
Blender-VR API

Release 1.0

Blender-VR Team

January 22, 2015

1 Processor File	2
1.1 Quick-Start	2
1.2 Examples	2
1.3 Architecture	4
2 Source Code	5
2.1 Utils	5
2.2 Modules	6
3 Indices and tables	37
Python Module Index	38

The documentation of the Blender-VR project includes the part that is exposed to the user to be used in the processor file, as well as the core code of Blender-VR.

Processor File

1.1 Quick-Start

[To be written]

1.2 Examples

For more examples, check the `processor` files in the [Samples Repository](#) of the Blender-VR project.

1.2.1 Basic Example

This is a basic `processor` file which can be considered a barebone and a start point for your own. All it does is to synchronize all the objects between the master and the slaves machines.

```

1 import blendervr
2
3 if blendervr.is_console():
4     class Processor(blendervr.processor.getProcessor()):
5         def __init__(self, console):
6             global try_wait_user_name, try_chooser, try_console_arc_balls
7             super(Processor, self).__init__(console)
8
9         def useLoader(self):
10            return True
11
12 elif blendervr.is_creating_loader():
13     import bpy
14
15     class Processor(blendervr.processor.getProcessor()):
16         def __init__(self, creator):
17             super(Processor, self).__init__(creator)
18
19 elif blendervr.is_virtual_environment():
20     import bge
21
22     class Processor(blendervr.processor.getProcessor()):
23         def __init__(self, parent):
24             super(Processor, self).__init__(parent)
25

```

```

26     if self.blenderVR.isMaster():
27         self.blenderVR.getSceneSynchronizer().\
28             getItem(bge.logic).activate(True, True)

```

The file is split in three parts:

1. Console
2. Update Loader
3. Virtual Environment

The processor file is called three times, and each time a section of it is called.

Console

The console part of the code is called first by the `console`. This runs before your `.blend` file is even loaded. The `useLoader()` determines if you need Blender-VR to modify your `.blend` on-the-fly.

Most of the time this won't need to change. The exception is when the file being loaded was already modified to work with Blender-VR (e.g., the file generated on-the-fly after running it once).

```

if blendervr.is_console():
    class Processor(blendervr.processor.getProcessor()):
        def __init__(self, console):
            global try_wait_user_name, try_chooser, try_console_arc_balls
            super(Processor, self).__init__(console)

        def useLoader(self):
            return True

```

Update Loader

If a project requires specific changes in the `.blend` file they are introduced here. This is the place where a specific Actuator can be added for a Head-Mounted display for example.

```

elif blendervr.is_creating_loader():
    import bpy

    class Processor(blendervr.processor.getProcessor()):
        def __init__(self, creator):
            super(Processor, self).__init__(creator)

```

Virtual Environment

This part of the code is called when the `.blend` file is loaded in the Blender Game Engine. The most basic usage is to syncronize all the scene objects, as it's being done here.

```

elif blendervr.is_virtual_environment():
    import bge

    class Processor(blendervr.processor.getProcessor()):
        def __init__(self, parent):
            super(Processor, self).__init__(parent)

        if self.blenderVR.isMaster():

```

```
self.blenderVR.getSceneSynchronizer().\  
    getItem(bge.logic).activate(True, True)
```

1.3 Architecture

[To be written]

Source Code

2.1 Utils

Files to support the Blender VR application.

2.1.1 Daemon

This script runs in the clients and is responsible for spawning the Blender Player.

class `daemon.Daemon(blenderVR_modules)`

Bases: `builtins.object`

Background management of the Blender Player and related stuff.

main()

Start the Daemon, quits any instance of BlenderPlayer running.

processCommand(command, argument)

Run the received commands

Parameters

- **command** (`str`) – Command to execute in the client machine

- **argument** – Value depends on the command

write(*messages)

Send message to the client

Parameters `messages (list)` – all the messages to send to the client (i.e., console commands)

`daemon.main()`

Main function to start the daemon.

Prepare execution (daemonize if necessary), then build a Daemon and call its main() method to manage background communications.

2.1.2 Update Loader

Script that runs in Blender in background mode to transform the `.blend` file into a Blender-VR ready file. A few Logic Bricks are created among other changes in the initial scene.

2.1.3 Blender-VR Console

Blender-VR console mode

```
blenderVR.main()
```

2.2 Modules

Main modules for the Blender-VR application.

2.2.1 blendifvr package

Subpackages

`blendifvr.console` package

Subpackages

`blendifvr.console.gui` package

Submodules

`blendifvr.console.gui.console` module

```
class blendifvr.console.gui.console.GUI
    Bases: builtins.object

    display_screen_sets(screenSets)

    main()

    quit()

    start()
```

`blendifvr.console.gui.logger` module

```
class blendifvr.console.gui.logger.Logger(parent, config_index)
    Bases: blendifvr.console.base.Base

    clear()

    flush()

    quit()

    set_log_level(log_level)

    set_stream_state(stream, state)

    start()

    write(*messages)
```

blendervr.console.gui.options module

```
class blendervr.console.gui.options.GUI
    Bases: builtins.object

    is_options_window_opened()
    quit()
    start()
```

blendervr.console.gui.screen module

```
class blendervr.console.gui.screen.GUI
    Bases: builtins.object

    getFrame()
    is_log_window_opened()
    quit()
    start()
```

blendervr.console.gui.screens module

```
class blendervr.console.gui.screens.GUI
    Bases: builtins.object

    quit()
    show(state)
    start()
    update_gui()
```

Module contents**blendervr.console.logic package****Submodules****blendervr.console.logic.console module**

```
class blendervr.console.logic.console.Logic
    Bases: builtins.object

    compile_BC()
    get_blender_player_state()
    load_configuration_file()
    quit()
    receivedFromVirtualEnvironment(message)
    sendToVirtualEnvironment(command, argument)
    set_screen_set()
    start()
```

```

start_simulation()
stop_simulation()
update_user_files (force=False)

blendervr.console.logic.file_name module
class blendervr.console.logic.file_name.FileName (file_name, anchor=None)
    Bases: builtins.object
        strip (anchor)
        unstrip (anchor)

blendervr.console.logic.screen module
class blendervr.console.logic.screen.Logic (net_console)
    Bases: builtins.object
        adapt_simulation_files_to_screen (loader_file, blender_file, processor_files)
        ask_blender_player_to_quit ()
        daemon_is_running ()
        getHostname ()
        get_blender_player_state ()
        is_master ()
        quit ()
        restartDaemon ()
        send_to_blender_player (command, argument='')
        setConfiguration (configuration, complements)
        setHierarchy (informations)
        setNetworkClient (origin, client, addr)
        set_blenderVR_state (state)
        start ()

blendervr.console.logic.screens module
class blendervr.console.logic.screens.Logic
    Bases: builtins.object
        adapt_simulation_files_to_screen (loader_file, blender_file, processor_files)
        getMaster ()
        getScreen (screen_name)
        getScreensNumber ()
        getStates ()
        quit ()
        send_to_blender_player (command, message)
        set_screens (configurations, net_console, master_name, port, complements)

```

```
start()
start_simulation()
stop_simulation()
```

Module contents

blendervr.console.qt package

Submodules

```
blendervr.console.qt.console module
class blendervr.console.qt.console.GUI
    Bases: blendervr.console.gui.console.GUI

    addListenTo (socket, callback, data=None)
    addTimeout (time, callback)
    cb_add_configuration_path()
    cb_close()
    cb_load_configuration_file()
    cb_processor_window()
    cb_remove_configuration_path(*args)
    cb_set_blender_file()
    cb_set_configuration_file()
    cb_set_current_tab()
    cb_set_link_processor_to_blender()
    cb_set_processor_file()
    cb_set_screen_set()
    cb_start()
    cb_stop()
    cb_update_liste_paths(*args)
    display_screen_sets(screenSets)
    getWindow()
    main()
    quit()
    removeListenTo(tag)
    start()
    updateStatus (message, state=None)
    update_processor()
```

```
class blndervr.console.qt.console.MainWindow(owner, profile, profile_indices)
    Bases: blndervr.tools.gui.qt.Common, dummy.QMainWindow

blndervr.console.qt.console.quit()

blndervr.console.qt.logger module
class blndervr.console.qt.logger.Logger(parent, config_index, window, log_level_selector)
    Bases: blndervr.console.gui.logger.Logger

    cb_set_log_level()
    clear()

blndervr.console.qt.options module
class blndervr.console.qt.options.GUI(parent)
    Bases: blndervr.console.base.Base, blndervr.console.gui.options.GUI

    blenderVR_state_changed(state)
    cb_debug_daemon()
    cb_debug_processor()
    cb_executables()
    cb_reload_processor()
    cb_restart_daemons()
    close()
    getMenu()
    getmenu(index)
    quit()
    start()

blndervr.console.qt.screen module
class blndervr.console.qt.screen.GUI
    Bases: blndervr.console.gui.screen.GUI

    cb_toggle_stderr_state()
    cb_toggle_stdout_state()
    close()
    getMenu(index)
    is_log_window_opened()
    quit()
    start()
```

blendervr.console.qt.screens module

class `blendervr.console.qt.screens.GUI`
Bases: `blendervr.console.gui.screens.GUI`

close_all()
quit()
start()
update_gui()

blendervr.console.qt.tools module

class `blendervr.console.qt.tools.MessagesColors`
Bases: `builtins.object`

getColors(level)

Module contents**blendervr.console.xml package****Subpackages****blendervr.console.xml.device package****Submodules****blendervr.console.xml.device.base module**

class `blendervr.console.xml.device.base.Base` (*parent, name, attrs*)
Bases: `blendervr.console.xml.base.XML`

blendervr.console.xml.device.hmd module

class `blendervr.console.xml.device.hmd.Device` (*parent, name, attrs*)
Bases: `blendervr.console.xml.device.base.Base`

blendervr.console.xml.device.screen module

class `blendervr.console.xml.device.screen.Screen` (*parent, name, attrs*)
Bases: `blendervr.console.xml.device.base.Base`

characters(string)
endElement(name)

blendervr.console.xml.device.wall module

class `blendervr.console.xml.device.wall.Device` (*parent, name, attrs*)
Bases: `blendervr.console.xml.device.screen.Screen`

Module contents

`blendervr.console.xml.device.getDevice` (*parent, name, attrs*)

Submodules

blendervr.console.xml.base module

```
class blendervr.console.xml.base.XML (parent, name, attrs)
    Bases: xml.sax.handler.ContentHandler, blendervr.console.base.Base

        endElement (name)
        getBoolean (value)
        getConfiguration ()
        getMain ()
        getParser ()
        getVector (vector, size, none_value=None)
        getXML_FileName ()
        getXML_LineNumber ()
        getXML_Position ()
        print_warning (msg)
        raise_error (msg)
        startElement (name, attrs)
        which (filename)

class blendervr.console.xml.base.mono (parent, name, attrs)
    Bases: blendervr.console.xml.base.XML

        characters (string)
        endElement (name)
        startElement (name, attrs)

class blendervr.console.xml.base.single (parent, name, attrs)
    Bases: blendervr.console.xml.base.XML

        characters (string)
        endElement (name)
        startElement (name, attrs)
```

blendervr.console.xml.behavior module

```
class blendervr.console.xml.behavior.XML (parent, name, attrs)
    Bases: blendervr.console.xml.reusable.XML

        characters (string)
        endElement (name)
```

blendervr.console.xml.blendervr module

```
class blendervr.console.xml.blendervr.XML (parent, name, attrs)
    Bases: blendervr.console.xml.base.XML
```

blendervr.console.xml.common_system module

class `blendervr.console.xml.common_system.XML` (*parent, name, attrs*)
Bases: `blendervr.console.xml.reusable.XML`

blendervr.console.xml.computer module

class `blendervr.console.xml.computer.container` (*parent, name, attrs*)
Bases: `blendervr.console.xml.base.XML`
class `blendervr.console.xml.computer.main` (*parent, name, attrs*)
Bases: `blendervr.console.xml.base.XML`

blendervr.console.xml.display module

class `blendervr.console.xml.display.XML` (*parent, name, attrs*)
Bases: `blendervr.console.xml.reusable.XML`
 characters (*string*)
 endElement (*name*)
class `blendervr.console.xml.display.graphic_buffer` (*parent, name, attrs*)
Bases: `blendervr.console.xml.base.mono`

blendervr.console.xml.plugins module

class `blendervr.console.xml.plugins.XML` (*parent, name, attrs*)
Bases: `blendervr.console.xml.base.XML`

blendervr.console.xml.processors module

class `blendervr.console.xml.processors.XML` (*parent, name, attrs*)
Bases: `blendervr.console.xml.base.single`
 characters (*string*)
 getConfiguration ()

blendervr.console.xml.reusable module

class `blendervr.console.xml.reusable.XML` (*parent, name, attrs*)
Bases: `blendervr.console.xml.base.XML`

blendervr.console.xml.root module

class `blendervr.console.xml.root.XML` (*parent*)
Bases: `blendervr.console.xml.base.XML`
 getConfiguration ()

blendervr.console.xml.screen module

class `blendervr.console.xml.screen.container` (*parent, name, attrs*)
Bases: `blendervr.console.xml.base.XML`
class `blendervr.console.xml.screen.main` (*parent, name, attrs*)
Bases: `blendervr.console.xml.base.XML`

blendervr.console.xml.starter module**class** `blendervr.console.xml.starter.XML (parent, name, attrs)`Bases: `blendervr.console.xml.common_system.XML`**characters** (*string*)**blendervr.console.xml.system module****class** `blendervr.console.xml.system.XML (parent, name, attrs)`Bases: `blendervr.console.xml.common_system.XML`**class** `blendervr.console.xml.system.blenderplayer (parent, name, attrs)`Bases: `blendervr.console.xml.base.single`**characters** (*string*)**class** `blendervr.console.xml.system.daemon (parent, name, attrs)`Bases: `blendervr.console.xml.base.single`**characters** (*string*)**class** `blendervr.console.xml.system.log (parent, name, attrs)`Bases: `blendervr.console.xml.base.mono`**class** `blendervr.console.xml.system.login (parent, name, attrs)`Bases: `blendervr.console.xml.base.mono`**blendervr.console.xml.user module****class** `blendervr.console.xml.user.container (parent, name, attrs)`Bases: `blendervr.console.xml.base.XML`**class** `blendervr.console.xml.user.main (parent, name, attrs)`Bases: `blendervr.console.xml.base.XML`**Module contents****class** `blendervr.console.xml.Configure (parent, config_paths, config_file)`Bases: `xml.sax.handler.ContentHandler, xml.sax.handler.EntityResolver, blendervr.console.base.Base`**getConfiguration ()****resolveEntity** (*publicID*, *systemID*)**Submodules****blendervr.console.base module****class** `blendervr.console.base.Base (parent)`Bases: `blendervr.base.Base`**getConsole ()****profile****quit ()**

blendervr.console.console module

```
class blendervr.console.console.Console (profile_file)
    Bases: blendervr.console.logic.console.Logic, blendervr.console.qt.console.GUI

    logger
    plugins
    profile
    quit ()
    start ()
```

blendervr.console.exceptions module

```
exception blendervr.console.exceptions.Main
    Bases: builtins.Exception
```

blendervr.console.profile module

```
class blendervr.console.profile.Profile (configuration_file)
    Bases: builtins.object

    appendValue (index, value, write=True)
    dump ()
    getValue (index)
    lock (lock)
    prependValue (index, value, write=True)
    setDefault (default, node=None, root=None)
    setValue (index, value, write=True)
```

blendervr.console.screen module

```
class blendervr.console.screen.Screen (screens, name, net_console)
    Bases: blendervr.console.base.Base, blendervr.console.logic.screen.Logic, blendervr.console.qt.screen.GUI

    getName ()
    logger
    main_logger
    quit ()
    start ()
```

blendervr.console.screens module

```
class blendervr.console.screens.Screens (parent)
    Bases: blendervr.console.base.Base, blendervr.console.logic.screens.Logic, blendervr.console.qt.screens.GUI

    quit ()
    start ()
```

Module contents

```
blendervr.console.main()
blendervr.console.stripAnchor(anchor, path)
blendervr.console.unstripAnchor(anchor, path)
```

blendervr.interactor package**Subpackages****blendervr.interactor.arc_ball package****Submodules****blendervr.interactor.arc_ball.console module**

```
class blendervr.interactor.arc_ball.console.Console(parent, name='update arcballs')
    Bases: blendervr.interactor.Interactor

    receivedFromConsole(command, argument)
    selectObject(_object)
```

Module contents

```
class blendervr.interactor.arc_ball.ArcBall(parent, NewWidth, NewHeight)
    Bases: blendervr.interactor.Interactor

    click(mouse_position)
    drag(Point2fT mouse_coord) → new_quaternion_rotation_vec
    getMatrixFromQuaternion(quaternion)
    setBounds(NewWidth, NewHeight)
    setOrientation(direct)
    removeScale(matrix)
```

Submodules**blendervr.interactor.head_controlled_navigation module**

```
class blendervr.interactor.head_controlled_navigation.HCNav(parent, method=None,
    one_per_user=True)
    Bases: blendervr.interactor.Interactor

    CALIBRATE = 'calibrate'
    RESET = 'reset'
    START = 'start'
    STOP = 'stop'
    TOGGLE = 'toggle'
    getNavigationState()
    receivedFromConsole(command, argument)
```

```

setDefaultUser(user)
setHeadLocation(user, info)
setHeadNeckLocation(location, user=None)
setOrientationFactors(attenuation, power, max=1.0, user=None)
setPositionFactors(component, attenuation, power, max=1.0, user=None)
update(state, user=None)

```

blendervr.interactor.landmarks module

```

class blendervr.interactor.landmarks.LandMarks(parent, name='land marks')
    Bases: blendervr.interactor.Interactor

receivedFromConsole(command, argument)

```

blendervr.interactor.laser module

```

class blendervr.interactor.laser.Laser(parent, object, sensor)
    Bases: blendervr.player.base.Base

allowDisallowObjects(allow, objects=None)
attachLaserToCamera()
display_laser()
getHitObject(grabbed=False)
getObject()
getSynchronizerBuffer()
grab()
processSynchronizerBuffer(buf)
release()
toggle()

```

blendervr.interactor.object_chooser module

```

class blendervr.interactor.object_chooser.Chooser(parent)
    Bases: blendervr.interactor.Interactor

```

blendervr.interactor.reset_objects module

```

class blendervr.interactor.reset_objects.ResetObjects(parent)
    Bases: blendervr.player.base.Base

reset()
save()

```

blendervr.interactor.viewpoint module

```

class blendervr.interactor.viewpoint.ViewPoint(parent)
    Bases: blendervr.interactor.Interactor

activation(activate)
isActivated()

```

```

keyboardAndMouse (info)
run ()
viewpointScale

blendervr.interactor.wavefront_obj module
class blendervr.interactor.wavefront_obj.Reader (file_name)
    Bases: builtins.object

        draw (centered=True, normalized=True)
        getMinMax ()

```

Module contents

```

class blendervr.interactor.Interactor (parent)
    Bases: blendervr.player.base.Base

        keyboardAndMouse (info)
        receivedFromConsole (command, argument)
        run ()
        sendToConsole (command, argument='')
        user_position (info)

```

blendervr.loader package**Submodules**

blendervr.loader.base module

```

class blendervr.loader.base.Base (parent)
    Bases: blendervr.base.Base

        process (controller)
    blendervr.loader.base.main ()

```

Module contents

```

class blendervr.loader.Creator (logger)
    Bases: builtins.object

        process ()
    blendervr.loader.main ()

```

blendervr.player package**Subpackages****blendervr.player.network package****Subpackages**

blendervr.player.network.synchronizer package**Subpackages****blendervr.player.network.synchronizer.objects package****Submodules****blendervr.player.network.synchronizer.objects.item_armature_bone module**

```
class blendervr.player.network.synchronizer.objects.item_armature_bone.ArmatureBone
    Bases: builtins.object
class blendervr.player.network.synchronizer.objects.item_armature_bone.Master (parent,
    item)
    Bases: blendervr.player.network.synchronizer.objects.item_armature_bone.ArmatureBone,
            blendervr.player.network.synchronizer.objects.item_base.Master
        getSynchronizerBuffer ()

class blendervr.player.network.synchronizer.objects.item_armature_bone.Slave (parent,
    item)
    Bases: blendervr.player.network.synchronizer.objects.item_armature_bone.ArmatureBone,
            blendervr.player.network.synchronizer.objects.item_base.Slave
        processSynchronizerBuffer (buff)
```

blendervr.player.network.synchronizer.objects.item_armature_channel module

```
class blendervr.player.network.synchronizer.objects.item_armature_channel.ArmatureChannel
    Bases: builtins.object
class blendervr.player.network.synchronizer.objects.item_armature_channel.Master (parent,
    item)
    Bases: blendervr.player.network.synchronizer.objects.item_armature_channel.ArmatureChannel,
            blendervr.player.network.synchronizer.objects.item_base.Master
        getSynchronizerBuffer ()

class blendervr.player.network.synchronizer.objects.item_armature_channel.Slave (parent,
    item)
    Bases: blendervr.player.network.synchronizer.objects.item_armature_channel.ArmatureChannel,
            blendervr.player.network.synchronizer.objects.item_base.Slave
        processSynchronizerBuffer (buff)
```

blendervr.player.network.synchronizer.objects.item_armature_object module

```
class blendervr.player.network.synchronizer.objects.item_armature_object.ArmatureObject
    Bases: builtins.object
    FRAME = b'f'
    OBJECT = b'o'
class blendervr.player.network.synchronizer.objects.item_armature_object.Master (parent,
    item)
    Bases: blendervr.player.network.synchronizer.objects.item_armature_object.ArmatureObject,
            blendervr.player.network.synchronizer.objects.item_object.Master
        getSynchronizerBuffer ()
```

```

class blendervr.player.network.synchronizer.objects.item_armature_object.Slave (parent, item)
    Bases: blendervr.player.network.synchronizer.objects.item_armature_object.ArmatureObject,
           blendervr.player.network.synchronizer.objects.item_object.Slave
        processSynchronizerBuffer (buff)

blendervr.player.network.synchronizer.objects.item_base module
class blendervr.player.network.synchronizer.objects.item_base.Base (parent, item)
    Bases: blendervr.player.base.Base
        addChildren (children)
        checkItems ()
        default ()
        getItemID ()
        isSynchronizable ()
        remove ()
class blendervr.player.network.synchronizer.objects.item_base.Master (parent, item)
    Bases: blendervr.player.network.synchronizer.objects.item_base.Base
        activate (enable, recursive=False)
        addChildren (children)
        getSynchronizerBuffer ()
        remove ()
exception blendervr.player.network.synchronizer.objects.item_base.NotExistingItem
    Bases: builtins.Exception

class blendervr.player.network.synchronizer.objects.item_base.Slave (parent, item)
    Bases: blendervr.player.network.synchronizer.objects.item_base.Base
        getItemByName (name, sg_parent)
        processSynchronizerBuffer (buffer)
        removeChildren (children)

blendervr.player.network.synchronizer.objects.item_camera module
class blendervr.player.network.synchronizer.objects.item_camera.Camera
    Bases: builtins.object
class blendervr.player.network.synchronizer.objects.item_camera.Master (parent, item)
    Bases: blendervr.player.network.synchronizer.objects.item_camera.Camera,
           blendervr.player.network.synchronizer.objects.item_object.Master
class blendervr.player.network.synchronizer.objects.item_camera.Slave (parent, item)
    Bases: blendervr.player.network.synchronizer.objects.item_camera.Camera,
           blendervr.player.network.synchronizer.objects.item_object.Slave

blendervr.player.network.synchronizer.objects.item_default module
class blendervr.player.network.synchronizer.objects.item_default.Default
    Bases: builtins.object
        isSynchronizable ()

```

```

class blendervr.player.network.synchronizer.objects.item_default.Master (parent, item)
    Bases:           blendervr.player.network.synchronizer.objects.item_default.Default,
                    blendervr.player.network.synchronizer.objects.item_base.Master

class blendervr.player.network.synchronizer.objects.item_default.Slave (parent, item)
    Bases:           blendervr.player.network.synchronizer.objects.item_default.Default,
                    blendervr.player.network.synchronizer.objects.item_base.Slave

blendervr.player.network.synchronizer.objects.item_font module
class blendervr.player.network.synchronizer.objects.item_font.Font
    Bases: builtins.object

class blendervr.player.network.synchronizer.objects.item_font.Master (parent, item)
    Bases:           blendervr.player.network.synchronizer.objects.item_font.Font,
                    blendervr.player.network.synchronizer.objects.item_object.Master

class blendervr.player.network.synchronizer.objects.item_font.Slave (parent, item)
    Bases:           blendervr.player.network.synchronizer.objects.item_font.Font,
                    blendervr.player.network.synchronizer.objects.item_object.Slave

blendervr.player.network.synchronizer.objects.item_light module
class blendervr.player.network.synchronizer.objects.item_light.Light
    Bases: builtins.object

class blendervr.player.network.synchronizer.objects.item_light.Master (parent, item)
    Bases:           blendervr.player.network.synchronizer.objects.item_light.Light,
                    blendervr.player.network.synchronizer.objects.item_object.Master

class blendervr.player.network.synchronizer.objects.item_light.Slave (parent, item)
    Bases:           blendervr.player.network.synchronizer.objects.item_light.Light,
                    blendervr.player.network.synchronizer.objects.item_object.Slave

blendervr.player.network.synchronizer.objects.item_object module
class blendervr.player.network.synchronizer.objects.item_object.Master (parent, item)
    Bases:           blendervr.player.network.synchronizer.objects.item_object.Object,
                    blendervr.player.network.synchronizer.objects.item_base.Master

        getSynchronizerBuffer()

class blendervr.player.network.synchronizer.objects.item_object.Object
    Bases: builtins.object

        ORIENTATION = b'o'
        POSITION = b'p'
        SCALE = b's'
        VISIBILITY = b'v'
        default()

class blendervr.player.network.synchronizer.objects.item_object.Slave (parent, item)
    Bases:           blendervr.player.network.synchronizer.objects.item_object.Object,
                    blendervr.player.network.synchronizer.objects.item_base.Slave

        processSynchronizerBuffer(buff)

```

blendervr.player.network.synchronizer.objects.item_root module

```
class blendervr.player.network.synchronizer.objects.item_root.Master (parent, item)
    Bases:               blendervr.player.network.synchronizer.objects.item_root.Root,
                        blendervr.player.network.synchronizer.objects.item_base.Master
class blendervr.player.network.synchronizer.objects.item_root.Root
    Bases: builtins.object

    default ()

    getItemID ()

class blendervr.player.network.synchronizer.objects.item_root.Slave (parent, buffer)
    Bases:               blendervr.player.network.synchronizer.objects.item_root.Root,
                        blendervr.player.network.synchronizer.objects.item_base.Slave
    processSynchronizerBuffer (buffer)
```

blendervr.player.network.synchronizer.objects.item_scene module

```
class blendervr.player.network.synchronizer.objects.item_scene.Master (parent, item)
    Bases:               blendervr.player.network.synchronizer.objects.item_scene.Scene,
                        blendervr.player.network.synchronizer.objects.item_base.Master
    getSynchronizerBuffer ()

class blendervr.player.network.synchronizer.objects.item_scene.Scene
    Bases: builtins.object

    default ()

class blendervr.player.network.synchronizer.objects.item_scene.Slave (parent, buff)
    Bases:               blendervr.player.network.synchronizer.objects.item_scene.Scene,
                        blendervr.player.network.synchronizer.objects.item_base.Slave
    getItemByName (name, sg_parent)
    processSynchronizerBuffer (buff)
```

blendervr.player.network.synchronizer.objects.master module

```
class blendervr.player.network.synchronizer.objects.master.Master (parent)
    Bases: blendervr.player.network.synchronizer.objects.Synchronizer

    addItem (children, parent)
    checkItems ()
    getSynchronizerBuffer ()
    removeItem (item)
    sendItemsUpdateToSlaves (buff)
```

blendervr.player.network.synchronizer.objects.slave module

```
class blendervr.player.network.synchronizer.objects.slave.Slave (parent)
    Bases: blendervr.player.network.synchronizer.objects.Synchronizer

    getObjectByMasterID (master_id)
    processSynchronizerBuffer (buff)
    start ()
```

Module contents

```
class blendervr.player.network.synchronizer.objects.Synchronizer (parent)
    Bases: blendervr.player.base.Base

    CREATE_ITEM = b'c'
    DELETE_ITEM = b'd'
    END_UPDATE_ITEM = b'e'
    SET_ATTRIBUTE = b'a'
    UPDATE_ITEM = b'u'

    checkItems ()
    getItem (item)
    removeSynchronizedItem (itemID)
    start ()
```

Module contents

```
class blendervr.player.network.synchronizer.Base (parent)
    Bases: blendervr.player.base.Base

    DEL_OBJECT = b'c'
    NEW_OBJECT = b'c'
    OBJECT = b'o'

    addObjectToSynchronize (object, name)
    addSynchronizedObject (object_id, object)
    getObjectByID (object_id)
    getSceneSynchronizer ()
    removeSynchronizedObject (object_id)
    run ()
    start ()

class blendervr.player.network.synchronizer.Master (parent)
    Bases: blendervr.player.network.synchronizer.Base

    sendSynchronization ()
    start ()

class blendervr.player.network.synchronizer.Slave (parent)
    Bases: blendervr.player.network.synchronizer.Base

    process (buffer)
```

Submodules

blendervr.player.network.connector module

```
class blendervr.player.network.connector.Connector (parent, config)
    Bases: blendervr.player.base.Base

        CMD_FINISHED = b'f'
        CMD_MSG = b'm'
        CMD_SYNCHRO = b's'

        receiveFrom (socket, size)
        selectSocket ()
        sendTo (socket, buffer)
class blendervr.player.network.connector.Master (parent, config)
    Bases: blendervr.player.network.connector.Connector

        barrier ()
        endFrame ()
        quit (reason)
        run ()
        send (session, buff)
        sendToSlave (buff)
        wait_for_everybody ()

class blendervr.player.network.connector.Slave (parent, config)
    Bases: blendervr.player.network.connector.Connector

        barrier ()
        endFrame ()
        quit (reason)
        run ()
        wait_for_everybody ()
```

blendervr.player.network.controller module

```
class blendervr.player.network.controller.Controller (parent)
    Bases: blendervr.player.base.Base

        flush ()
        getConfiguration ()
        run ()
        sendToConsole (command, argument)
        startSimulation ()
        write (*messages)
```

Module contents

```
blendervr.player.network.getNetworks (blenderVR, config)
```

blendervr.player.screen package

Submodules

blendervr.player.screen.base module

blendervr.player.screen.hmd module

blendervr.player.screen.wall module

Module contents

`blendervr.player.screen.getScreen (parent, configuration)`

Submodules

blendervr.player.base module

`class blendervr.player.base.Base (parent)`

Bases: `blendervr.base.Base`

`blenderVR`

blendervr.player.buffer module

`class blendervr.player.buffer.Buffer`

Bases: `builtins.object`

`addPrefix (prefix)`

`boolean (data=None)`

`command (data=None)`

`float (data=None)`

`integer (data=None)`

`isEmpty ()`

`itemID (data=None)`

`matrix_3x3 (data=None)`

`matrix_4x4 (data=None)`

`size (data=None)`

`string (data=None)`

`subBuffer (data=None)`

`unsigned_char (data=None)`

`vector_3 (data=None)`

`vector_4 (data=None)`

`blendervr.player.buffer.data_size (data_type)`

blendervr.player.device module

```
class blendervr.player.device.Base (parent, configuration)
    Bases: blendervr.player.base.Base
class blendervr.player.device.Receiver (main, configuration)
    Bases: blendervr.player.device.Base
        run ()

class blendervr.player.device.Sender (main, configuration)
    Bases: blendervr.player.device.Base
        checkMethod (display_missing)
        process (info)
```

blendervr.player.exceptions module

```
exception blendervr.player.exceptions.Common (reason)
    Bases: builtins.Exception
exception blendervr.player.exceptions.Controller (reason)
    Bases: blendervr.player.exceptions.Common
exception blendervr.player.exceptions.Main (reason)
    Bases: blendervr.player.exceptions.Common
exception blendervr.player.exceptions.Processor (reason)
    Bases: blendervr.player.exceptions.Common
exception blendervr.player.exceptions.Processor_Invalid_Device (reason)
    Bases: blendervr.player.exceptions.Common
exception blendervr.player.exceptions.Processor_Invalid_Device_Method (reason)
    Bases: blendervr.player.exceptions.Processor\_Invalid\_Device
exception blendervr.player.exceptions.Synchronizer (reason)
    Bases: blendervr.player.exceptions.Common
exception blendervr.player.exceptions.User (reason)
    Bases: blendervr.player.exceptions.Common
exception blendervr.player.exceptions.VirtualEnvironment (reason)
    Bases: blendervr.player.exceptions.Common
```

blendervr.player.keyboardAndMouse module

```
class blendervr.player.keyboardAndMouse.KeyboardAndMouse (parent)
    Bases: blendervr.player.device.Sender
        getScreen ()
        run ()
        start ()
```

blendervr.player.splash module

```
class blendervr.player.splash.Splash (parent)
    Bases: blendervr.player.base.Base
        isRunning ()
        sceneIsSplash ()
```

```

setMessage (message)
start ()
stop ()

blendervr.player.user module
class blendervr.player.user.User (parent, id, config)
    Bases: blendervr.player.base.Base

        SYNCHRONIZER_COMMAND_USER_POSITION = b'u'
        SYNCHRONIZER_COMMAND_VEHICLE_POSITION = b'v'
        getEyeSeparation ()
        getID ()
        getName ()
        getPosition ()
        getSynchronizerBuffer ()
        getVehiclePosition (internal=False)
        localTransform
        processSynchronizerBuffer (buff)
        resetVehiclePosition ()
        setParent (parent)
        setPosition (position)
        setVehiclePosition (position)
        worldTransform

```

Module contents @package blendervr

Player Module

```

class blendervr.player.Main
    Bases: builtins.object

    Constructor : load all necessary elements

    MESSAGE_PAUSE = b'p'
    MESSAGE_PROCESSOR = b'r'

    addObjectToSynchronize (object, name)
        Add an object to the synchronizer

    getAllUsers ()
        Get the array of all the users

    getController ()
    getNumberOfNodes ()
    getPlugin (name)
    getProcessor ()

```

```

getScale()
    Get the scale between the virtual World and the Vehicle
    This method always return 1 for the moment: we have to improve the scale management !

getSceneSynchronizer()
    Get the main synchronizer module

getScreenName()

getScreenUsers()

getUserByName(userName)
    Given a user name, get its object, or raise an exception if the user does not exists

getVersion()

isMaster()
    Are we the master rendering node ?

    Many treatment must be done only on the master rendering node. Some others must be done only on one node. In such case, you can check with this method

isPaused()

logger

pause(title='')

quit(reason)
    Main quit method

    This method must be call instead of any other method to properly quit blenderVR. Otherwise, you may have problem of not closed socket next time you run blenderVR. The reason is printed inside the log file of displayed on the console

    Parameters reason (str) –
```

stopDueToError()

wait_for_everybody()

blendervr.plugins package

Subpackages

blendervr.plugins.osc package

Subpackages

blendervr.plugins.osc.virtual_environment package

Submodules

blendervr.plugins.osc.virtual_environment.base module

```
class blendervr.plugins.osc.virtual_environment.base.Base (parent, name, OSC_ID_1=None,
OSC_ID_2=None)
    Bases: blendervr.plugins.base.Base

    define_commands ()
    getAttribut (name)
    getID_1 ()
    getID_2 ()
    run ()
    runAttribut (attribut)
```

blendervr.plugins.osc.virtual_environment.object module

```
class blendervr.plugins.osc.virtual_environment.object.Object (parent, obj)
    Bases: blendervr.plugins.osc.virtual\_environment.base.Base

    getName ()
    getObject ()
    run ()
```

blendervr.plugins.osc.virtual_environment.objectuser module

```
class blendervr.plugins.osc.virtual_environment.objectuser.ObjectUser (parent, obj, usr)
    Bases: blendervr.plugins.osc.virtual\_environment.base.Base
```

blendervr.plugins.osc.virtual_environment.user module

```
class blendervr.plugins.osc.virtual_environment.user.User (parent, configuration, id)
    Bases: blendervr.plugins.osc.virtual\_environment.base.Base

    getName ()
    getUser ()
    run ()
```

Module contents

```
class blendervr.plugins.osc.virtual_environment.OSC (parent, configuration)
```

Bases: [blendervr.base.Base](#)

getGlobal ()

Return handle on OSC_global, a controller for high level sound control, defined in ./base.py

`getGlobal().start(True)`: send OSC message '/global/start 1'

`getGlobal().volume(30) / getGlobal().volume(-3)`: send OSC message '/global/volume 30' and '/global/volume -3'

`getGlobal().mute(False)`: send OSC message '/global/mute 0'

getObject (*obj*)

Instantiate/Access OSC_Object attached to a KX_GameObject, return OSC_Object

Parameters **obj** (*KX_GameObject*) – KX_GameObject in the scene

getObjectUser (*obj, usr*)

Instantiate OSC_ObjectUser, the ‘audio link’ between OSC_User and OSC_Object

Parameters

- **obj** (*OSC_User*) – OSC_Object
- **obj** – OSC_User

getUser (*usr*)

Instantiate OSC_User attached to a viewer, return OSC_User

Parameters **usr** (*string*) – listener, defined in the .xml configuration file**isAvailable** ()

Return true if the OSC client is available, else return false

reset ()**run** ()**sendCommand** (*cmd*)**start** ()**Submodules****blendervr.plugins.osc.client module****class** `blendervr.plugins.osc.client.Client` (*parent, host, port*)Bases: `blendervr.base.Base`**send** (*msg*)**blendervr.plugins.osc.exceptions module****exception** `blendervr.plugins.osc.exceptions.OSC_Invalid_Already_Created`Bases: `builtins.Exception`**exception** `blendervr.plugins.osc.exceptions.OSC_Invalid_Object`Bases: `builtins.Exception`**exception** `blendervr.plugins.osc.exceptions.OSC_Invalid_Type`Bases: `builtins.Exception`**blendervr.plugins.osc.msg module****class** `blendervr.plugins.osc.msg.MSG` (*parent, command*)Bases: `blendervr.plugins.base.Base`**append** (*argument*)**getBinary** ()`blendervr.plugins.osc.msg.getString` (*value*)**blendervr.plugins.osc.xml module****class** `blendervr.plugins.osc.xml.Room` (*parent, name, attrs*)Bases: `blendervr.console.xml.base.mono`**class** `blendervr.plugins.osc.xml.User` (*parent, name, attrs*)Bases: `blendervr.console.xml.base.mono`**class** `blendervr.plugins.osc.xml.XML` (*parent, name, attrs*)Bases: `blendervr.console.xml.base.XML`

Module contents

OSC Plugin This script instantiates the OSC plugin: an OSC API embedded in BlenderVR, used to send OSC messages (for sound control) to a third party program (e.g. PureData, MaxMSP, etc.)

```
class blendervr.plugins.osc.Base (parent)
    Bases: blendervr.base.Base

    setConfiguration (configuration)
    start ()
```

blendervr.plugins.vrpn package

Subpackages

blendervr.plugins.vrpn.virtual_environment package

Submodules

blendervr.plugins.vrpn.virtual_environment.analog module

blendervr.plugins.vrpn.virtual_environment.button module

blendervr.plugins.vrpn.virtual_environment.text module

blendervr.plugins.vrpn.virtual_environment.tracker module

Module contents

```
class blendervr.plugins.vrpn.virtual_environment.VRPN (parent, configuration)
    Bases: blendervr.base.Base

    checkMethods ()
    run ()
    start ()
```

blendervr.plugins.vrpn.xml package

Submodules

blendervr.plugins.vrpn.xml.analog module

```
class blendervr.plugins.vrpn.xml.analog.XML (parent, name, attrs)
    Bases: blendervr.plugins.vrpn.xml.vrpn_base.Receiver, blendervr.plugins.vrpn.xml.vrpn_base.Se
```

blendervr.plugins.vrpn.xml.button module**class** `blendervr.plugins.vrpn.xml.button.XML` (*parent, name, attrs*)Bases: `blendervr.plugins.vrpn.xml.vrpn_base.Receiver`, `blendervr.plugins.vrpn.xml.vrpn_base.XML`**blendervr.plugins.vrpn.xml.sensor module****class** `blendervr.plugins.vrpn.xml.sensor.XML` (*parent, name, attrs*)Bases: `blendervr.plugins.vrpn.xml.vrpn_base.Sender`**blendervr.plugins.vrpn.xml.text module****class** `blendervr.plugins.vrpn.xml.text.XML` (*parent, name, attrs*)Bases: `blendervr.plugins.vrpn.xml.vrpn_base.Receiver`, `blendervr.plugins.vrpn.xml.vrpn_base.XML`**blendervr.plugins.vrpn.xml.tracker module****class** `blendervr.plugins.vrpn.xml.tracker.XML` (*parent, name, attrs*)Bases: `blendervr.plugins.vrpn.xml.vrpn_base.Receiver`**blendervr.plugins.vrpn.xml.transformation module****class** `blendervr.plugins.vrpn.xml.transformation.XML` (*parent, name, attrs*)Bases: `blendervr.console.xml.reusable.XML`**getConfiguration()****blendervr.plugins.vrpn.xml.vrpn_base module****class** `blendervr.plugins.vrpn.xml.vrpn_base.Receiver` (*parent, name, attrs*)Bases: `blendervr.console.xml.base.XML`**class** `blendervr.plugins.vrpn.xml.vrpn_base.Sender` (*parent, name, attrs*)Bases: `blendervr.console.xml.base.XML`**Module contents****class** `blendervr.plugins.vrpn.xml.XML` (*parent, name, attrs*)Bases: `blendervr.console.xml.base.XML`**Module contents****class** `blendervr.plugins.vrpn.Base` (*parent*)Bases: `blendervr.base.Base`**setConfiguration** (*configuration*)**start()****Submodules****blendervr.plugins.base module****class** `blendervr.plugins.base.Base` (*parent*)Bases: `blendervr.plugins.base.Common`, `blendervr.player.base.Base`**isValid()****quit()****run()**

```
setConfiguration (configuration)
start ()

virtual_environment
class blendervr.plugins.base.Common
    Bases: builtins.object

    getName ()
```

blendervr.plugins.exceptions module**blendervr.plugins.xml module****Module contents****blendervr.plugins.getPlugins** (*parent, logger*)**blendervr.processor package****Submodules****blendervr.processor.base module**

```
class blendervr.processor.base.Processor (parent)
    Bases: blendervr.processor.base.ProcessorCommon

    keyboardAndMouse (info)
        receivedFromConsole (command, argument)
            Received command from the master node
        receivedFromMaster (command, argument)
        run ()
        sendToConsole (command, argument='')
            Send command to run in the console
        sendToSlaves (command, argument='')
            Send command to run in the slaves nodes
        setAsObjectToSynchronize (name)
        start ()
        user_position (info)
class blendervr.processor.base.ProcessorCommon (parent)
    Bases: blendervr.player.base.Base

    registerInteractor (interactor)
    unregisterInteractor (interactor)
```

blendervr.processor.default module

```
class blendervr.processor.default.Processor (parent)
    Bases: blendervr.processor.base.Processor
```

Module contents Handle all the processors, which are defined per project.

A processor is a python module that allows Blender-VR to run special code for specific project.

It can, for example, load OSC modules, or link the HMD to the scene camera.

`blendervr.processor.appendProcessor(processor_file)`

Include the processor file to be executed.

Parameters `processor_file` (*string*) – Python file with the process functions for the application

`blendervr.processor.getProcessor()`

Get the latest added Processor file.

Return type Processor

blendervr.tools package

Subpackages

blendervr.tools.gui package

Submodules

blendervr.tools.gui.qt module

```
class blendervr.tools.gui.qt.Common(owner, profile, profile_indices)
    Bases: builtins.object

    moveEvent(event)
    resizeEvent(event)
    setGeometry()
    start()

class blendervr.tools.gui.qt.Dialog(owner, profile, profile_indices)
    Bases: blendervr.tools.gui.qt.Common, dummy.Qt4.QDialog

    cb_close()
    cb_display_log_window()
    getMenu(name, shortcut)
    setMenuAction(menu_action)
    start()
```

Module contents

`blendervr.tools.gui.insertWidgetInsideAnother(parent, child)`

`blendervr.tools.gui.load(ui_file, parent_widget)`

Submodules

blendervr.tools.connector module

```
class blendervr.tools.connector.Client (controller, module, screen_name)
    Bases: blendervr.tools.connector.Common
class blendervr.tools.connector.Common
    Bases: builtins.object
        BUFFER_LEN = 1024
        SIZE_LEN = 10
        close()
        filenogetClientrunsend (command, argument='')
        setCallback (callback)
        setClient (client, callback=None)
        setWait (block)
class blendervr.tools.connector.Server (client)
    Bases: blendervr.tools.connector.Common
        getClientInformation()
```

blendervr.tools.logger module Handle all the errors, warnings and debug info

```
class blendervr.tools.logger.Console (msg='Console logger: ')
    Bases: builtins.object
        write (*messages)
class blendervr.tools.logger.Logger (name)
    Bases: logging.Logger
        addLoginWindow (login_window, addName=False)
        getVerbosities ()
        get_position ()
        log_position ()
        log_traceback (error)
        setLevel (verbosity)
    blendervr.tools.logger.getLogger (name)
```

blendervr.tools.profiler module**blendervr.tools.protocol module**

```
blendervr.tools.protocol.composeMessage (command, argument='')
blendervr.tools.protocol.decomposeMessage (message)
```

Module contents**blendervr.tools.getModulePath()**

The path of the current module

blendervr.tools.getRootPath()

The root path of the project

Submodules**blendervr.base module****class blendervr.base.Base (parent)**

Bases: builtins.object

Main struct for the blender vr environment

Parameters parent (module or class) –**getMainRunningModule()****Return type** module**getParent()**

Returns the module that inherits the Base class

Return type module or class**logger**

Logger of the main running module

Return type callback function**quit()****blendervr.exceptions module****exception blendervr.exceptions.PluginError (hasToClear=True, msg='')**

Bases: builtins.Exception

hasToClear()**Module contents**

Main module of the Blender-VR application

blendervr.is_console()

Check if it is in console mode.

Return type bool**blendervr.is_creating_loader()**

Check if BPY is available.

Return type bool**blendervr.is_virtual_environment()**

Check if the Blender Game Engine is available.

Return type bool**blendervr.main()**

Indices and tables

- *genindex*
- *modindex*
- *search*

b

blendervr, 36
blenderVR, 5
blendervr.base, 36
blendervr.console, 16
blendervr.console.base, 14
blendervr.console.console, 15
blendervr.console.exceptions, 15
blendervr.console.gui, 7
blendervr.console.gui.console, 6
blendervr.console.gui.logger, 6
blendervr.console.gui.options, 7
blendervr.console.gui.screen, 7
blendervr.console.gui.screens, 7
blendervr.console.logic, 9
blendervr.console.logic.console, 7
blendervr.console.logic.file_name, 8
blendervr.console.logic.screen, 8
blendervr.console.logic.screens, 8
blendervr.console.profile, 15
blendervr.console.qt, 11
blendervr.console.qt.console, 9
blendervr.console.qt.logger, 10
blendervr.console.qt.options, 10
blendervr.console.qt.screen, 10
blendervr.console.qt.screens, 11
blendervr.console.qt.tools, 11
blendervr.console.screen, 15
blendervr.console.screens, 15
blendervr.console.xml, 14
blendervr.console.xml.base, 12
blendervr.console.xml.behavior, 12
blendervr.console.xml.blendervr, 12
blendervr.console.xml.common_system, 13
blendervr.console.xml.computer, 13
blendervr.console.xml.device, 11
blendervr.console.xml.device.base, 11
blendervr.console.xml.device.hmd, 11
blendervr.console.xml.device.screen, 11
blendervr.console.xml.device.wall, 11

blendervr.console.xml.display, 13
blendervr.console.xml.plugins, 13
blendervr.console.xml.processors, 13
blendervr.console.xml.reusable, 13
blendervr.console.xml.root, 13
blendervr.console.xml.screen, 13
blendervr.console.xml.starter, 14
blendervr.console.xml.system, 14
blendervr.console.xml.user, 14
blendervr.exceptions, 36
blendervr.interactor, 18
blendervr.interactor.arc_ball, 16
blendervr.interactor.arc_ball.console, 16
blendervr.interactor.head_controlled_navigation, 16
blendervr.interactor.landmarks, 17
blendervr.interactor.laser, 17
blendervr.interactor.object_chooser, 17
blendervr.interactor.reset_objects, 17
blendervr.interactor.viewpoint, 17
blendervr.interactor.wavefront_obj, 18
blendervr.loader, 18
blendervr.loader.base, 18
blendervr.player, 27
blendervr.player.base, 25
blendervr.player.buffer, 25
blendervr.player.device, 26
blendervr.player.exceptions, 26
blendervr.player.keyboardAndMouse, 26
blendervr.player.network, 24
blendervr.player.network.connector, 24
blendervr.player.network.controller, 24
blendervr.player.network.synchronizer, 23
blendervr.player.network.synchronizer.objects, 23
blendervr.player.network.synchronizer.objects.item, 19
blendervr.player.network.synchronizer.objects.item, 19
blendervr.player.network.synchronizer.objects.item, 19

```
blendervr.player.network.synchronizer.objects.item_processor, 34
    20                      blendervr.processor.base, 33
blendervr.player.network.synchronizer.objects.item_processor.default, 33
    20                      blendervr.tools, 36
blendervr.player.network.synchronizer.objects.item_processor.connector, 35
    20                      blendervr.tools.gui, 34
blendervr.player.network.synchronizer.objects.item_processor.gui.qt, 34
    21                      blendervr.tools.logger, 35
blendervr.player.network.synchronizer.objects.item_processor.lighting, 35
    21                      blendervr.tools.protocol, 35
blendervr.player.network.synchronizer.objects.item_object,
    21                      d
blendervr.player.network.synchronizer.objects.item_root,
    22
blendervr.player.network.synchronizer.objects.item_scene,
    22                      update_loader, 5
blendervr.player.network.synchronizer.objects.master,
    22
blendervr.player.network.synchronizer.objects.slave,
    22
blendervr.player.screen, 25
blendervr.player.splash, 26
blendervr.player.user, 27
blendervr.plugins, 33
blendervr.plugins.base, 32
blendervr.plugins.exceptions, 33
blendervr.plugins.osc, 31
blendervr.plugins.osc.client, 30
blendervr.plugins.osc.exceptions, 30
blendervr.plugins.osc.msg, 30
blendervr.plugins.osc.virtual_environment,
    29
blendervr.plugins.osc.virtual_environment.base,
    29
blendervr.plugins.osc.virtual_environment.object,
    29
blendervr.plugins.osc.virtual_environment.object_user,
    29
blendervr.plugins.osc.virtual_environment.user,
    29
blendervr.plugins.osc.xml, 30
blendervr.plugins.vrpn, 32
blendervr.plugins.vrpn.virtual_environment,
    31
blendervr.plugins.vrpn.xml, 32
blendervr.plugins.vrpn.xml.analog, 31
blendervr.plugins.vrpn.xml.button, 32
blendervr.plugins.vrpn.xml.sensor, 32
blendervr.plugins.vrpn.xml.text, 32
blendervr.plugins.vrpn.xml.tracker, 32
blendervr.plugins.vrpn.xml.transformation,
    32
blendervr.plugins.vrpn.xml.vrpn_base, 32
blendervr.plugins.xml, 33
```

A

activate() (blendervr.player.network.synchronizer.objects.item_base.Master method), 20
activation() (blendervr.interactor.viewpoint.ViewPoint method), 17
adapt_simulation_files_to_screen() (blendervr.console.logic.screen.Logic method), 8
adapt_simulation_files_to_screen() (blendervr.console.logic.screens.Logic method), 8
addChildren() (blendervr.player.network.synchronizer.objects.item_base.Base method), 20
addChildren() (blendervr.player.network.synchronizer.objects.item_base.Master method), 20
addItem() (blendervr.player.network.synchronizer.objects.master.Master method), 22
addListenTo() (blendervr.console.qt.console.GUI method), 9
addLoginWindow() (blendervr.tools.logger.Logger method), 35
addObjectToSynchronize() (blendervr.player.Main method), 27
addObjectToSynchronize() (blendervr.player.network.synchronizer.Base method), 23
addPrefix() (blendervr.player.buffer.Buffer method), 25
addSynchronizedObject() (blendervr.player.network.synchronizer.Base method), 23
addTimeout() (blendervr.console.qt.console.GUI method), 9
allowDisallowObjects() (blendervr.interactor.laser.Laser method), 17
append() (blendervr.plugins.osc.msg.MSG method), 30
appendProcessor() (in module blendervr.processor), 34
appendValue() (blendervr.console.profile.Profile method), 15
ArcBall (class in blendervr.interactor.arc_ball), 16
ArmatureBone (class in blendervr.player.network.synchronizer.objects.item_armature_bone), 19
ArmatureChannel (class in blendervr.player.network.synchronizer.objects.item_armature_channel), 19
ArmatureObject (class in blendervr.player.network.synchronizer.objects.item_armature_object), 19
ask_blender_player_to_quit() (blendervr.console.logic.screen.Logic method), 8
attachLaserToCamera() (blendervr.interactor.laser.Laser method), 17

B

barrier() (blendervr.player.network.connector.Master method), 24
barrier() (blendervr.player.network.connector.Slave method), 24
Base (class in blendervr.base), 36
Base (class in blendervr.console.base), 14
Base (class in blendervr.console.xml.device.base), 11
Base (class in blendervr.loader.base), 18
Base (class in blendervr.player.base), 25
Base (class in blendervr.player.device), 26
Base (class in blendervr.player.network.synchronizer), 23
Base (class in blendervr.player.network.synchronizer.objects.item_base), 20
Base (class in blendervr.plugins.base), 32
Base (class in blendervr.plugins.osc), 31
Base (class in blendervr.plugins.osc.virtual_environment.base), 29
Base (class in blendervr.plugins.vrpn), 32
blenderplayer (class in blendervr.console.xml.system), 14
blenderVR (blendervr.player.base.Base attribute), 25
blenderVR (module), 5
blendervr (module), 36
blendervr.base (module), 36
blendervr.console (module), 16
blendervr.console.base (module), 14
blendervr.console.console (module), 15
blendervr.console.exceptions (module), 15
blendervr.console.gui (module), 7
blendervr.console.gui.console (module), 6
blendervr.console.gui.logger (module), 6
blendervr.console.gui.options (module), 7
blendervr.console.gui.screen (module), 7
blendervr.console.gui.screens (module), 7
blendervr.console.logic (module), 9

blendervr.console.logic.console (module), 7
blendervr.console.logic.file_name (module), 8
blendervr.console.logic.screen (module), 8
blendervr.console.logic.screens (module), 8
blendervr.console.profile (module), 15
blendervr.console.qt (module), 11
blendervr.console.qt.console (module), 9
blendervr.console.qt.logger (module), 10
blendervr.console.qt.options (module), 10
blendervr.console.qt.screen (module), 10
blendervr.console.qt.screens (module), 11
blendervr.console.qt.tools (module), 11
blendervr.console.screen (module), 15
blendervr.console.screens (module), 15
blendervr.console.xml (module), 14
blendervr.console.xml.base (module), 12
blendervr.console.xml.behavior (module), 12
blendervr.console.xml.blendervr (module), 12
blendervr.console.xml.common_system (module), 13
blendervr.console.xml.computer (module), 13
blendervr.console.xml.device (module), 11
blendervr.console.xml.device.base (module), 11
blendervr.console.xml.device.hmd (module), 11
blendervr.console.xml.device.screen (module), 11
blendervr.console.xml.device.wall (module), 11
blendervr.console.xml.display (module), 13
blendervr.console.xml.plugins (module), 13
blendervr.console.xml.processors (module), 13
blendervr.console.xml.reusable (module), 13
blendervr.console.xml.root (module), 13
blendervr.console.xml.screen (module), 13
blendervr.console.xml.starter (module), 14
blendervr.console.xml.system (module), 14
blendervr.console.xml.user (module), 14
blendervr.exceptions (module), 36
blendervr.interactor (module), 18
blendervr.interactor.arc_ball (module), 16
blendervr.interactor.arc_ball.console (module), 16
blendervr.interactor.head_controlled_navigation (module), 16
blendervr.interactor.landmarks (module), 17
blendervr.interactor.laser (module), 17
blendervr.interactor.object_chooser (module), 17
blendervr.interactor.reset_objects (module), 17
blendervr.interactor.viewpoint (module), 17
blendervr.interactor.wavefront_obj (module), 18
blendervr.loader (module), 18
blendervr.loader.base (module), 18
blendervr.player (module), 27
blendervr.player.base (module), 25
blendervr.player.buffer (module), 25
blendervr.player.device (module), 26
blendervr.player.exceptions (module), 26
blendervr.player.keyboardAndMouse (module), 26
blendervr.player.network (module), 24

blendervr.player.network.connector (module), 24
blendervr.player.network.controller (module), 24
blendervr.player.network.synchronizer (module), 23
blendervr.player.network.synchronizer.objects (module), 23
blendervr.player.network.synchronizer.objects.item_armature_bone (module), 19
blendervr.player.network.synchronizer.objects.item_armature_channel (module), 19
blendervr.player.network.synchronizer.objects.item_armature_object (module), 19
blendervr.player.network.synchronizer.objects.item_base (module), 20
blendervr.player.network.synchronizer.objects.item_camera (module), 20
blendervr.player.network.synchronizer.objects.item_default (module), 20
blendervr.player.network.synchronizer.objects.item_font (module), 21
blendervr.player.network.synchronizer.objects.item_light (module), 21
blendervr.player.network.synchronizer.objects.item_object (module), 21
blendervr.player.network.synchronizer.objects.item_root (module), 22
blendervr.player.network.synchronizer.objects.item_scene (module), 22
blendervr.player.network.synchronizer.objects.master (module), 22
blendervr.player.network.synchronizer.objects.slave (module), 22
blendervr.player.screen (module), 25
blendervr.player.splash (module), 26
blendervr.player.user (module), 27
blendervr.plugins (module), 33
blendervr.plugins.base (module), 32
blendervr.plugins.exceptions (module), 33
blendervr.plugins.osc (module), 31
blendervr.plugins.osc.client (module), 30
blendervr.plugins.osc.exceptions (module), 30
blendervr.plugins.osc.msg (module), 30
blendervr.plugins.osc.virtual_environment (module), 29
blendervr.plugins.osc.virtual_environment.base (module), 29
blendervr.plugins.osc.virtual_environment.object (module), 29
blendervr.plugins.osc.virtual_environment.objectuser (module), 29
blendervr.plugins.osc.virtual_environment.user (module), 29
blendervr.plugins.osc.xml (module), 30
blendervr.plugins.vrpn (module), 32
blendervr.plugins.vrpn.virtual_environment (module), 31
blendervr.plugins.vrpn.xml (module), 32
blendervr.plugins.vrpn.xml.analog (module), 31
blendervr.plugins.vrpn.xml.button (module), 32
blendervr.plugins.vrpn.xml.sensor (module), 32

blendervr.plugins.vrpn.xml.text (module), 32	cb_set_current_tab() (blendervr.console.qt.console.GUI method), 9
blendervr.plugins.vrpn.xml.tracker (module), 32	cb_set_link_processor_to_blender() (blendervr.console.qt.console.GUI method), 9
blendervr.plugins.vrpn.xml.transformation (module), 32	cb_set_log_level() (blendervr.console.qt.logger.Logger method), 10
blendervr.plugins.vrpn.xml.vrpn_base (module), 32	cb_set_processor_file() (blendervr.console.qt.console.GUI method), 9
blendervr.plugins.xml (module), 33	cb_set_screen_set() (blendervr.console.qt.console.GUI method), 9
blendervr.processor (module), 34	cb_start() (blendervr.console.qt.console.GUI method), 9
blendervr.processor.base (module), 33	cb_stop() (blendervr.console.qt.console.GUI method), 9
blendervr.processor.default (module), 33	cb_toggle_stderr_state() (blendervr.console.qt.screen.GUI method), 10
blendervr.tools (module), 36	cb_toggle_stdout_state() (blendervr.console.qt.screen.GUI method), 10
blendervr.tools.connector (module), 35	cb_update_liste_paths() (blendervr.console.qt.console.GUI method), 9
blendervr.tools.gui (module), 34	characters() (blendervr.console.xml.base.mono method), 12
blendervr.tools.gui.qt (module), 34	characters() (blendervr.console.xml.base.single method), 12
blendervr.tools.logger (module), 35	characters() (blendervr.console.xml.behavior.XML method), 12
blendervr.tools.profiler (module), 35	characters() (blendervr.console.xml.device.screen.Screen method), 11
blendervr.tools.protocol (module), 35	characters() (blendervr.console.xml.display.XML method), 13
blenderVR_state_changed() (blendervr.console.qt.options.GUI method), 10	characters() (blendervr.console.xml.processors.XML method), 13
boolean() (blendervr.player.buffer.Buffer method), 25	characters() (blendervr.console.xml.starter.XML method), 14
Buffer (class in blendervr.player.buffer), 25	characters() (blendervr.console.xml.system.blenderplayer method), 14
BUFFER_LEN (blendervr.tools.connector.Common attribute), 35	characters() (blendervr.console.xml.system.daemon method), 14
C	
CALIBRATE (blendervr.interactor.head_controlled_navigation.HCNav attribute), 16	checkItems() (blendervr.player.network.synchronizer.objects.item_base.Base method), 20
Camera (class in blendervr.player.network.synchronizer.objects.item_base), 20	checkItems() (blendervr.player.network.synchronizer.objects.master.Master method), 22
cb_add_configuration_path() (blendervr.console.qt.console.GUI method), 9	checkItems() (blendervr.player.network.synchronizer.objects.Synchronizer method), 23
cb_close() (blendervr.console.qt.console.GUI method), 9	checkMethod() (blendervr.player.device.Sender method), 26
cb_close() (blendervr.tools.gui.qt.Dialog method), 34	checkMethods() (blendervr.plugins.vrpn.virtual_environment.VRPN method), 31
cb_debug_daemon() (blendervr.console.qt.options.GUI method), 10	Chooser (class in blendervr.interactor.object_chooser), 17
cb_debug_processor() (blendervr.console.qt.options.GUI method), 10	clear() (blendervr.console.gui.logger.Logger method), 6
cb_display_log_window() (blendervr.tools.gui.qt.Dialog method), 34	clear() (blendervr.console.qt.logger.Logger method), 10
cb_executables() (blendervr.console.qt.options.GUI method), 10	click() (blendervr.interactor.arc_ball.ArcBall method), 16
cb_load_configuration_file() (blendervr.console.qt.console.GUI method), 9	Client (class in blendervr.plugins.osc.client), 30
cb_processor_window() (blendervr.console.qt.console.GUI method), 9	Client (class in blendervr.tools.connector), 35
cb_reload_processor() (blendervr.console.qt.options.GUI method), 10	close() (blendervr.console.qt.options.GUI method), 10
cb_remove_configuration_path()	close() (blendervr.console.qt.screen.GUI method), 10
(blendervr.console.qt.console.GUI method), 9	close() (blendervr.tools.connector.Common method), 35
cb_restart_daemons() (blendervr.console.qt.options.GUI method), 10	close_all() (blendervr.console.qt.screens.GUI method), 11
cb_set_blender_file() (blendervr.console.qt.console.GUI method), 9	CMD_FINISHED (blendervr.player.network.connector.Connector attribute), 24
cb_set_configuration_file() (blendervr.console.qt.console.GUI method), 9	

CMD_MSG (blendervr.player.network.connector.Connector attribute), 24
CMD_SYNCHRO (blendervr.player.network.connector.Connect display_screen_sets() (blendervr.console.gui.console.GUI method), 6
attribute), 24 display_screen_sets() (blendervr.console.qt.console.GUI method), 9
command() (blendervr.player.buffer.Buffer method), 25 drag() (blendervr.interactor.arc_ball.ArcBall method), 16
Common, 26 draw() (blendervr.interactor.wavefront_obj.Reader method), 18
Common (class in blendervr.plugins.base), 33 dump() (blendervr.console.profile.Profile method), 15
Common (class in blendervr.tools.connector), 35
Common (class in blendervr.tools.gui.qt), 34
compile_BC() (blendervr.console.logic.console.Logic method), 7
composeMessage() (in module blendervr.tools.protocol), 35
Configure (class in blendervr.console.xml), 14
Connector (class in blendervr.player.network.connector), 24
Console (class in blendervr.console.console), 15
Console (class in blendervr.interactor.arc_ball.console), 16
Console (class in blendervr.tools.logger), 35
container (class in blendervr.console.xml.computer), 13
container (class in blendervr.console.xml.screen), 13
container (class in blendervr.console.xml.user), 14
Controller, 26
Controller (class in blendervr.player.network.controller), 24
CREATE_ITEM (blendervr.player.network.synchronizer.objects attribute), 23
Creator (class in blendervr.loader), 18

D

daemon (class in blendervr.console.xml.system), 14
Daemon (class in daemon), 5
daemon (module), 5
daemon_is_running() (blendervr.console.logic.screen.Logic method), 8
data_size() (in module blendervr.player.buffer), 25
decomposeMessage() (in module blendervr.tools.protocol), 35
Default (class in blendervr.player.network.synchronizer.objects.item_default), 20
default() (blendervr.player.network.synchronizer.objects.item_base.Base method), 20
default() (blendervr.player.network.synchronizer.objects.item_object), 21
default() (blendervr.player.network.synchronizer.objects.item_root.Root method), 22
default() (blendervr.player.network.synchronizer.objects.item_scene.Scene method), 22
define_commands() (blendervr.plugins.osc.virtual_environment.base method), 29
DEL_OBJECT (blendervr.player.network.synchronizer.Base attribute), 23
DELETE_ITEM (blendervr.player.network.synchronizer.objects attribute), 23
Device (class in blendervr.console.xml.device.hmd), 11
Device (class in blendervr.console.xml.device.wall), 11
Dialog (class in blendervr.tools.gui.qt), 34
display_laser() (blendervr.interactor.laser.Laser method), 17

E

END_UPDATE_ITEM (blendervr.player.network.synchronizer.objects.Synch attribute), 23
endElement() (blendervr.console.xml.base.mono method), 12
endElement() (blendervr.console.xml.base.single method), 12
endElement() (blendervr.console.xml.base.XML method), 12
endElement() (blendervr.console.xml.behavior.XML method), 12
endElement() (blendervr.console.xml.device.Screen method), 11
endElement() (blendervr.console.xml.display.XML method), 13
endFrame() (blendervr.player.network.connector.MasterSync method), 24
endFrame() (blendervr.player.network.connector.Slave method), 24

F

FileName (class in blendervr.console.logic.file_name), 8
fileno() (blendervr.tools.connector.Common method), 35
float() (blendervr.player.buffer.Buffer method), 25
flush() (blendervr.console.gui.logger.Logger method), 6
flush() (blendervr.player.network.controller.Controller method), 24
Font (class in blendervr.player.network.synchronizer.objects.item_font), 11
FRAME (blendervr.player.network.synchronizer.objects.item_armature_obj)
get(Object
 get.blender_player_state() (blendervr.console.logic.console.Logic method), 7
 get.blender_player_state() (blendervr.console.logic.screen.Logic method), 8
 get.position() (blendervr.tools.logger.Logger method), 35
 getBaseUsers() (blendervr.player.Main method), 27
 getAttribut() (blendervr.plugins.osc.virtual_environment.base.Base method), 29
 getBinary() (blendervr.plugins.osc.msg.MSG method), 30
 getBOLEvent() (blendervr.console.xml.base.XML method), 12
 getClient() (blendervr.tools.connector.Common method), 35
 getClientInformation() (blendervr.tools.connector.Server method), 35
 getColors() (blendervr.console.qt.tools.MessagesColors method), 11

getConfiguration() (blendervr.console.xml.base.XML method), 12
 getConfiguration() (blendervr.console.xml.Configure method), 14
 getConfiguration() (blendervr.console.xml.processors.XML method), 13
 getConfiguration() (blendervr.console.xml.root.XML method), 13
 getConfiguration() (blendervr.player.network.controller.Controller method), 24
 getConfiguration() (blendervr.plugins.vrpn.xml.transformation.XML method), 32
 getConsole() (blendervr.console.base.Base method), 14
 getController() (blendervr.player.Main method), 27
 getDevice() (in module blendervr.console.xml.device), 11
 getEyeSeparation() (blendervr.player.user.User method), 27
 setFrame() (blendervr.console.gui.screen.GUI method), 7
 getGlobal() (blendervr.plugins.osc.virtual_environment.OSC method), 29
 getHitObject() (blendervr.interactor.laser.Laser method), 17
 getHostname() (blendervr.console.logic.screen.Logic method), 8
 getID() (blendervr.player.user.User method), 27
 getID_1() (blendervr.plugins.osc.virtual_environment.base.Base method), 29
 getID_2() (blendervr.plugins.osc.virtual_environment.base.Base method), 29
 getItem() (blendervr.player.network.synchronizer.objects.SynchrgntScale() (blendervr.player.Main method), 27
 getItemByName() (blendervr.player.network.synchronizer.objects.item_base.Base method), 20
 getItemByName() (blendervr.player.network.synchronizer.objects.item_base.Base method), 22
 getItemID() (blendervr.player.network.synchronizer.objects.item_base.Base method), 20
 getItemID() (blendervr.player.network.synchronizer.objects.item_base.Base method), 22
 getLogger() (in module blendervr.tools.logger), 35
 getMain() (blendervr.console.xml.base.XML method), 12
 getMainRunningModule() (blendervr.base.Base method), 36
 getMaster() (blendervr.console.logic.screens.Logic method), 8
 getMatrixFromQuaternion() (blendervr.interactor.arc_ball.ArcBall method), 16
 getMenu() (blendervr.console.qt.options.GUI method), 10
 getMenu() (blendervr.console.qt.options.GUI method), 10
 getMenu() (blendervr.console.qt.screen.GUI method), 10
 getMenu() (blendervr.tools.gui.qt.Dialog method), 34
 getMinMax() (blendervr.interactor.wavefront_obj.Reader method), 18
 getModulePath() (in module blendervr.tools), 36
 getName() (blendervr.console.screen.Screen method), 15
 getName() (blendervr.player.user.User method), 27
 getName() (blendervr.plugins.base.Common method), 33
 getName() (blendervr.plugins.osc.virtual_environment.object.Object method), 29
 getName() (blendervr.plugins.osc.virtual_environment.user.User method), 29
 getNavigationState() (blendervr.interactor.head_controlled_navigation.HCN method), 16
 getNetworks() (in module blendervr.player.network), 24
 getNumberOfNodes() (blendervr.player.Main method), 27
 getObject() (blendervr.interactor.laser.Laser method), 17
 getObject() (blendervr.plugins.osc.virtual_environment.object.Object method), 29
 getObject() (blendervr.plugins.osc.virtual_environment.OSC method), 29
 getObjectByID() (blendervr.player.network.synchronizer.Base method), 23
 getObjectByMasterID() (blendervr.player.network.synchronizer.objects.slav method), 22
 getObjectUser() (blendervr.plugins.osc.virtual_environment.OSC method), 29
 getParent() (blendervr.base.Base method), 36
 getParser() (blendervr.console.xml.base.XML method), 12
 getPlugin() (blendervr.player.Main method), 27
 getPlugins() (in module blendervr.plugins), 33
 getPosition() (blendervr.player.user.User method), 27
 getProcessor() (blendervr.player.Main method), 27
 getProcessor() (in module blendervr.processor), 34
 getRootPath() (in module blendervr.tools), 36
 getSceneSynchronizer() (blendervr.player.Main method), 28
 getScreenSynchronizer() (blendervr.player.network.synchronizer.Base method), 23
 getScreen() (blendervr.console.logic.screens.Logic method), 8
 getScreen() (blendervr.player.keyboardAndMouse.KeyboardAndMouse method), 26
 getScreen() (in module blendervr.player.screen), 25
 getScreenName() (blendervr.player.Main method), 28
 getScreensNumber() (blendervr.console.logic.screens.Logic method), 8
 getScreenUsers() (blendervr.player.Main method), 28
 getStates() (blendervr.console.logic.screens.Logic method), 8
 getString() (in module blendervr.plugins.osc.msg), 30
 getSyncronizerBuffer() (blendervr.interactor.laser.Laser method), 17
 getSyncronizerBuffer() (blendervr.player.network.synchronizer.objects.item method), 19
 getSyncronizerBuffer() (blendervr.player.network.synchronizer.objects.item method), 19
 getSyncronizerBuffer() (blendervr.player.network.synchronizer.objects.item method), 19
 getSyncronizerBuffer() (blendervr.player.network.synchronizer.objects.item method), 20
 getSyncronizerBuffer() (blendervr.player.network.synchronizer.objects.item method), 21

getSynchronizerBuffer() (blendervr.player.network.synchronizer.objects.item_base.[itemSyncable](#) method) (in module blendervr), 36
 method), 22
 getSynchronizerBuffer() (blendervr.player.network.synchronizer.objects.[master](#))¹⁷
 method), 22
 getSynchronizerBuffer() (blendervr.player.user.User method),
 27
 getUser() (blendervr.plugins.osc.virtual_environment.OSC
 method), 30
 getUser() (blendervr.plugins.osc.virtual_environment.user.User
 method), 29
 getUserByName() (blendervr.player.Main method), 28
 getValue() (blendervr.console.profile.Profile method), 15
 getVector() (blendervr.console.xml.base.XML method), 12
 getVehiclePosition() (blendervr.player.user.User method), 27
 getVerbosities() (blendervr.tools.logger.Logger method), 35
 getVersion() (blendervr.player.Main method), 28
 getWindow() (blendervr.console.qt.console.GUI method), 9
 getXML_FileName() (blendervr.console.xml.base.XML
 method), 12
 getXML_LineNumber() (blendervr.console.xml.base.XML
 method), 12
 getXML_Position() (blendervr.console.xml.base.XML
 method), 12
 grab() (blendervr.interactor.laser.Laser method), 17
 graphic_buffer (class in bladervr.console.xml.display), 13
 GUI (class in bladervr.console.gui.console), 6
 GUI (class in bladervr.console.gui.options), 7
 GUI (class in bladervr.console.gui.screen), 7
 GUI (class in bladervr.console.gui.screens), 7
 GUI (class in bladervr.console.qt.console), 9
 GUI (class in bladervr.console.qt.options), 10
 GUI (class in bladervr.console.qt.screen), 10
 GUI (class in bladervr.console.qt.screens), 11

H

hasToClear() (bladervr.exceptions.PluginError method), 36
 HNav (class in bladervr.interactor.head_controlled_navigation)
 16

I

insertWidgetInsideAnother() (in module bladervr.tools.gui),
 34
 integer() (bladervr.player.buffer.Buffer method), 25
 Interactor (class in bladervr.interactor), 18
 is_console() (in module bladervr), 36
 is_creating_loader() (in module bladervr), 36
 is_log_window_opened() (bladervr.console.gui.screen.GUI
 method), 7
 is_log_window_opened() (bladervr.console.qt.screen.GUI
 method), 10
 is_master() (bladervr.console.logic.screen.Logic method), 8
 is_options_window_opened()
 (bladervr.console.gui.options.GUI
 method),
 7

isActivated() (bladervr.interactor.viewpoint.ViewPoint
 method), 17
 isAvailable() (bladervr.plugins.osc.virtual_environment.OSC
 method), 30
 isEmpty() (bladervr.player.buffer.Buffer method), 25
 isMaster() (bladervr.player.Main method), 28
 isPaused() (bladervr.player.Main method), 28
 isRunning() (bladervr.player.splash.Splash method), 26
 isSynchronizable() (bladervr.player.network.synchronizer.objects.item_base.[itemSyncable](#)
 method), 20
 isSynchronizable() (bladervr.player.network.synchronizer.objects.item_def
 method), 20
 isValid() (bladervr.plugins.base.Base method), 32
 itemID() (bladervr.player.buffer.Buffer method), 25

K

KeyboardAndMouse (class in bladervr.player.keyboardAndMouse), 26
 keyboardAndMouse() (bladervr.interactor.Interactor
 method), 18
 keyboardAndMouse() (bladervr.interactor.viewpoint.ViewPoint
 method), 17
 keyboardAndMouse() (bladervr.processor.base.Processor
 method), 33

L

LandMarks (class in bladervr.interactor.landmarks), 17
 Laser (class in bladervr.interactor.laser), 17
 Light (class in bladervr.player.network.synchronizer.objects.item_light),
 21
 load() (in module bladervr.tools.gui), 34
 load_configuration_file() (bladervr.console.logic.console.Logic
 method), 7
 localTransform (bladervr.player.user.User attribute), 27
 lock() (bladervr.console.profile.Profile method), 15
 log (class in bladervr.console.xml.system), 14
 log_position() (bladervr.tools.logger.Logger method), 35
 log_traceback() (bladervr.tools.logger.Logger method), 35
 logger (bladervr.base.Base attribute), 36
 logger (bladervr.console.console.Console attribute), 15
 logger (bladervr.console.screen.Screen attribute), 15
 logger (bladervr.player.Main attribute), 28
 Logger (class in bladervr.console.gui.logger), 6
 Logger (class in bladervr.console.qt.logger), 10
 Logger (class in bladervr.tools.logger), 35
 Logic (class in bladervr.console.logic.console), 7
 Logic (class in bladervr.console.logic.screen), 8
 Logic (class in bladervr.console.logic.screens), 8
 login (class in bladervr.console.xml.system), 14

M

Main, 15, 26
 main (class in bladervr.console.xml.computer), 13

main (class in `blendervr.console.xml.screen`), 13
 main (class in `blendervr.console.xml.user`), 14
 Main (class in `blendervr.player`), 27
 main() (`blendervr.console.gui.console`.GUI method), 6
 main() (`blendervr.console.qt.console`.GUI method), 9
 main() (daemon.Daemon method), 5
 main() (in module `blenderVR`), 6
 main() (in module `blendervr`), 36
 main() (in module `blendervr.console`), 16
 main() (in module `blendervr.loader`), 18
 main() (in module `blendervr.loader.base`), 18
 main() (in module `daemon`), 5
 main_logger (`blendervr.console.screen`.Screen attribute), 15
 MainWindow (class in `blendervr.console.qt.console`), 10
 Master (class in `blendervr.player.network.connector`), 24
 Master (class in `blendervr.player.network.synchronizer`), 23
 Master (class in `blendervr.player.network.synchronizer.objects.item_armature_base`), 19
 Master (class in `blendervr.player.network.synchronizer.objects.item_object`), 19
 Master (class in `blendervr.player.network.synchronizer.objects.item_armature_channel`),
 19
 Master (class in `blendervr.player.network.synchronizer.objects.item_armature_object`),
 19
 Master (class in `blendervr.player.network.synchronizer.objects.item_base`), 20
 Master (class in `blendervr.player.network.synchronizer.objects.item_camera`), 20
 Master (class in `blendervr.player.network.synchronizer.objects.item_default`), 21
 Master (class in `blendervr.player.network.synchronizer.objects.item_font`), 21
 Master (class in `blendervr.player.network.synchronizer.objects.item_light`), 21
 Master (class in `blendervr.player.network.synchronizer.objects.item_object`), 21
 Master (class in `blendervr.player.network.synchronizer.objects.item_root`), 22
 Master (class in `blendervr.player.network.synchronizer.objects.item_scene`), 22
 Master (class in `blendervr.player.network.synchronizer.objects.master`), 22
 matrix_3x3() (`blendervr.player.buffer`.Buffer method), 25
 matrix_4x4() (`blendervr.player.buffer`.Buffer method), 25
 MESSAGE_PAUSE (`blendervr.player.Main` attribute), 27
 MESSAGE_PROCESSOR (`blendervr.player.Main` attribute), 27
 MessagesColors (class in `blendervr.console.qt.tools`), 11
 mono (class in `blendervr.console.xml.base`), 12
 moveEvent() (`blendervr.tools.gui.qt`.Common method), 34
 MSG (class in `blendervr.plugins.osc.msg`), 30

N

NEW_OBJECT (`blendervr.player.network.synchronizer.Base`
 attribute), 23
 NotExistingItem, 20

O

OBJECT (`blendervr.player.network.synchronizer.Base` at-
 tribute), 23
 OBJECT (`blendervr.player.network.synchronizer.objects.item_armature_obj`
 attribute), 19
 Object (class in `blendervr.player.network.synchronizer.objects.item_object`),
 21
 Object (class in `blendervr.plugins.osc.virtual_environment.object`),
 29
 ObjectUser (class in `blendervr.plugins.osc.virtual_environment.objectuser`),
 29
 ORIENTATION (`blendervr.player.network.synchronizer.objects.item_objec`
 attribute), 21
 OSC (class in `blendervr.plugins.osc.virtual_environment`), 29
 OSC_Invalid_Already_Created, 30
 OSC_Invalid_Object, 30
 OSC_Invalid_Type, 30
 POSITION (`blendervr.player.network.synchronizer.objects.item_object.Obj`
 attribute), 21
 pause() (`blendervr.player.Main` method), 28
 PluginError, 36
 plugins (`blendervr.console.console`.Console attribute), 15
 POSITION (`blendervr.player.network.synchronizer.objects.item_object.Obj`
 attribute), 21
 prependValue() (`blendervr.console.profile`.Profile method), 15
 print_warning() (`blendervr.console.xml.base.XML` method),
 12
 process() (`blendervr.loader.base`.Base method), 18
 process() (`blendervr.loader.Creator` method), 18
 process() (`blendervr.player.device.Sender` method), 26
 process() (`blendervr.player.network.synchronizer.Slave`
 method), 23
 processCommand() (daemon.Daemon method), 5
 Processor, 26
 Processor (`class in blendervr.processor.base`), 33
 Processor (`class in blendervr.processor.default`), 33
 Processor_Invalid_Device, 26
 Processor_Invalid_Device_Method, 26
 ProcessorCommon (`class in blendervr.processor.base`), 33
 processSynchronizerBuffer() (`blendervr.interactor.laser.Laser`
 method), 17
 processSynchronizerBuffer() (`blendervr.player.network.synchronizer.object`
 method), 19
 processSynchronizerBuffer() (`blendervr.player.network.synchronizer.object`
 method), 19
 processSynchronizerBuffer() (`blendervr.player.network.synchronizer.object`
 method), 20
 processSynchronizerBuffer() (`blendervr.player.network.synchronizer.object`
 method), 20
 processSynchronizerBuffer() (`blendervr.player.network.synchronizer.object`
 method), 21
 processSynchronizerBuffer() (`blendervr.player.network.synchronizer.object`
 method), 22

processSynchronizerBuffer() (blendervr.player.network.synchronizer.ReceiveObjectInSlaveInBlendervrPlayer.device), 26
 method), 22

processSynchronizerBuffer() (blendervr.player.network.synchronizer.ReceiveObjectInSlaveInBlendervrPlayer.device), 26
 method), 32

processSynchronizerBuffer() (blendervr.player.user.User method), 22

processSynchronizerBuffer() (blendervr.player.user.User method), 27

profile (blendervr.console.base.Base attribute), 14

profile (blendervr.console.console.Console attribute), 15

Profile (class in blendervr.console.profile), 15

Q

quit() (blendervr.base.Base method), 36

quit() (blendervr.console.base.Base method), 14

quit() (blendervr.console.console.Console method), 15

quit() (blendervr.console.gui.console.GUI method), 6

quit() (blendervr.console.gui.logger.Logger method), 6

quit() (blendervr.console.gui.options.GUI method), 7

quit() (blendervr.console.gui.screen.GUI method), 7

quit() (blendervr.console.gui.screens.GUI method), 7

quit() (blendervr.console.logic.console.Logic method), 7

quit() (blendervr.console.logic.screen.Logic method), 8

quit() (blendervr.console.logic.screens.Logic method), 8

quit() (blendervr.console.qt.console.GUI method), 9

quit() (blendervr.console.qt.options.GUI method), 10

quit() (blendervr.console.qt.screen.GUI method), 10

quit() (blendervr.console.qt.screens.GUI method), 11

quit() (blendervr.console.screen.Screen method), 15

quit() (blendervr.console.screens.Screens method), 15

quit() (blendervr.player.Main method), 28

quit() (blendervr.player.network.connector.Master method), 24

quit() (blendervr.player.network.connector.Slave method), 24

quit() (blendervr.plugins.base.Base method), 32

quit() (in module blendervr.console.qt.console), 10

R

raise_error() (blendervr.console.xml.base.XML method), 12

Reader (class in blendervr.interactor.wavefront_obj), 18

receivedFromConsole() (blendervr.interactor.arc_ball.console.Console method), 16

receivedFromConsole() (blendervr.interactor.head_controlled_navigation.HCNav method), 16

receivedFromConsole() (blendervr.interactor.Interactor method), 18

receivedFromConsole() (blendervr.interactor.landmarks.LandMarks method), 17

receivedFromConsole() (blendervr.processor.base.Processor method), 33

receivedFromMaster() (blendervr.processor.base.Processor method), 33

receivedFromVirtualEnvironment()
 (blendervr.console.logic.console.Logic method), 7

receiveFrom() (blendervr.player.network.connector.Connector method), 24

receiveObjectInSlaveInBlendervrPlayer.device, 26

Receiver (class in blendervr.plugins.vrpn.xml.vrpn_base), 32

registerInteraction() (blendervr.processor.base.ProcessorCommon method), 33

release() (blendervr.interactor.laser.Laser method), 17

remove() (blendervr.player.network.synchronizer.objects.item_base.Base method), 20

remove() (blendervr.player.network.synchronizer.objects.item_base.Master method), 20

removeChildren() (blendervr.player.network.synchronizer.objects.item_base method), 20

removeItem() (blendervr.player.network.synchronizer.objects.master.Master method), 22

removeListenTo() (blendervr.console.qt.console.GUI method), 9

removeScale() (in module blendervr.interactor.arc_ball), 16

removeSynchronizedItem() (blendervr.player.network.synchronizer.objects.item_base method), 23

removeSynchronizedObject()
 (blendervr.player.network.synchronizer.Base method), 23

RESET (blendervr.interactor.head_controlled_navigation.HCNav attribute), 16

reset() (blendervr.interactor.reset_objects.ResetObjects method), 17

reset() (blendervr.plugins.osc.virtual_environment.OSC method), 30

ResetObjects (class in blendervr.interactor.reset_objects), 17

resetVehiclePosition() (blendervr.player.user.User method), 27

resizeEvent() (blendervr.tools.gui.qt.Common method), 34

resolveEntity() (blendervr.console.xml.Configure method), 14

restartDaemon() (blendervr.console.logic.screen.Logic method), 8

Room (class in blendervr.plugins.osc.xml), 30

Root (class in blendervr.player.network.synchronizer.objects.item_root), 22

run() (blendervr.interactor.Interactor method), 18

run() (blendervr.interactor.viewpoint.ViewPoint method), 18

run() (blendervr.player.device.Receiver method), 26

run() (blendervr.player.keyboardAndMouse.KeyboardAndMouse method), 26

run() (blendervr.player.network.connector.Master method), 24

run() (blendervr.player.network.connector.Slave method), 24

run() (blendervr.player.network.controller.Controller method), 24

run() (blendervr.player.network.synchronizer.Base method), 23

run() (blendervr.plugins.base.Base method), 32

run() (blendervr.plugins.osc.virtual_environment.base.Base method), 29

run() (blendervr.plugins.osc.virtual_environment.object.Object method), 29

run()	(blendervr.plugins.osc.virtual_environment.OSC method), 30	sendToVirtualEnvironment() (blendervr.console.logic.console.Logic method), 7
run()	(blendervr.plugins.osc.virtual_environment.user.User method), 29	Server (class in blendervr.tools.connector), 35
run()	(blendervr.plugins.vrpn.virtual_environment.VRPN method), 31	SET_ATTRIBUTE (blendervr.player.network.synchronizer.objects.Synchronizer attribute), 23
run()	(blendervr.processor.base.Processor method), 33	set_blenderVR_state() (blendervr.console.logic.screen.Logic method), 8
run()	(blendervr.tools.connector.Common method), 35	set_log_level() (blendervr.console.gui.logger.Logger method), 6
runAttribut()	(blendervr.plugins.osc.virtual_environment.base.Base method), 29	set_screen_set() (blendervr.console.logic.console.Logic method), 7
S		
save()	(blendervr.interactor.reset_objects.ResetObjects method), 17	set_screens() (blendervr.console.logic.screens.Logic method), 8
SCALE	(blendervr.player.network.synchronizer.objects.item_object.Object attribute), 21	set_stream_state() (blendervr.console.gui.logger.Logger method), 6
Scene	(class in blendervr.player.network.synchronizer.objects.item_scene), 22	setAsObjectToSynchronize() (blendervr.processor.base.Processor method), 33
sceneIsSplash()	(blendervr.player.splash.Splash method), 26	setBounds() (blendervr.interactor.arc_ball.ArcBall method), 16
Screen	(class in blendervr.console.screen), 15	setCallback() (blendervr.tools.connector.Common method), 35
Screen	(class in blendervr.console.xml.device.screen), 11	setClient() (blendervr.tools.connector.Common method), 35
Screens	(class in blendervr.console.screens), 15	setConfiguration() (blendervr.console.logic.screen.Logic method), 8
selectObject()	(blendervr.interactor.arc_ball.console.Console method), 16	setConfiguration() (blendervr.plugins.base.Base method), 32
selectSocket()	(blendervr.player.network.connector.Connector method), 24	setConfiguration() (blendervr.plugins.osc.Base method), 31
send()	(blendervr.player.network.connector.Master method), 24	setConfiguration() (blendervr.plugins.vrpn.Base method), 32
send()	(blendervr.plugins.osc.client.Client method), 30	setDefault() (blendervr.console.profile.Profile method), 15
send()	(blendervr.tools.connector.Common method), 35	setDefaultUser() (blendervr.interactor.head_controlled_navigation.HCNav method), 16
send_to_blender_player()	(blendervr.console.logic.screen.Logic method), 8	setGeometry() (blendervr.tools.gui.qt.Common method), 34
send_to_blender_player()	(blendervr.console.logic.screens.Logic method), 8	setHeadLocation() (blendervr.interactor.head_controlled_navigation.HCNav method), 17
sendCommand()	(blendervr.plugins.osc.virtual_environment.OSC method), 30	setHeadNeckLocation() (blendervr.interactor.head_controlled_navigation.HCNav method), 17
Sender	(class in blendervr.player.device), 26	setHierarchy() (blendervr.console.logic.screen.Logic method), 8
Sender	(class in blendervr.plugins.vrpn.xml.vrpn_base), 32	setLevel() (blendervr.tools.logger.Logger method), 35
sendItemsUpdateToSlaves()	(blendervr.player.network.synchronizer.objects.Master method), 22	setMenuAction() (blendervr.tools.gui.qt.Dialog method), 34
sendSynchronization()	(blendervr.player.network.synchronizer.Master method), 23	setMessage() (blendervr.player.splash.Splash method), 26
sendTo()	(blendervr.player.network.connector.Connector method), 24	setNetworkClient() (blendervr.console.logic.screen.Logic method), 8
sendToConsole()	(blendervr.interactor.Interactor method), 18	setOrientation() (blendervr.interactor.arc_ball.ArcBall method), 16
sendToConsole()	(blendervr.player.network.controller.Controller method), 24	setOrientationFactors() (blendervr.interactor.head_controlled_navigation.HCNav method), 17
sendToConsole()	(blendervr.processor.base.Processor method), 33	setParent() (blendervr.player.user.User method), 27
sendToSlave()	(blendervr.player.network.connector.Master method), 24	setPosition() (blendervr.player.user.User method), 27
sendToSlaves()	(blendervr.processor.base.Processor method), 33	setPositionFactors() (blendervr.interactor.head_controlled_navigation.HCNav method), 17
		setValue() (blendervr.console.profile.Profile method), 15
		setVehiclePosition() (blendervr.player.user.User method), 27
		setWait() (blendervr.tools.connector.Common method), 35
		show() (blendervr.console.gui.screens.GUI method), 7

single (class in `blendervr.console.xml.base`), 12
`size()` (`blendervr.player.buffer` method), 25
`SIZE_LEN` (`blendervr.tools.connector.Common` attribute), 35
`Slave` (class in `blendervr.player.network.connector`), 24
`Slave` (class in `blendervr.player.network.synchronizer`), 23
`Slave` (class in `blendervr.player.network.synchronizer.objects.item_base`), 19
`Slave` (class in `blendervr.player.network.synchronizer.objects.item_base`), 20
`Slave` (class in `blendervr.player.network.synchronizer.objects.item_base`), 20
`Slave` (class in `blendervr.player.network.synchronizer.objects.item_base`), 21
`start()` (`blendervr.player.network.synchronizer.objects.slave.Slave` method), 22
`start()` (`blendervr.player.network.synchronizer.objects.Synchronizer` method), 23
`start()` (`blendervr.player.splash.Splash` method), 27
`start()` (`blendervr.plugins.base.Base` method), 33
`start()` (`blendervr.plugins.osc.Base` method), 31
`start()` (`blendervr.plugins.osc.virtual_environment.OSC` method), 30
`start()` (`blendervr.plugins.vrpn.Base` method), 32
`start()` (`blendervr.plugins.vrpn.virtual_environment.VRPN` method), 31
`start()` (`blendervr.processor.base.Processor` method), 33
`start()` (`blendervr.tools.gui.qt.Common` method), 34
`start()` (`blendervr.tools.gui.qt.Dialog` method), 34
`start()` (`blendervr.console.logic.console.Logic` method), 7
`start()` (`blendervr.console.logic.screens.Logic` method), 9
`start()` (`blendervr.console.xml.base.mono` method), 12
`startElement()` (`blendervr.console.xml.base.single` method), 12
`startElement()` (`blendervr.console.xml.base.XML` method), 12
`startSimulation()` (`blendervr.player.network.controller.Controller` method), 24
`STOP` (`blendervr.interactor.head_controlled_navigation.HCNav` method), 24
`stop()` (`blendervr.player.splash.Splash` method), 27
`stop_simulation()` (`blendervr.console.logic.console.Logic` method), 8
`stop_simulation()` (`blendervr.console.logic.screens.Logic` method), 9
`stopDueToError()` (`blendervr.player.Main` method), 28
`string()` (`blendervr.player.buffer.Buffer` method), 25
`strip()` (`blendervr.console.logic.fileName` method), 8
`stripAnchor()` (in module `blendervr.console`), 16
`subBuffer()` (`blendervr.player.buffer.Buffer` method), 25
`Synchronizer`, 26
`Synchronizer` (class in `blendervr.player.network.synchronizer.objects`), 23
`SYNCHRONIZER_COMMAND_USER_POSITION` (`blendervr.player.user.User` attribute), 27
`SYNCHRONIZER_COMMAND_VEHICLE_POSITION` (`blendervr.player.user.User` attribute), 27

T

`TOGGLE` (`blendervr.interactor.head_controlled_navigation.HCNav` attribute), 16
`toggle()` (`blendervr.interactor.laser.Laser` method), 17

U

`unregisterInteractor()` (`blendervr.processor.base.ProcessorCommon` method), 33
`unsigned_char()` (`blendervr.player.buffer.Buffer` method), 25

unstrip() (blendervr.console.logic.FileName method), 8
 unstripAnchor() (in module blendervr.console), 16
 update() (blendervr.interactor.head_controlled_navigation.HCNaXML (class in blendervr.console.xml.display), 13
 method), 17
 update_gui() (blendervr.console.gui.screens.GUI method), 7
 update_gui() (blendervr.console.qt.screens.GUI method), 11
 UPDATE_ITEM (blendervr.player.network.synchronizer.objects.XMLTechnik in blendervr.console.xml.root), 13
 attribute), 23
 update_loader (module), 5
 update_processor() (blendervr.console.qt.console.GUI method), 9
 update_user_files() (blendervr.console.logic.console.Logic method), 8
 updateStatus() (blendervr.console.qt.console.GUI method), 9
 User, 26
 User (class in blendervr.player.user), 27
 User (class in blendervr.plugins.osc.virtual_environment.user), 29
 User (class in blendervr.plugins.osc.xml), 30
 user_position() (blendervr.interactor.Interactor method), 18
 user_position() (blendervr.processor.base.Processor method), 33

V

vector_3() (blendervr.player.buffer.Buffer method), 25
 vector_4() (blendervr.player.buffer.Buffer method), 25
 ViewPoint (class in blendervr.interactor.viewpoint), 17
 viewpointScale (blendervr.interactor.viewpoint.ViewPoint attribute), 18
 virtual_environment (blendervr.plugins.base.Base attribute), 33
 VirtualEnvironment, 26
 VISIBILITY (blendervr.player.network.synchronizer.objects.item_object.Object attribute), 21
 VRPN (class in blendervr.plugins.vrpn.virtual_environment), 31

W

wait_for_everybody() (blendervr.player.Main method), 28
 wait_for_everybody() (blendervr.player.network.connector.Master method), 24
 wait_for_everybody() (blendervr.player.network.connector.Slave method), 24
 which() (blendervr.console.xml.base.XML method), 12
 worldTransform (blendervr.player.user.User attribute), 27
 write() (blendervr.console.gui.logger.Logger method), 6
 write() (blendervr.player.network.controller.Controller method), 24
 write() (blendervr.tools.logger.Console method), 35
 write() (daemon.Daemon method), 5

X

XML (class in blendervr.console.xml.base), 12