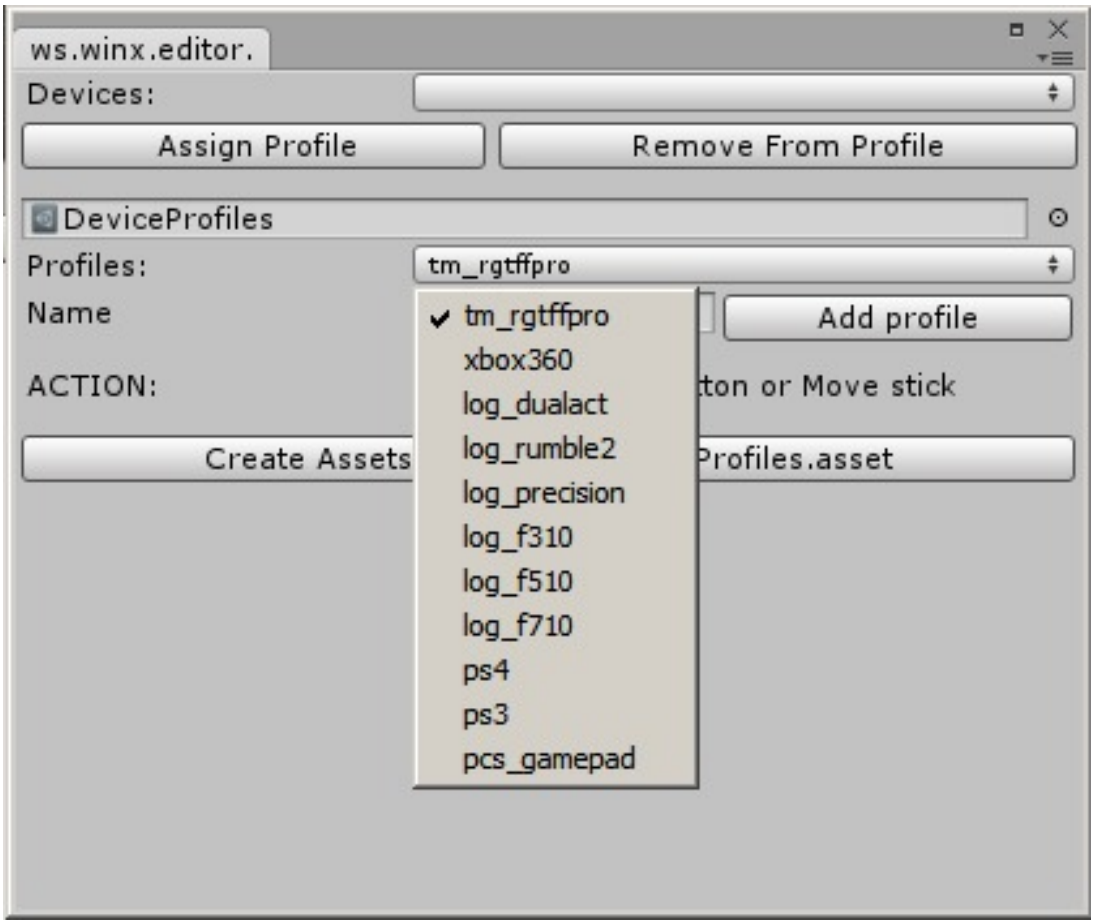
Advanced:

```
InputManager.addEventListener ((int)States.Wave, Player).UP += onUp;
InputManager.addEventListener ((int)States.Wave, Player).DOWN += onDown;
```

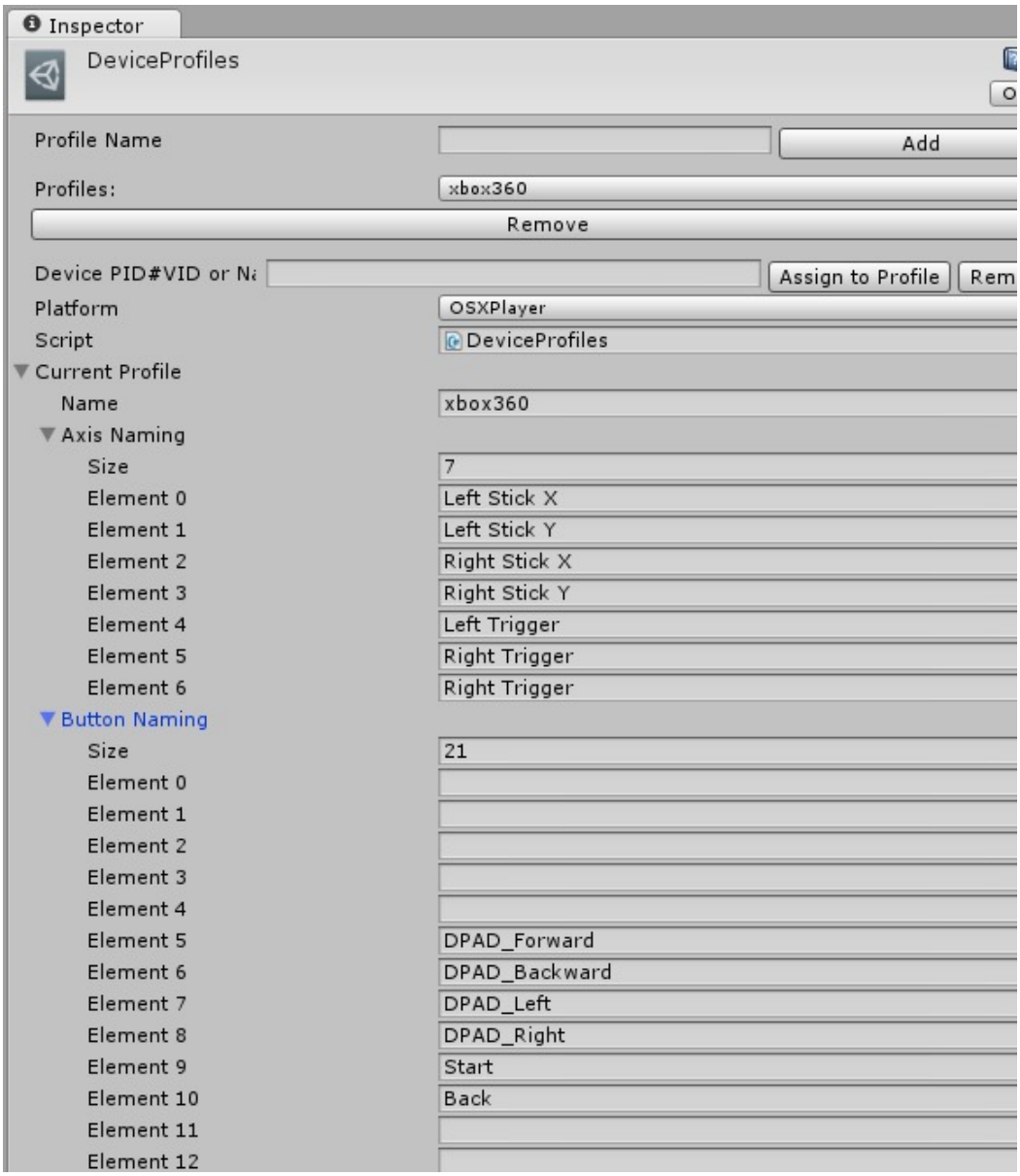
```
public void onInformPlayerOfOtherWaveUp ()
{
        Debug.Log ("Inform "+Player+" of Player0 Wave UP");

}

void onHold ()
{
        Debug.Log (Player + ">Wave state trigger Hold");
}

void onUp ()
{
        Debug.Log (Player + ">Wave state trigger Up");
}
```

```
                void onDown ()
                {
                                Debug.Log (Player + ">Wave state trigger Down");
                }

        // Update is called once per frame
        void Update ()
        {

        if (InputManager.GetInputDown ((int)States.Wave, Player, true)) {
                                                animator.Play ((int)States.Wave);
                                }

        if (InputManager.GetInputDown ((int)States.Jump, Player)) {
                                                animator.Play ((int)States.Jump);
                                }


        //
        //Math.Abs prevent code to function differently when axis is inverted
float forward = Math.Abs (InputManager.GetInput ((int)States.WalkForward, Player, 0.25f))
                                                - Math.Abs (InputManager.GetInput
((int)States.WalkBackward, Player, 0.25f));



        animator.SetFloat (forwardHash, forward);
                                //
                                //
float turn = Math.Abs (InputManager.GetInput ((int)States.TurnRight, Player, 0.25f))
                                                - Math.Abs (InputManager.GetInput
((int)States.TurnLeft, Player, 0.25f));

                                animator.SetFloat (turnHash, turn);
```

11) Create Device Profile by use of Window->InputMapper->Device Profiling



12) Click buttons or move axis to see the action and name the action. Data is stored in Resources>DataProfiles.asset for 85 devices.
13) Select DataProfiles.asset to add or edit profile's entries for designated OS. Add more devices in VID#PID (vendor ID product ID) or device name (UnityDriver only).



14) Open PlayerVSPlayerDemo.unity and hit PLAY.
15) Write custom device driver on high level (see XinputDriver.cs,WiimoteDriver.cs,ThrustMasterDriver.cs in source) and add it to the manger.

```
InputManager.AddDriver(new XinputDriver());
```

16) Set UnityDriver as default.

```
InputManager.hidInterface.defaultDriver=new UnityDriver();
```

Warring:Unity doesn't make distinction between triggers/axis/pointOfView and doesn't make controller

distinction as multiply instances of same type have same name and can hard code index of devices no matter position in GetJoystickNames list

Still hard to understand? Go watch and read on bellow links.
https://www.youtube.com/watch?v=Pir49v16aOQ

GITHUB: https://github.com/winalex/Unity3d-InputMapper

WiiDriver Demo
https://www.youtube.com/watch?v=tEwMcA2ZaMk

ThrustMasterDriver Demo (Win+Droid+Osx)
https://www.youtube.com/watch?v=uoH-RfopGzk

Web Player demo(thru Gamepad API):
https://www.youtube.com/watch?v=sXMWmn4Kc9c
MORE:
http://unity3de.blogspot.com/2014/06/unity-input-manger-pain-is-spreading.html

Feel free to contact me with your contribution, bugs reports, idea....
winxalex@gmail.com